TESIS DE LA UNIVERSIDAD DE ZARAGOZA
2022
56

Jacobo Ayensa Jiménez

Study of the effect of the tumour microenvironment on cell response using a combined simulation and machine learning approach. Application to the evolution of Glioblastoma

Director/es
Doblaré Castellano, Manuel
Doweidar Mohyeldin, Mohamed

http://zaguan.unizar.es/collection/Tesis

ISSN 2254-7606

Prensas de la Universidad
Universidad Zaragoza

© Universidad de Zaragoza
Servicio de Publicaciones

ISSN 2254-7606

Universidad Zaragoza
1542

Tesis Doctoral

STUDY OF THE EFFECT OF THE TUMOUR MICROENVIRONMENT ON CELL RESPONSE USING A COMBINED SIMULATION AND MACHINE LEARNING APPROACH. APPLICATION TO THE EVOLUTION OF GLIOBLASTOMA

Autor

Jacobo Ayensa Jiménez

Director/es

Doblaré Castellano, Manuel
Doweidar Mohyeldin, Mohamed

UNIVERSIDAD DE ZARAGOZA
Escuela de Doctorado

Programa de Doctorado en Ingeniería Mecánica

2022

Repositorio de la Universidad de Zaragoza – Zaguan http://zaguan.unizar.es

University of Zaragoza

Programa de Doctorado en Ingeniería Mecánica

Ph. D. Dissertation

Study of the effect of the tumour microenvironment on cell response using a combined simulation and machine learning approach. Application to the evolution of Glioblastoma

Author:

Jacobo Ayensa Jiménez

January 2022

Advisors:

Prof. Manuel Doblare
Department of Mechanical Engineering
Universidad de Zaragoza

Prof. Mohamed H. Doweidar
Department of Mechanical Engineering
Universidad de Zaragoza

Ayensa Jiménez, Jacobo:
Study of the effect of the tumour microenvironment on cell response using a combined simulation and machine learning approach. Application to the evolution of Glioblastoma
Ph. D. Dissertation, University of Zaragoza, 2022.

Abstract

Cancer is the second leading cause of death in the world. In particular, Glioblastoma is the deadliest and most frequent tumour affecting brain tissues. The paradigms focused only on tumour cells and genetics have nowadays ushered in a different scenario that integrates different cell populations, physical signals and a great variety of external stimuli. Mathematical models for evaluating cancer progression have been demonstrated to be, in conjunction with experimental data, a valuable tool for a better understanding, prognosis and therapy design or *in silico* drug testing. However, despite their predictive capacity, these models fail to unravel hidden complex biological processes, frequently masked due to the high non-linearity and coupling of the different phenomena involved.

Recently, a new paradigm is rising in Simulation-Based Engineering and Sciences, which aims for incorporating Data Science and Artificial Intelligence tools to conventional mathematical modelling of physical systems. In the last decades, these tools have been shown to posses an enormous predictive power, though some scepticism exists about their "black box" nature. This lack of explanatory capacity may be supplied by centuries of research, which have resulted in high knowledge about our environment and cannot fall on deaf ears. Cancer processes are not the exception.

In this thesis, I propose a new approach for the analysis of tumour cell evolution, combining the intensive use of data and Machine Learning tools with the explanatory capacity of physical models about cancer progression, something nowadays possible thanks to the great advances in cell culture monitoring technologies. This approach is included in what nowadays is known as Physically-Informed Data Science.

First, I adapt some of these methods to lower data quality contexts (missing or noisy data). Next, I build a computational framework in which I reproduce Glioblastoma evolution in microfluidic devices under different experimental configurations, and, what is more important, unveil the metabolic switch from migratory to proliferative cell activity occurring during Glioblastoma progression. This is done by means of a special family of artificial neural networks, able to represent non-measurable data and relationships within a consistent physical context. These neural networks learn faster, are less data demanding, and have an extra filtering capacity than conventional ones. Moreover, they outperform standard parametric approaches in both predictive and explanatory capacity when used for analysing Glioblastoma progression.

In combination with mathematical models and computational simulations, a workflow integrating patient-specific data and cell culture monitoring lays the foundations of personalised medicine, and lets us glimpse an encouraging future in the fight against Glioblastoma.

Resumen

El cáncer es la segunda causa de muerte en el mundo. En particular, el Glioblastoma es el cáncer cerebral más común y también el más letal. Los paradigmas centrados únicamente en las células tumorales y en la genética han dado paso en la actualidad a un escenario diferente, en el que se integran diferentes poblaciones celulares, señales físicas y una gran variedad de estímulos externos. Los modelos matemáticos para evaluar la progresión del cáncer han demostrado ser, combinados con datos experimentales, una herramienta de gran valor para una mejor comprensión y pronóstico, así como para facilitar el diseño de terapias o permitir pruebas de fármacos *in silico*. Sin embargo, a pesar de su capacidad predictiva, estos modelos no consiguen desentrañar complejos procesos biológicos, a menudo ocultos debido a las grandes no linealidades, y los importantes acoplamientos existentes entre los diferentes fenómenos implicados.

Recientemente, está surgiendo un nuevo paradigma en la Ingeniería y Ciencias Basadas en Simulación, que apunta a la incorporación de la Ciencia de Datos y la Inteligencia Artificial al modelado matemático convencional de sistemas físicos. De hecho, en las últimas décadas, se ha demostrado que estas herramientas poseen un enorme poder predictivo, aunque existe cierto escepticismo debido a su naturaleza de "caja negra". Esta falta de capacidad explicativa puede ser suplida por siglos de investigación que han resultado en grandes conocimientos sobre nuestro entorno que no pueden caer en el olvido. Los procesos tumorales no son una excepción.

En esta tesis, propongo una nueva aproximación al análisis de la evolución tumoral, combinando el uso intensivo de datos y herramientas de Aprendizaje Automático con la capacidad explicativa de los modelos físico-matemáticos sobre la progresión del cáncer, algo que es hoy posible gracias a los grandes avances en las técnicas de monitorización de cultivos celulares. Este enfoque se enmarca en lo que actualmente se conoce como Ciencia de Datos Informada por la Física.

En primer lugar, adapto algunos de estos métodos a ámbitos con datos de baja calidad (datos incompletos o con ruido). A continuación, desarrollo un marco computacional en el que se reproduce la evolución del Glioblastoma en dispositivos microfluídicos en distintas configuraciones experimentales y, lo que es más importante, se revela la transición metabólica entre los comportamientos migrativo y proliferativo que tienen lugar durante la progresión del Glbiolastoma. Esto se consigue mediante el uso de una familia especial de redes neuronales artificiales, capaces de representar datos y relaciones no medibles dentro de un contexto físico consistente. Estas redes neuronales aprenden más rápido, requieren menos datos y tienen mayor capacidad de filtrado que las convencionales. Además, superan a los enfoques paramétricos

estándar tanto en capacidad predictiva como explicativa cuando se utilizan para analizar la progresión del Glioblastoma.

En combinación con los modelos matemáticos y las simulaciones computacionales, un flujo de trabajo integrando datos de paciente específico y monitorización de cultivos celulares puede ayudar a sentar las bases de la medicina personalizada, y nos permite atisbar un futuro más alentador en la lucha contra el Glioblastoma.

Agradecimientos (Acknowledgments)

Si tuviera que enumerar una a una a las personas que han contribuído a esta tesis de una forma o de otra, no acabaría nunca. Hay decenas, incluso cientos de personas que han forjado mi carácter, han aportado ideas, han moldeado mi pensamiento, me han apoyado o acompañado, o directamente han propuesto algunas de las ideas que hay aquí. Pero como sé que esta sección de la tesis es la única que mucha gente leerá, cosa que entiendo, dado su volumen, que menos que pararse unos segundos a agradecer, aunque sea someramente, a algunas personas o colectivos en particular.

En primer lugar me gustaría agradecer a mis directores, Manuel Doblaré y Mohamed H. Doweidar. Manolo, de tu talento ya habla tu currículo, así que en ello no me voy a detener. En lo que sí que voy a hacerlo es en la que yo creo que es una de tus mayores virtudes: eres enormemente inspirador. Ese aura es la que ha hecho que me mude dos veces de ciudad (en una ocasión, eso sí, jugabas "en casa"), y que empiece dos tesis doctorales diferentes (¿o deberíamos decir tres?). Trabajar con colegas así es, sin duda alguna, de lo más estimulante que uno puede encontrar en su carrera profesional. Al margen de esto, gracias por tu confianza y por tu exigencia, que seguramente haya podido frustrarme en ocasiones, pero que ha exprimido lo mejor de mi. Parafraseando a Mario Andretti, "si crees que lo tienes todo bajo control, es que no vas lo suficientemente rápido". Mohamed, a ti tengo que agradecerte que desde el principio, cuando llegué de la mano de Manolo, me acogiste con los brazos abiertos. Me has recibido en tu despacho siempre que lo he necesitado (recuerdo ese primer año de tesis en el que revisábamos juntos los códigos intentando depurarlos). Siempre has estado pendiente de que me adaptara lo mejor posible al mundo académico, ejerciendo de contrapunto al idealismo investigador de Manolo, al que es imposible no sucumbir, en aspectos que al final resultan igualmente importantes, como acumular los méritos de docencia y de investigación que se esperan de mí en la carrera académica.

I would like to dedicate a sincere thank you to Prof. Eamonn Gaffney, from the Mathematical Institute at Oxford University. He opened his arms to any kind of collaboration, despite the awkward pandemic situation. He gave me the opportunity of enjoying again with maths, and he was always willing to share all his knowledge with me. I hope that, once this situation will over once and for all, we could meet in person and work in more challenging problems.

Aunque, como he empezado diciendo, en esta tesis seguro que hay trazas de muchísima gente, quiero destacar por encima de todos ellos a mis colaboradores más cercanos. A Tea por ser las manos experimentales que yo nunca podría ser, y que alimentan mis modelos, y a Iñaki por proponer ideas sin descanso, aunque con los años haya aprendido a dejar pasar la mayoría de ellas. A José Antonio, por sus valiosas

aportaciones científicas, sus muchas ideas y tantas soluciones. Algún día sacaré tiempo para ellas. A Marina, porque es difícil encontrar a alguien con quien sea tan productivo trabajar y por lo muchísimo que he aprendido trabajando contigo.

Esta tesis no hubiera sido posible sin el entorno privilegiado en el que me encuentro, desde el punto de vista humano y profesional. En primer lugar, agradecer a mis compañeros, colegas y amigos de TMELab por hacer agradable el trabajo día a día (Guillermo, María, Sandra, Alodia, Isabel, Tea, David, Marina, Laura, Héctor, Sara, Clara, Claudia, Vira). En segundo lugar, por permitir este entorno tan singular en el que la "gente computacional" tenemos contacto continuo con el problema biológico y, aunque menos de lo que desde nuestra cómoda e ilusa posición nos gustaría, datos experimentales reales. Me gustaría hacer un inciso para agradecer enormemente a Marina su ayuda durante la redacción del manuscrito: sugiriendo mejoras, ayudándome con la maquetación y con las figuras, revisando el inglés y detectando erratas, así como por su apoyo moral en estos meses de estrés. El manuscrito no tendría un acabado tan redondo sin tu ayuda.

To write and to publish is sometimes a hard task from which at the last step, the authors take the credit. Here I would like to expressly thank the anonymous reviewers that have peer reviewed my work. Some of their ideas and proposals have been reflected in my work and in this precise manuscript. It is therefore fair to highlight that point.

Finalmente, yo no sería lo que soy, ni tendría las motivaciones que tengo, si no hubiera sido por todos los profesores que he tenido, desde mi infancia hasta la actualidad, tanto del Liceo Francés de Zaragoza, como de la Universidad Politécnica de Catalunya, la Universidad de Sevilla, la Universidad Nacional de Educación a Distancia y la Universidad de Zaragoza. A todos ellos. *Merci*, *Gràcies*. Gracias.

Dicho esto, y como es habitual decir.

Seguimos.

Contents

List of Figures

List of Tables

List of Code Listings

List of Algorithms

List of Acronyms

t-SNE	t-distributed Stochastic Neighbour embedding
AI	Artificial Intelligence
ANN	Artificial Neural Network
ASA	Adjoint Sensitivity Analysis
BED	Bayesian Experimental Design
BINN	Biologically-Informed Neural Networks
BNN	Bayesian Neural Network
BSDE	Backward Stochastic Differential Equation
CDF	Cumulative Distribution Function
CF	Cost Function
CM	Computational Mechanics
CNN	Convolutional Neural Network
DAE	Differential-Algebraic Equation
DD	Data-Driven
DDDAS	Dynamic Data-Driven Application Systems
DDSBES	Data-Driven Simulation-Based Engineering and Sciences
DDSBP	Data-Driven Simulation-Based Problem
DDSBS	Data-Driven Simulation-Based Statistics
DEM	Diffuse Element Method
DIC	Digital Image Correlation
DL	Deep Learning
DMD	Dynamic Mode Decomposition
DoE	Design of Experiments
ECDF	Empirical Cumulative Distribution Function
ECM	ExtraCellular Matrix
FD	Finite Differences
FDM	Finite Difference Method
FE	Finite Elements
FEM	Finite Element Method
FPGA	Field Programmable Gate Array

FSA	Forward Sensitivity Analysis
FVM	Finite Volume Method
GBM	Glioblastoma
GCNN	Graph Convolutional Neural Network
GDL	Geometric Deep Learning
GENERIC	General Equation for the Non-Equilibrium Reversible-Irreversible Coupling
GP	Gaussian processes
GPLVM	Gaussian Process Latent Variable Model
GPU	Graphical Processing Units
HDR	Highest Density Regions
HIF	Hypoxia-Inducible Factor
HMM	Hidden markov Model
IoT	Internet of Things
kPCA	Kernel Principal Component Analysis
LD	Listwise Deletion
LLE	Locally Linear Embedding
MAD	Mean Absolute Deviation
MCM	MonteCarlo Method
MCMCM	Markov Chain MonteCarlo Method
ML	Machine Learning
MLE	Maximum Likelihood Estimation
MLP	MultiLayer Perceptron
MNAR	Missing Not At Random
MoM	Method of Moments
MOR	Model Order Reduction
MRI	Magnetic Resonance Imaging
MSE	Mean Square Error
MSLE	Mean Square Logarithmic Error
NEM	Natural Element Method
NLPCA	NonLinear Principal Component Analysis
NN	Neural Networks
ODE	Ordinary Differential Equation
PCA	Principal Component Analysis
PD	Pairwise Deletion
PDE	Partial Differential Equation
PDF	Probability Density Function
PGD	Proper Generalized Decomposition
PGDS	Physically-Guided Data Science

PGNN	Physically-Guided Neural Networks
PGNNIV	Physically-Guided Neural Networks with Internal Variables
PIDS	Physically-Informed Data Science
PIL	Predefined Internal Layer
PINN	Physics-Informed Neural Network
POD	Proper Orthogonal decomposition
QoI	Quantity of Interest
RANS	Reynolds-Averaged Navier-Stokes
RBDD	Reliability-Based Data-Driven
RBDDSBP	Reliability-Based Data-Driven Simulation-Based Problem
RBF	Radial Basis Function
RBM	Reduced Basis Method
ReLU	Rectified Linear Unit
RMSE	Root Mean Square Error
RNN	Recurrent Neural Network
ROM	Reduced Order Model
RVE	Representative Volume Element
SBES	Simulation-Based Engineering and Science
SGD	Stochastic Gradient Descent
SI	Single Imputation
SOM	Self-Organizing Map
SPDE	Stochastic Partial Differential Equation
SPH	Smoothed Particle Hydrodynamics
SPNN	Structure Preserving Neural Network
SQF	Stochastic Quadratic Form
SVD	Singular Value Decomposition
SVM	Support-Vector Machine
TDA	Topological Data Analysis
TGNN	Theory-Guided Neural Network
TME	Tumor Micro-Environment
TPU	Tensorial Processing Units
UQ	Uncertainty Quantification
WKBJ	Wentzel-Kramers-Brillouin-Jeffreys
WYPIWYG	What You Prescribe Is What You Get
XAI	eXplainable Artificial Intelligence

List of Symbols

General

$\mathcal{P}(A)$	Power set of a set A
$\cap$	Set intersection
$\cup$	Set union
$\sqcup$	Disjoint set union

Probability

$\mathbb{E}[\cdot]$	Expected value of the random variable or vector
$\mathbb{P}[\cdot]$	General position operator of the random variable or vector
$\mathrm{Var}(\cdot)$	Variance of the random variable or vector
$\mathrm{Cov}(\cdot, \cdot)$	Covariance of two random variables
$\mathbf{Cov}(\cdot)$	Variance - covariance matrix of the random vector
$\mathcal{N}(\mu, \sigma)$	Normal distribution with expected value μ and standard deviation σ
$\mathcal{U}(a, b)$	Uniform distribution at the interval $[a; b]$
$\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$	Multivariate normal distribution with expected value $\boldsymbol{\mu}$ and variance - covariance matrix $\boldsymbol{\Sigma}$
$\mathcal{GP}(\mu, k(x, x'; \boldsymbol{\theta}))$	Gaussian process with expected value μ and kernel $k(x, x')$ dependent on the parameters $\boldsymbol{\theta}$

Operators

$\mathrm{Tr}(\cdot)$	Trace of a matrix
$\det(\cdot)$	Determinant of a matrix
$\mathrm{MSE}(\cdot)$	Mean Square Error function
$\mathrm{RMSE}(\cdot)$	Root Mean Square Error function

Introduction

1. Introduction

1.1 Object and scope of the thesis

Cell processes and cancer

Cells are the major players in all biological processes, constantly adjusting their structure and function to the changing demands of their environment, with the aim of keeping their intracellular state within a narrow range of physiological parameters. As cells receive physiological or pathological (chemical or physical) stimuli, they modify their shape, location, internal structure and even their genome expression, achieving a new state while maintaining their viability and overall function. The main cell adaptative responses are hypertrophy (enlargement of individual cells) (Hunter and Chien, 1999), hyperplasia (increase in cell number) (Goss, 1966), atrophy (reduction in size and cell number) (Schwartz, 2008), and metaplasia (transformation from one type of epithelium to another) (Slack and Tosh, 2001). These processes modify the cell's ability to proliferate, migrate, differentiate, produce extracellular matrix, or express particular proteins. If the adaptive capacity of the cell is exceeded or if the external stimulus is inherently harmful, cell damage can occur. Within certain limits, that injury is reversible, and cells return to a stable baseline, while severe or persistently high cyclical stimulation can result in irreversible injury and death of the affected cells (Kumar et al., 2014).

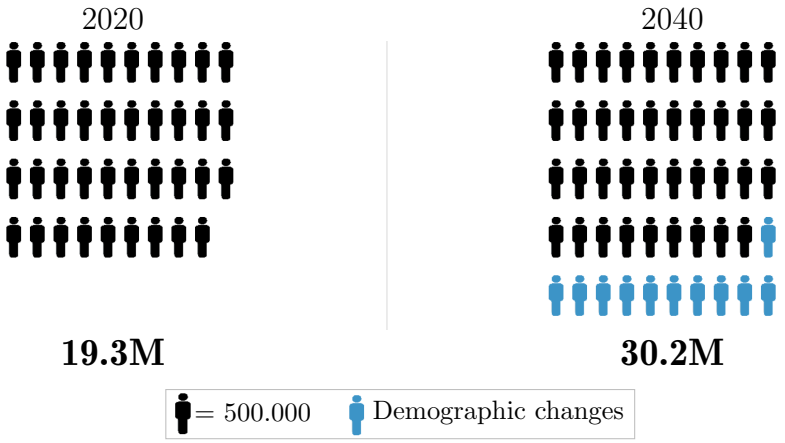
Cell adaptation and cell death are crucial in physiological processes such as embryogenesis, homeostasis organ development, repair and long-term adaptation of tissues and organs. But they are also key in many pathological processes and diseases such as atherosclerosis or cancer (Fodale et al., 2011). All these processes are characterised by a changing microenvironment subjected to internal and external stimuli and various cell phenotypes that interact with each other, inducing different signals that are received and interpreted by other cells, thus producing changes within themselves and in the extracellular matrix (ECM). Understanding the mechanisms of cell adaptation to short and long-term changes in their microenvironment (e.g. oxygen and nutrient levels, substrate stiffness, mechanical stress, drug dose, etc.) would help to better understand processes such as those mentioned above and to develop new therapeutic

strategies aimed at promoting or blocking desirable or undesirable cellular adaptation events respectively.

A particular but very important case is the development of tumours. Cancer is the leading cause of death in developed countries and the second in developing ones (Siegel et al., 2016). Out of the 19.3 million new cancer diagnoses in 2020 worldwide, there were 10.0 million deaths, and at that time, 32.6 million people had a 5-year or less survival from diagnosis. Approximately one-sixth of the total number of deaths are caused by cancer. Furthermore, the GLOBOCAN 2020 database (Sung et al., 2021) estimates an overall increase of 47% in cancer incidence in 2040 with respect to 2020 (see Fig. 1.1). This means that approximately 15-20 million people will die from cancer every year in the next two decades. The cancer landscape has changed dramatically in the last decades, proving to be a process much more complex than initially thought. The old paradigms focused only on tumour cells and genetics have become a different scenario that integrates different cell populations, ECM, chemical gradients (oxygen, nutrients) and physical signals such as local deformation and stiffness (Nagelkerke et al., 2015; Orr et al., 2006), all forming a much more complex and multi-interactive system, the so-called tumour microenvironment (TME) (Hanahan and Weinberg, 2011). Furthermore, tumour cells evolve differently within the same tumour, following different evolutionary paths, causing the appearance of specific microenvironments in each type of cancer and tissue, which explains the high heterogeneity of tumours (Quail and Joyce, 2013).

While normal cells require specific physiological signals to proliferate, tumour cells can divide independently of these signals and ignore those that inhibit growth. Furthermore, they acquire unlimited replicative potential (a characteristic that normal cells do not possess) and are capable of evading apoptosis signals (Harrington, 2011). In this situation, the exacerbated consumption of oxygen and nutrients leads to hypoxia and malnutrition, thus subjecting cells to extreme stress. This harsh environment favours the survival of cells capable of resisting and adapting to these highly demanding conditions (Espina and Liotta, 2011). In addition, tumour cells are also capable of changing the TME by promoting new blood vasculature that improves the supply of oxygen and nutrients to help maintain their growth, as well as the colonisation of new tissues through the process known as metastasis (Mehlen and Puisieux, 2006).

To summarise, tumour development is a highly dynamic process that originates from genomic instability induced by external signals generated by other cells or external stimuli modulated by the particular microenvironment. Moreover, when a given treatment is applied (surgery, chemotherapy, radiotherapy, immunotherapy, hormones or combination therapy), the tumour and its microenvironment undergo significant alterations. This leads tumour cells to proliferate and generate harsh microenvironments that promote the death of surrounding cell types and the survival of tumour cells that are more adaptable and resistant. In Hanahan and Weinberg (2011), a comprehensive review of the different processes involved in cancer is described and the general panorama is depicted, in what is now known in the literature as *the hallmarks of cancer* (Fouad and Aanei, 2017; Hanahan and Weinberg, 2000, 2011). Fig. 1.2 summarises the processes involved as well as the therapies targetting each of the components of the system.



(a) Incidence.

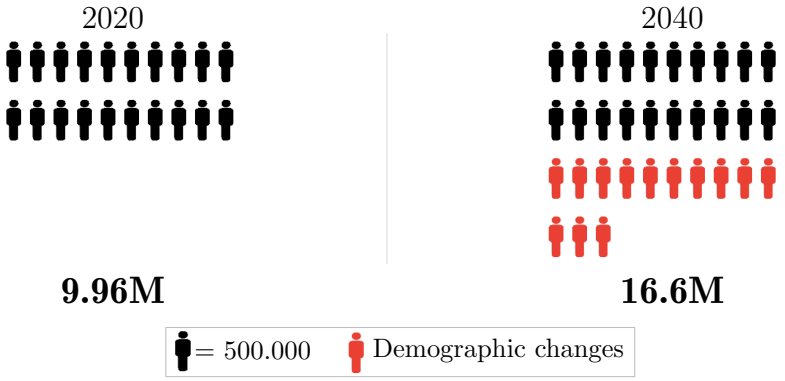


Figure 1.1: Cancer incidence and mortality in figures. The figure illustrates the incidence change foreseen for the next 20 years due only to demographic changes. Source: GLOBOCAN 2020 Database (Sung et al., 2021).

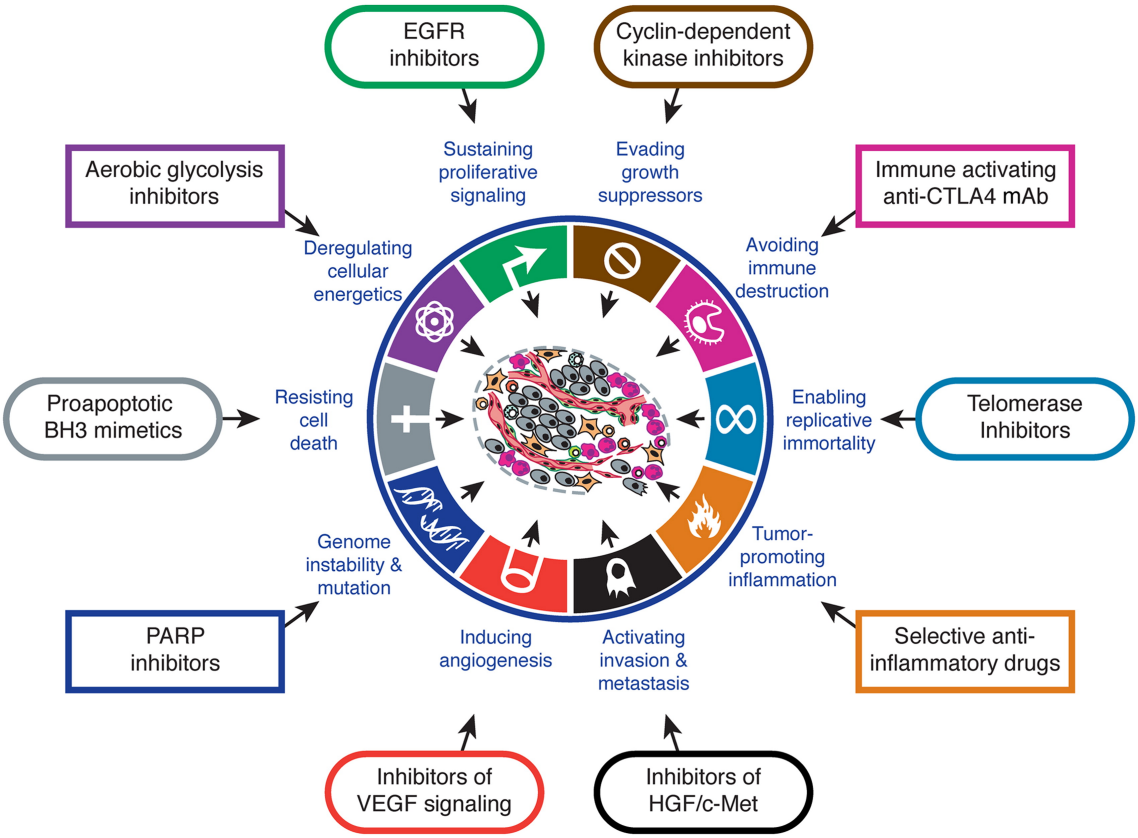
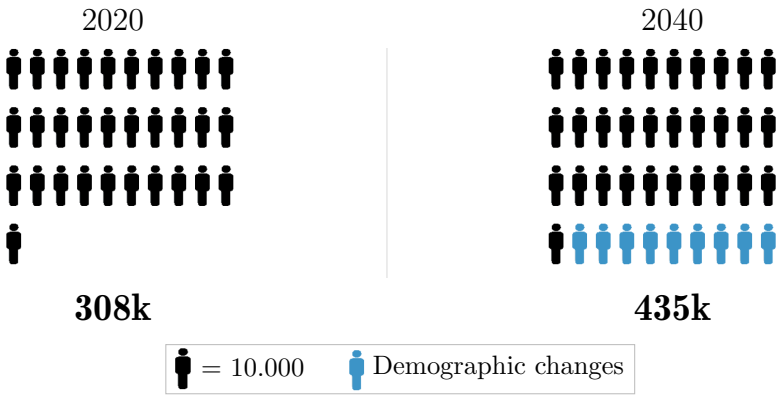


Figure 1.2: The hallmarks of cancer. The different processes involved in cancer genesis and progression and the therapeutic strategies targetting each of them. Figure extracted from Hanahan and Weinberg (2011).

Among the more than 200 types of cancer, glioblastoma (GBM) is the most aggressive and common primary brain tumour, accounting for 17% of these tumours (Brat, 2012). These figures entail approximately 43.000 deaths and will rise to 63.000 by 2040 (see Fig. 1.3). Indeed, survival of GBM patients undergoing the first-line standard treatments (surgery followed by local radiotherapy with concomitant and adjuvant chemotherapy) has a median of 14 months since diagnosis and the 5-year survival rate is only 6.8% (Oike et al., 2013; Ostrom et al., 2013).



(a) Incidence.

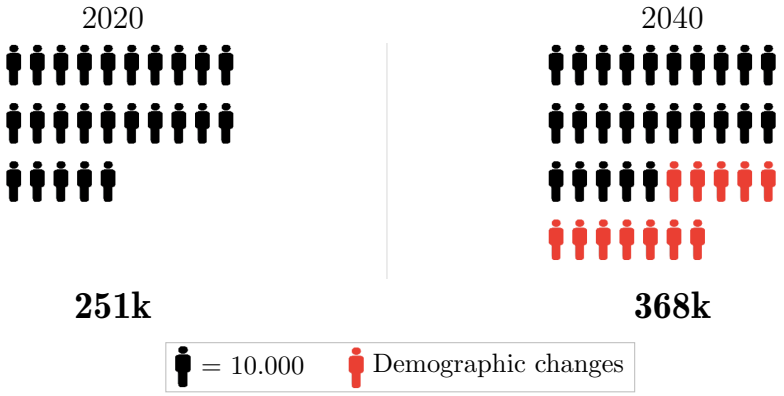


Figure 1.3: Brain and nervous system cancer incidence and mortality in figures. The figure illustrates the incidence change foreseen for the next 20 years due only to demographic changes. Source: GLOBOCAN 2020 Database (Sung et al., 2021).

GBM is a highly infiltrating and rapidly progressive tumour, characterised by two main histopathological conditions: necrotic foci, typically surrounded by areas of high cellularity known as pseudopalisades, and microvascular proliferation (Brat et al., 2004; Jensen, 2009). At an early stage, proliferation and secretion by GBM cells of pro-coagulant signals would cause thrombotic events, leading to hypoxia and nutrient depletion. As a consequence, GBM cells begin to migrate towards enriched regions guided by nutrient and oxygen gradients. This leads to the generation of the GBM pseudopalisade feature. These migrating cells would reach another healthy blood vessel, allowing the proliferation of GBM cells. Eventually, the GBM cells will again cause this blood vessel to collapse, restarting the process and creating a growing wave of migrating tumour cells through the brain (see Fig. 1.4). Therefore, it has been proposed that one of the driving forces behind glioma aggressiveness is the lack

of nutrients and oxygen due to thrombotic events (Jensen, 2009). The causes of these densely populated pseudopalisades remain poorly understood. Although the high cellularity was initially thought to be due to the rapid proliferation of GBM cells, recent histological studies have shown that proliferation in areas of pseudopalisades is significantly lower while apoptosis is substantially greater than in neighbouring regions. This suggests that pseudopalisades are due to causes other than simply a higher proliferation or survival rate (Jensen, 2009).

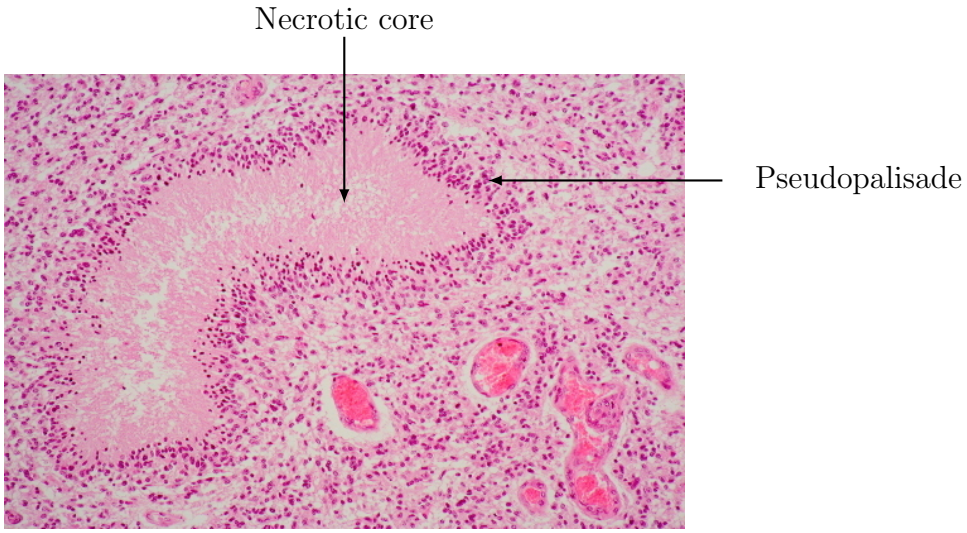


Figure 1.4: Histological features of GBM. The two main histological features: necrotic cores around occluded vessels and pseudopalisades migrating towards functional vessels. Source: *The Internet Pathology Laboratory for Medical Education*: `https://webpath.med.utah.edu/`.

Due to the complexity, heterogeneity, and dynamic changes that take place in the TME, it is difficult to investigate *in vivo* all possible interactions in the tumour and the surrounding stroma. Therefore, laboratory experiments are designed to address these questions at different scales and to best recreate the spatial and structural complexity of cellular distributions and of physiological and pathological environments. However, there is still a significant lack of predictive power of currently available *in vitro* models, being this one of the main reasons for the continuing decline in the number of new drugs appearing per year and billion dollars invested (Nolan, 2007; Scannell et al., 2012), as illustrated in Fig. 1.5.

Among the most important characteristics of *in vitro* models to reproduce with reasonable accuracy the dynamic behaviour of physiological or pathological biological processes such as tumours, structural three-dimensionality plays a central role. The three-dimensional microenvironment affects cell behaviour in many ways through adhesive, mechanical, and chemical signals (Edmondson et al., 2014). Despite this well-known fact, cells are generally grown in traditional Petri dishes where the complexity of the tumour is mostly lost. Therefore, the behaviour of tumour cells is dramatically different in these Petri dishes than that observed in real tissue, so important efforts have been made to reproduce the three-dimensionality of the TME in *in vitro* experiments (Nath and Devi, 2016; Wang et al., 2014).

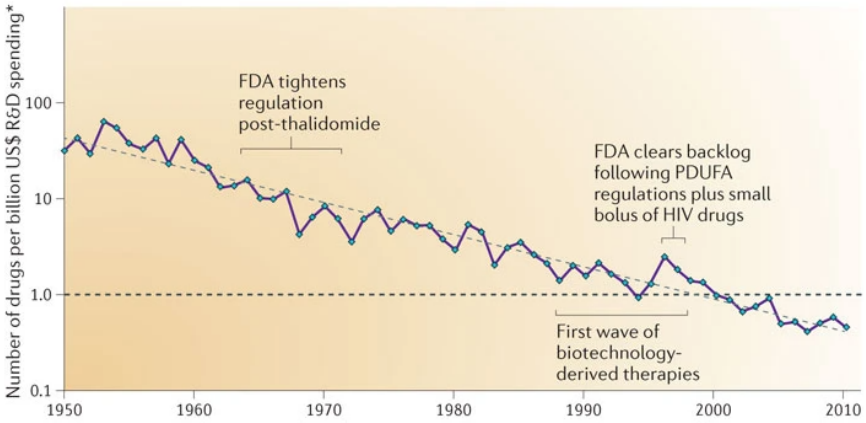


Figure 1.5: Decline in drug discovery over years. The number of new drugs approved by the US Food and Drug Administration (FDA) per billion US dollars (inflation-adjusted) spent on R&D has halved roughly every 9 years. Figure extracted from Scannell et al. (2012).

Microfluidic technologies

Recently, microfluidics and micro-technologies have emerged as powerful tools to recreate the complex microenvironment that governs tumour dynamics (Sackmann et al., 2014). These techniques make it possible to reproduce important characteristics of tumour evolution that were not previously observed in 2D cultures, as well as to test drugs in a much more efficient way and in a controlled environment (Guckenberger et al., 2015; Shin et al., 2012).

One of the most exploited applications of microfluidic devices has been precisely cell culture, since they provide a much more realistic scenario to study these processes than classic Petri dishes. In particular, one of the most important application frameworks for microfluidics has been the study of tumour chemotaxis (Mosadegh et al., 2008; Tatárová et al., 2016). Hydrogels are interesting options to generate chemical gradients within microfluidic devices since they oppose a high resistance to interstitial flow, while having enough porosity to allow the diffusion of biological molecules (Ramanujan et al., 2002). This approach has been exploited to study many different chemotactic processes, such as tumour-induced angiogenesis (Jiménez-Torres et al., 2016), tumour invasion (Boussommier-Calleja et al., 2016), tumour intravasation (Zervantonakis et al., 2012), tumour extravasation (Jeon et al., 2015) and metastasis (Bersini et al., 2014).

Other important applications are: i) analyses of the effect of hypoxia achieved in standard cell cultures through different methods: spheroid cell culture, chemical preconditioning and incubation in hypoxic chambers, and characterised by quantification of HIF-1α (Western Blotting, Immunofluorescence and Luciferase assays) or by directly measuring oxygen concentration (Wu and Yotnda, 2011); ii) cell co-culture using distinctive fluorophores, or performing immunofluorescence assays with distinctive markers to distinguish different cell phenotypes (Nam et al., 2003); iii) Cell mechanotaxis including cell migration, changes in cell shape and differentiation due to changes in the stiffness of the extracellular matrix (Von Der Mark et al., 1977).

In conclusion, microfluidic systems are capable of reproducing physiological or pathological microenvironments such as tumours in a way that is not easy to achieve using traditional cell culture systems, thus bridging the gap between traditional *in vitro* research and *in vivo* models, shedding light on the tumour process as well as increasing the effectiveness of drug development (Katt et al., 2016). Of course, these technologies are still emerging and many technical issues have to be addressed before they are fully disseminated in the biomedical community.

The need for mathematical models and data assimilation tools

Despite these new experimental possibilities, the complexity and heterogeneity of the TME, as well as its dynamic interactions with tumour cells, make it difficult to separate effects, control new hypotheses, quantify the effect of each parameter or predict the result of potential ("what if") situations even in these most realistic and controlled conditions. To achieve this, the only way to proceed is to combine the new possibilities of *in vitro* assays with the quantitative power and versatility of mathematical modelling and computational techniques (Altrock et al., 2015; Byrne, 2010; Byrne et al., 2006; Hatzikirou et al., 2005). *In silico* models, based on cancer biology and driven by experimental or clinical data, provide critical tools for integrating knowledge across different biological scales, combining tools from various scientific fields, conducting rigorous quantitative analyzes, and testing hypotheses in a cheaper, faster and more rigorous way. The *in vitro* and *in vivo* models will then feed these mathematical models with data for calibration and validation, as well as for the estimation of particular parameters.

Many studies have focused on a better understanding and quantification of the effect of different parameters (oxygen, biochemical molecules, ECM stiffness, cell proliferation rate, among others) on tumour evolution (Bearer et al., 2009; Deisboeck et al., 2011; Kim et al., 2016). In particular, in Rejniak (2016), in addition to reviewing the mathematical models available to incorporate the main components of the tumour microenvironment, the different authors analyse aspects such as the importance of the hypoxic environment for the progression of GBM and in the formation of cellular pseudopalisades. The role of the biophysical and biomechanical properties of ECM has been suggested as a crucial component in tumour cell invasion, including the mechanical aspects of circulating tumour cells that enable them to survive (Ayuso et al., 2017), as well as tumour vasculature formation (angiogenesis).

Despite the important advances achieved by the combination of experimental and modelling tools towards the understanding of complex biological processes such as tumour evolution, there are many aspects that remain hidden and unknown, and important correlations between variables such as genomics, proteomics, and metabolomics, as well as their relations to macroscopic characteristics such as bioimaging biomarkers (Adams, 2015). Recently, the powerful tools of artificial intelligence (AI), based on the smart treatment of available data, are helping researchers to unravel these complex internal interactions, extracting hidden characteristics and allowing a more complete predictive power than physical standard models (Bishop, 2006).

From the rise of data mining and massive data generation, the next revolution is emerging in terms of generating new knowledge and improving our predictive

capacity in almost any field (LeCun et al., 2015). In particular, in life sciences, the enormous volume of information generated by various high-throughput next-generation sequencing studies, as well as structured and unstructured biomedical data from electronic and digital health, is facilitating the emergence of a new vision in which prediction, personalisation and prevention will be the pillars of medicine in the XXI-st century (Hansen et al., 2014).

Cancer, as discussed before, is an incredibly complex disease, always changing, evolving and adapting. Data Science techniques, and in particular AI and machine learning (ML) tools may play an important role in better understanding its causes and evolution. For instance, clinicians and researchers use snapshots of the tumour's genetics, as well as temporary dynamic changes in the cancer features. Huge amounts of information of different characteristics are generated: molecular at DNA and RNA level, proteins and metabolites, images at the cell, tissue and organ scales, and many other types at the individual and population levels related to the environment, habits, the environmental characteristics and context. This sea of data is not only enormous in volume and complex in structure, but also vast in dynamic scale and depth, loosely connected, inherently noisy, and heterogeneous, thus requiring a holistic approach to the disease (Cho, 2015).

Nowadays, more and more databases are being established. For example, the Catalogue of Somatic Mutations in Cancer (COSMIC), which is the largest database of somatic mutations and their effects on human cancer, the Dialogue of Reverse Engineering Assessment and Methods (DREAM) consortium that identifies the best computational methods to infer genetic heterogeneity from sequencing data provided by a large multi-institutional community of cancer genomics projects, including the International Cancer Genome Consortium and The Cancer Genome Atlas. This is where the Big Data framework and ML have a major role to play. For example, in Keutgen et al. (2012), a learning model is constructed that predicts whether a patient has thyroid cancer by analysing the expression levels of specific genes. Using the genetic measurements of the nodule, the model is able to predict with more than 90% accuracy (higher than standard diagnostic tools) whether a nodule is malignant or benign. The future of cancer research and treatment is evolving thanks to these new technologies with the ultimate goal of integrating all the information to make better treatments for each individual patient (Bolouri et al., 2016).

A particular applied field where these groundbreaking techniques are of particular interest is precisely microfluidics. The broad possibilities offered by microfluidic devices related to massive data monitoring and acquisition open the door to the use of ML technologies in a very promising field: cell culture monitoring. Indeed, the combination of the increasing power of microfluidic technologies and AI tools in order to characterise cell culture behaviour, has been recently coined as *Intelligent Microfluidics* (Galan et al., 2020; Riordon et al., 2019).

From Data Science to conventional modelling: Physically-Informed Data Science

Although still recent, a paradigm shift in predictive models has begun to take place. The staggering and growing amount of information available is pushing science to a

point where predicting the evolution of a system is easier using data directly than using the hitherto useful but complex mathematical models (Marsland, 2011). These methodologies allow the extraction of relevant knowledge from unstructured data, which enables us to make precise predictions that can be continuously improved by comparing these predictions with new data (Hildebrandt, 2016). This new paradigm has undoubtedly been promoted by the increase in available data, the emergence of the internet of things (IoT) and the explosion of computing power of both high performance computers and personal laptops and smartphones. Some scientists even speculate that, given the impressive development of Big Data and computing, theoretical physics might also be tractable by its tremendous power. In Overbye (2020), a recent article published in The New York Times, the author claims that "the Theory of Everything is still not in sight, but with computers taking over many of the chores in life — translating languages, recognising faces, driving cars, recommending whom to date — it is not so crazy to imagine them taking over from the Hawkings and the Einsteins of the world".

With this paradigm shift, a sceptical trend against AI has been born, pointing out the many dangers of AI in its various forms (Besse et al., 2019; Maguolo and Nanni, 2021; Ribeiro et al., 2016). One of these dangers is its black-box nature (Bathaee, 2017; Castelvecchi, 2016), with many scientists claiming for explainability. In addition to the lacks and flaws of AI, the progressive replacement of physical knowledge by ML models in the last decades raises disturbing questions: are we witnessing the end of science as we know it? In particular, will the rapid growth of this black-box Data Science make existing theory-based models obsolete? Are we really going to trust black-box models above scientific theories? Or, in a more romantic and fashionable way: are we going to forget all the knowledge about the universe accumulated during centuries by humanity? Are we going to replace the legacy of geniuses like Einstein or Newton with the opaque algorithms of AI?

This criticism has led to the new concept of explainable artificial intelligence (XAI) with the claim of whitening the black box of conventional AI (Emmert-Streib et al., 2020; Gunning et al., 2019; Samek et al., 2019). XAI is recognised as the *sine qua non* condition for AI to continue making steady progress without disruption, in an attempt to justify, control, discover and improve the different ML algorithms (Adadi and Berrada, 2018; Gunning and Aha, 2019) and avoid past mistakes (Hryniewska et al., 2020).

One possibility for gaining explainability in ML methods is to incorporate, in one way or another, the known information about our surrounding world into the algorithms and computations of ML. This information is nothing other than the known physics about the universe. Thus, this strategy is a loophole for answering the previous questions in a satisfactory manner. This approach is situated between pure Data Sciences and Theoretical Sciences as illustrated in Fig. 1.6 and has been coined as physically-informed data science (PIDS). The idea is to promote the synergistic use of both approaches. The raw data treatment is now complemented with physical knowledge characteristic of the analysed system; for example, universal physical laws or working hypotheses.

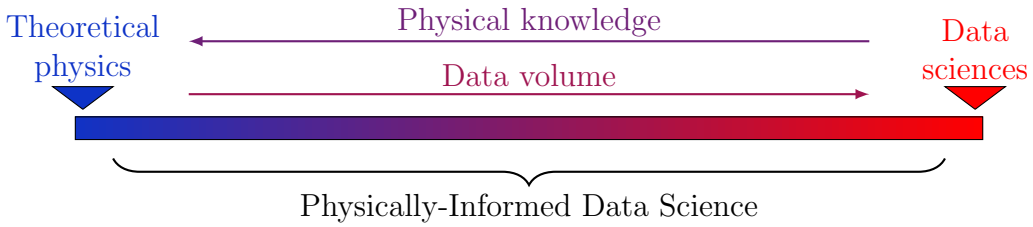


Figure 1.6: Physically-Informed Data Science. PIDS are situated between theoretical physics and data sciences.

Physically-Informed Data Science as a valuable tool in the fight against Glioblastoma

It seems, therefore, reasonable to combine all the knowledge, acquired in the last decades, about cancer in general, and about GBM in particular, with data from cell cultures in microfluidic devices. Indeed, there are a lot of mechanisms driving GBM evolution that have been mathematically treated using transport partial differential equations (PDEs) (Bearer et al., 2009; Byrne et al., 2006), including cell pedesis, migration in response to chemical species, proliferation, differentiation and death. However, there is still room for improvement as many of these mechanisms are partially known or subject to scientific controversy. This is the case of what is known in literature as the *go-or-grow* paradigm: the migration/proliferation switch in cell metabolic activity that occurs in the cell tissue when cells undergo oxygen gradients in hypoxic conditions. This particular feature, as explained before, is one of the main causes of tumour fast progression and invasive capacity.

The key idea is, therefore, to focus the predictive power of AI and in particular ML methods on this particular phenomenon, making mathematically explicit the rest of phenomena that affect the cell culture and affect its evolution. Thus, the physical knowledge about the system should not *determine*, but only *guide* the computations, whereas the ML features are intended to unravel the opaque components of the model, therefore acquiring a certain explanatory capacity. This is where PIDS can be tremendously useful, as it offers the possibility of combining sound physical knowledge with the pattern extraction capacity of ML.

Nevertheless, none of the PIDS techniques would work if there is not enough data feeding the models, and here is where microfluidic technologies, combined with the appropriate biomarkers, microscopy and image acquisition techniques, come to the rescue. Our approach is therefore multidisciplinary and looks for the synergy between Data Science (ML methods), mathematical modelling (tumour evolution models) and new technologies (microfluidics), as represented in Fig. 1.7.

1.2 Objectives and overview of the thesis

1.2.1 Objectives

From all the above, the scientific objective of this thesis is to carry out a study of the mechanisms by which changes in environmental variables (both chemical and physical) modify the cellular response in the long and short term, in the process

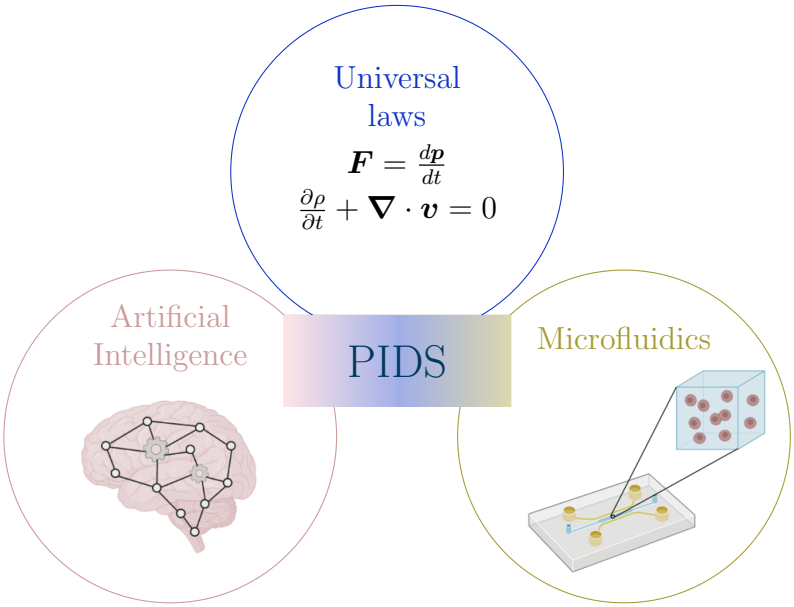


Figure 1.7: Combination of data, physics and artificial intelligence. Thanks to microfluidic technology, the combination of the acquired knowledge with experimental data and ML techniques is now a possibility.

known as cell conditioning, and how this latter affects biological processes such as tumour evolution. In particular, I am interested in analysing cell conditioning under hypoxic conditions, nutrient supply or starvation, imposition or depletion of a chemical gradient or mechanical, electrical or thermal stimuli and cellular conditioning on the efficacy of some drugs and combined therapies in GBM tumour. In this thesis, an *in silico* modelling platform will be developed based on general multiphysics computational tools, using both an in-house developed software based on 3D finite elements (FE) technology and PDE one-dimensional solvers. This platform will also allow to incorporate data obtained from measurements in the microfluidic devices, but incorporating the known physics of the problem, leaving the unknown ones as the main target of ML algorithms and methods. For such purpose, I dedicate a big effort to explore the different PIDS techniques, to adapt the existing ones to problems where data are noisy or missing and to extend them for working with internal variables, applying the most powerful AI techniques and computational tools to our problem.

Scientific goals of the thesis

- To adapt PIDS methods to more general contexts and lower quality data and to define new techniques combining the most performant methods of AI with physical knowledge, accounting for internal variables inclusion.
- To better understand the processes that govern GBM progression using data obtained from microfluidic devices and to improve their prediction by means of a combined approach of standard physically-based mathematical models and data-driven (DD) tools.

More in detail, the main objectives of this thesis are:

- **From the methodological point of view:**
  - **To formulate** an abstract and general framework for PIDS and place it in the state of the art.
  - **To adapt** the existing DD methods to contexts where data are imperfect: they are subjected to uncertainty or to incompleteness.
  - **To establish** new PIDS methods able to combine the power of deep learning (DL) methods with the known physics and to work with internal non-measurable variables.
  - **To build** a mathematical framework and a computer software for the analysis of cell cultures in microfluidic devices.
  - **To propose** an appropriate way to study the structural nonlinear dependence between the different parameters of a complex non-explicit parametric model.
  - **To exploit** the appropriate mathematical tools for analysing the dynamics of cell cultures in microfluidic devices under heterogeneous growth and chemotaxis.
- **From an application point of view:**
  - **To define and fit** a parametric model recreating the evolution of GBM under hypoxic conditions in microfluidic devices.
  - **To explore** its parametric structure and detect its weaknesses and strengths.
  - **To apply** the PIDS framework to the GBM progression for a better understanding of the different cell mechanisms driving the tumour evolution, able to combine measurements from the microfluidic devices and the known physics of the problem.
  - **To estimate** some biological features of the evolution of GBM such as propagating waves, cell concentration at the wave front and oscillatory structures, by means of mathematical models.

1.2.2 Current status

We shall start from previous works from our group, in which we demonstrated the possibility of developing pseudopalisades GBM *in vitro* (Ayuso et al., 2017). Then I shall build a general mathematical framework for simulating cell cultures evolution in microfluidic devices in response to external stimuli along with the reaction-diffusion model of various biochemicals, TME cells and interactions between cells and drugs. This framework will be particularised to GBM evolution. As an additional task, I shall explore the possibility of extracting fundamental hidden patterns in the complex behaviour of cells in GBM development by using ML approaches extracting knowledge from the data, specially images, obtained in our hybrid platform, something that is possible due to the high performant nature of microfluidic technology.

As far as I know, although some of these aspects have been treated in the literature, as presented in this introduction, there is no approach that attacks this fundamental

problem in a multidisciplinary way, combining both microfluidic technologies data acquisition capabilities, computational and multiphysical modelling with application to biological processes and the development of new PIDS techniques able to combine data and models in a proper manner.

1.3 Structure of the thesis

This thesis is singular in the sense that it collects the research done in two different directions, although with one common objective: the better understanding of complex cellular processes related to cancer. The first direction is to develop new and appropriate methods able to incorporate data in physically-based computations, and the second, to push on new mathematical models to describe the evolution of cancer, particularising them to GBM evolution *in vitro*. Thus, I present the work structured in two different parts. Part I relates to new PIDS methods while Part II focuses on GBM evolution modelling and, in particular, the application of the new presented methods to this problem.

The structure of the thesis and the dependence between chapters are summarised in Fig. 1.8 and the structure of each part is detailed next.

1.3.1 New Physically-Informed Data Science methods

This first part of the thesis is dedicated to the formulation of the general PIDS problem and the presentation of new strategies and methods in that direction. The methods are presented in a general context, and the problem statement and notations are described using an abstract mathematical framework. Due to the experience of the author, the theoretical developments are usually illustrated using the concepts and conventions of continuum mechanics. Anyway, the examples used for illustrating the methods and evaluating their performance are extracted from different fields, including solid mechanics, hydraulics, conventional data science problems, multiscale computational mechanics and heat transfer, showing the broad scope of the presented methodology. I leave the applications related to GBM to the second part of the thesis, as they require a deeper biological understanding that will be developed in advance.

The exposition begins with Chapter 2, where I present an abstract general framework of the PIDS problem. First, I introduce the concept of PIDS and I build a mathematical framework able to cover all the different PIDS approaches. I present the different notations and conventions that I shall use all along the exposition. Then, I present a taxonomy of the different PIDS approaches existing in the literature and I frame them in the scheme that has just been built. This classification is done attending both to the way that physics is included in the algorithms and to the Data Science algorithm used. The reader is warned about two major issues: first, PIDS is a hot spot of current research and a glowing field nowadays, so the listed methods and references presented are only a few of the many and the exposition is far from being exhaustive, as this would require a proper monograph. However, I have tried to present the most relevant works and research directions and the published works that have represented a milestone for the scientific community. Second, this thesis is the work of many years in a really fertile field, so it is not only possible, but indeed

certain, that some of the cited works have not only mimicked, but surpassed the scientific contents of this thesis. Even so, I decided to include them in the exposition. The chapter finishes with a brief discussion on the weaknesses and strengths of the different approaches, and puts it in relation to the work of this thesis.

Chapter 3 presents our first scientific contribution to this field, what has been coined as reliability-based data-driven (RBDD) solver. Starting from the recent work by Kirchdoerfer and Ortiz (2016b), I extended it to a stochastic framework and I discussed the performance of the method when compared to other existing PIDS methods. Also, I exploited the second-order statistical nature of the solver to describe how to use it in multiscale contexts, where the data information is statistically propagated from one scale to the other. The chapter closes with an illustration example using real data of concrete uniaxial tests. The content of this chapter is basically the one published in Ayensa-Jiménez et al. (2018). However, I have adapted the approach and the notations just to place it in the framework of this thesis in a more general context.

In Chapter 4, I extend the work to deal with incomplete data. Data quality is one of the fundamental features in Data Sciences and in particular in PIDS, so here, I address one of the main issues in data quality: the missing data problem, or data incompleteness. Even if the missing data problem has been fruitfully addressed in conventional statistics, we need new methods and procedures to situate it in the context of PIDS methods. The methods are developed using a statistical framework compatible with the stochastic nature of RBDD solvers, so this chapter, beyond its own scientific relevance, may be seen as an extension of the previous one to contexts of data paucity or data incompleteness. The performance of the method is illustrated for three problems of increasing complexity, ranging from pure Data Science to multiscale computational mechanics, showing its general scope due to its abstract formulation. The concepts, theoretical background and fundamental applications of this chapter were published in Ayensa-Jiménez et al. (2019b) while the multiscale application can be found in Mora-Macías et al. (2020), although I have not included all the work related with the experimental tests and data obtention, that is part of this latter work but has not been carried out by the author of this thesis.

In Chapter 5, I slightly change the approach and move to the concept of physically-guided neural networks with internal variables (PGNNIV), which is one of the major contributions of this thesis. The DD methods presented in the two previous chapters, and many others described in the scientific literature, rely on the fundamental assumption that the variables of the state space are always measurable and therefore may be sampled by any experimental procedure. This is a very strong, and what is more serious, unrealistic assumption, as many of the variables used for describing physical problems are mathematical constructs defined by the modeller (the so-called internal state variables). In this chapter, I relax this condition and, inspired by the ideas and tools from the field of neural networks, which is nowadays the main hot spot of AI, I build a framework to include the internal variables in PIDS methods in a consistent way. This approach does not require a direct measurement of any internal variable since their definition depends only on how they are mathematically related to the rest of the variables of the problem. A first application of PGNNIV is shown

for a problem extracted from hydraulics: this problem is simple, illustrative and has a solid justification, as hydraulics is one paradigmatic area where a huge number of constitutive equations for describing water head loss have been proposed (representing turbulence and viscous dissipation) and the replacement of this certainly empirical and context-dependent knowledge by data is clearly useful. This application is also used for demonstrating how PGNNIV outperforms conventional artificial neural network (ANN) approaches. Then, I adapt the ideas to PDEs and the study of continuum physics. I build a framework to demonstrate how the method may be adapted to any physical problem, including biology and ecology, engineering, and even to non-physical problems, such as sociology or finance. The performance of the method is shown in a simple but illustrative problem of heat transfer, even if it might be applied to any PDE. The content of this chapter was published as two papers. In Ayensa-Jiménez et al. (2021a), we presented the foundations of the methods, their application to the hydraulics discrete problem and we evaluated the performance when compared to standard ANN approaches. In Ayensa-Jiménez et al. (2020a), we presented the general framework for continuum mechanics and the diffusion/heat transfer application.

1.3.2 Application to the study of Glioblastoma evolution

The second part of the thesis is dedicated to the study of GBM from a mathematical point of view. As the study of GBM in microfluidic devices is recent, to establish general models and frameworks for incorporating both scientific knowledge and data flow to computations may be too ambitious at once. We need, first, to develop a framework, although using conventional approaches, in which to incorporate later the data flux according to the methods developed in the first part of this work. Indeed, it is necessary to evaluate what are the really known phenomena, generating also a scientific consensus, and which are not. Additionally, we need to explore the biological roots of the different hypotheses of the model and to evaluate their impact (including uncertainty considerations) on the predictions.

This is why, in Chapter 6, I formulate a general computational framework for analysing cell cultures in microfluidic devices. This framework includes the general mathematical model, in terms of a system of PDEs, and two software solutions, adapted to different situations: one-dimensional problems and multidimensional problems where the different time scales are of the same order. The framework was published as a book chapter and can be consulted in Ayensa-Jiménez et al. (2019a). The general framework is then particularised for GBM. The biological evidences/hypotheses about this tumour are translated into some particular nonlinear corrections. The parametric model is formulated in terms of 11 model parameters and one parameter related to experimental operation. As the model is highly multiparametric, a comprehensive literature review is performed to narrow the searching space and reduce the computational cost, before proceeding with a standard parametric fitting, using data from real experiments in microfluidic devices under hypoxic conditions. The GBM evolution model and the results of the parametric fitting have been published in Ayensa-Jiménez et al. (2020b), together with a detailed description of the experiments performed in the microfluidic devices, that were not carried out by the author of the thesis. Once the parametric model has been settled

down, a sensitivity analysis is performed. This analysis was carried out under a probabilistic approach suited for multiparametric models where they might be strong and nonlinear hidden correlations between the different parameters. The concept of *copula* proves to be appropriate for analysing these correlations, particularly the one relative to the *go-or-grow* associated parameters and for the design of experiments (DoE) using information theory. This discussion about *copulae* utility was published in Ayensa-Jiménez et al. (2021b).

In response to the shortcomings of the parametric approach, Chapter 7 is the direct application of the methods and procedures of Part I to our problem of interest, that is the GBM progression in microfluidic devices. This is motivated by the sensitivity analysis of the previous chapter, which demonstrates the impact of accurate characterisation of the *go-or-grow* metabolic switch. In particular, I use the concept of PGNNIV to formulate a couple of predictive-explanatory networks based on (a slightly simplified version of) the parametric model presented in Chapter 6, except for the fact that, now, the *go-or-grow* behaviour is learned in a non-parametric fashion. The PGNNIV approach demonstrates to be able both to explain the hidden nature of the *go-or-grow* metabolic switch and to predict the evolution of the cell culture under different and variable external stimuli, obtaining, therefore, non-parametric digital twins of the cell cultures using PIDS techniques. This ability to unravel the hidden metabolic switch of different cell cultures, combined with the *in silico* test of the obtained digital twins under different drugs and therapies is a clear example where PIDS helps to take the first steps towards personalised medicine. The content of this chapter has resulted in a paper that is under review.

Finally, in Chapter 8, the model is analysed from the mathematical point of view for a general class of functions related to the *go-or-grow* switch. In previous chapters, the PDE model for GBM evolution is solved numerically. This may be very demanding from the computational point of view, specially for uncertainty quantifications, inverse problems and when evaluating the response to a certain drug or treatment for therapy optimisation. Thus, I present an analytic approach to the dynamic problem of the evolution of cell cultures under heterogeneous migration-proliferation balances, where many features associated with the tissue evolution, such as propagating waves, cell concentration close to the blood vessels or periodic solutions may be obtained by applying the method of characteristics to the system of PDEs. As explained, the analysis is formulated in terms of the switch mechanism driving the *go-or-grow* behaviour, so it may be applied to any non-parametric behaviour learned using a PIDS approach, provided that it is possible to measure the concentration of the chemotactic agent. The results are particularised for two special cases where there is no need of measuring the chemoattractant concentration: a point source of a generic chemoattractant and the GBM evolution for low cell concentrations. This work was carried out in collaboration with Prof. Eammon Gaffney at Wolfson Centre for Mathematical Biology of Oxford University. This chapter has resulted in a paper that is under preparation.

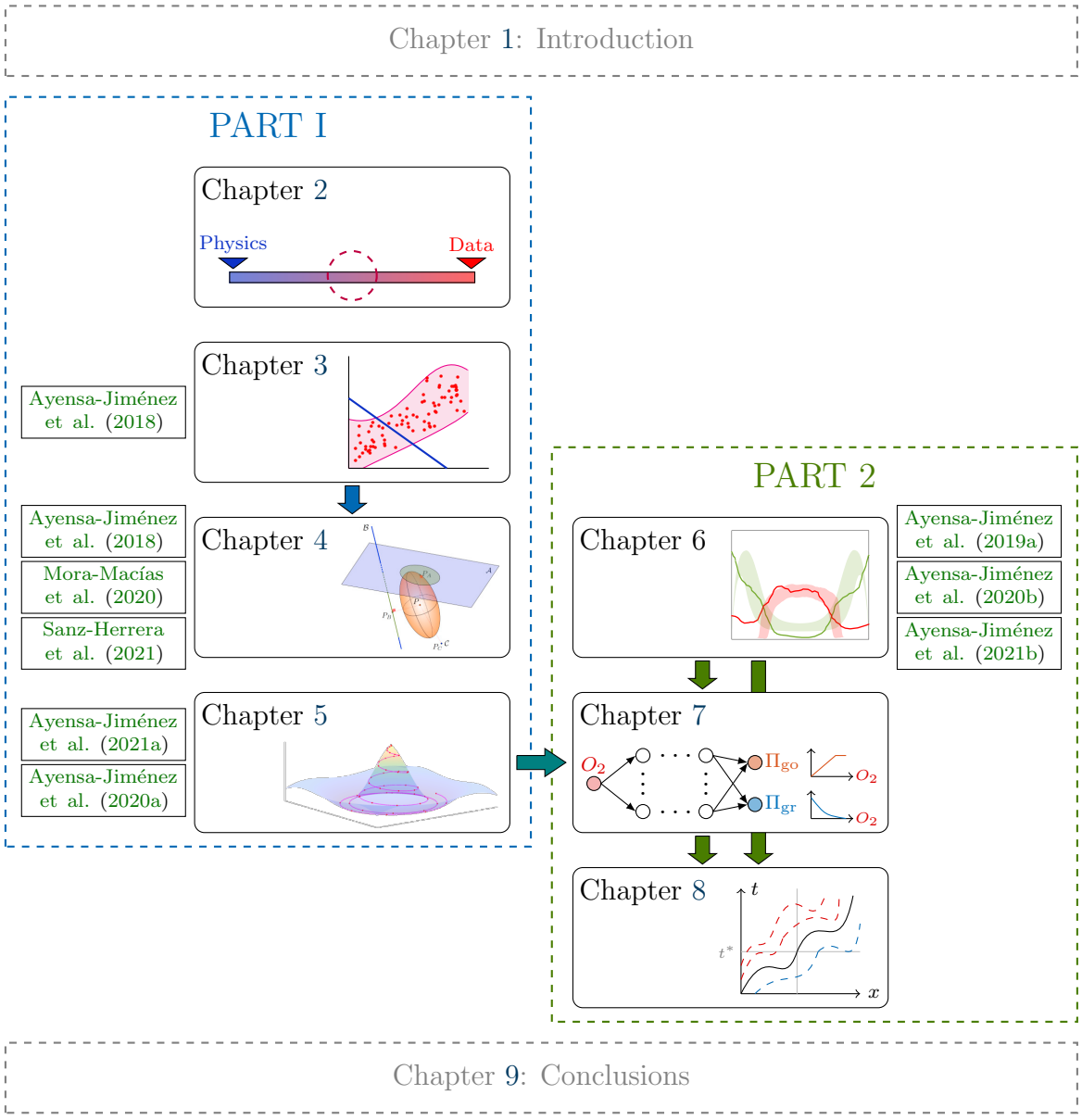


Figure 1.8: Structure summary of the thesis. Part I is dedicated to new PIDS methods and Part II to the understanding and simulation of GBM progression in microfluidic devices.

Part I

Physically-Informed Data Science methods

2. Data Science in predictive Physics

Contents

2.1 Introduction

Every day hundreds of millions of people take photos, make videos and send them with texts to the cloud. Billions of sensors and transducers get data in the ever-increasing tendency of prioritising prevention and control of complex systems to therapies, thus progressively translating control from humans to machines. Businesses of every kind search and collect data across the globe on consumer preferences, purchases and trends. Governments regularly collect all sorts of data, from census information to incident reports in police departments. According to the 2016 IDC directives presented in its yearly event in San Jose (US), this deluge of data is set to rise steeply from the estimated world total amount of 4,4 zettabytes of data in 2013 to 180 zettabytes by 2025, with one zettabyte equivalent to one trillion gigabytes. Moreover, with the advent of the internet of things (IoT), it is likely that these figures (Atzori et al., 2010) will be surpassed. This amazing amount of available information is pushing Science to a point where predicting results in a system or

population in a particular state and under specific inputs is becoming easier from the data available directly, than from the, until now useful but complex and sometimes cumbersome, physico-mathematical models. All this justifies the current prosperity of Data Science and disciplines such as Data Mining or Data Analytics (Manyika et al., 2011).

These methodologies permit to extract patterns and/or relevant information (we should better say knowledge) from available unstructured data, thus allowing us to make predictions with incredible accuracy (Khan, 2008; Larose, 2014), in what is nowadays known as machine learning (ML). Since the main ideas and concepts were introduced at the beginning of the century (Mitchell, 1997; Vapnik, 2013), an extensive literature may be found in this broad area. Some approaches are based on linear approximations, thus resulting in affine manifolds, although manifold identification has now grown to become a mature field where much more flexible strategies that overcome that initial linearity have been proposed (LJP and Van Den, 2007). A particular sub-discipline of ML is Manifold Learning (Lee and Verleysen, 2007; Yunquan and Yun, 2011) in which the particular aim is to get newer and richer hidden knowledge related to the underlying structure or, in mathematical terms, the dimensionality and local bases of the relevant working space. In the same direction, since Rosenblatt (1958) developed the perceptron, artificial neural networks (ANNs) are another pushful field that has nowadays achieved impressive results due to the modern hardware and software solutions, where new concepts as deep learning (DL) and dynamic networks are in continuous development and are able to identify more abstract features and solve more complex problems (Krenker et al., 2011; Silver et al., 2016; Suzuki, 2011). In a complementary and more modern direction, in dynamic data-driven application system (DDDAS) approaches (Darema, 2004), the idea is to continuously improve the accuracy of predictions by means of new available data, adding new explicit knowledge from the actual response to previous predictions.

These ML techniques try, therefore, to mimic the process of human knowledge acquisition and structuring and have become amenable only after remarkable advances in measurement and monitoring, data generation, management and storage and, finally, improvements in the performance of computer machines and algorithms, besides continuous contributions in their theoretical and algorithmic foundations.

Despite the great number of Data Science applications in areas such as marketing and e-commerce (Hill et al., 2006), social sciences (Aneshensel, 2013), or healthcare (Raghupathi and Raghupathi, 2014), there are other fields where very little has been done. An example are the disciplines where physical models and the corresponding mathematical and numerical simulation tools are well established, like Computational Physics, Computational Chemistry or Computational Engineering (all of them referred jointly as simulation-based engineering and sciences (SBESs)). In the last few years, however, a huge paradigm shift is taking place, where physical knowledge is being progressively surpassed by artificial intelligence (AI) techniques. Works as Brunton et al. (2016); Flaschel et al. (2021); Schmidt and Lipson (2009) are tremendously impacting the way we do Science. Very recently, a groundbreaking work by Udrescu and Tegmark (2020) has created the so-called Feynman AI[1], a

[1]The authors boast of calling this artificial intelligence Feynman AI, in honour of the mastery in all areas of Physics of the American genius.

Physics-inspired methodology for learning hidden structural relationships in data by means of recursive symbolic regression, able to unravel most of the physical fundamental equations derived by the most extraordinary minds of our history.

Nevertheless, data gathering in Physics is soaked by centuries of scientific knowledge and the associated human bias (Berry, 2011; Gould, 1981; Kitchin, 2013, 2014; Leonelli, 2012); so, a "blind" algorithm without any information on that bias may lead to wrong predictions. Also, scientific problems suffer many times from paucity of data while involving a large number of variables that interact in complex and non-stationary ways. Therefore, we can expect poor predictive capability of purely data-based approaches in problems far from the training set (Xue, 2019). Finally, a physically-based model is not only useful for making predictions, but it is also expected to help in gaining knowledge by the interpretation of its structure, parameters, and mathematical properties. In fact, physical interpretability is, in many cases, at least as important as predictive performance. Therefore, it is not strange that important efforts have been made towards "whitening" the "black-box" way of working of current ML predictive algorithms (Shwartz-Ziv and Tishby, 2017; Xu et al., 2019).

One possible solution to this shortcoming of data-only models is the seamless blending of their tremendous predictive power with the scientific consistency and interpretability of physically-based models. The term coined for this hybrid paradigm is physically-informed data science (PIDS) or physically-guided data science (PGDS) (when the focus is on the Physics inclusion) or data-driven simulation-based engineering and sciences (DDSBES) (when the focus is on the data inclusion). The aim is to integrate physical knowledge into Data Science models, that is, to constrain the prediction domain of the standard data model by physical constraints.

Unlike these approaches, based on the direct treatment of available data, PIDSs may incorporate, or not, in addition to data, some *a priori* characteristic physical knowledge of the analysed system. At this point, it is crucial to distinguish between two kinds of knowledge. On the one hand, physical general principles, such as conservation and thermodynamic laws that are universally accepted as able to describe the underlying universe structure, while, on the other hand, we find phenomenological models, such as macroscopic material constitutive relations, being the latter no more than intelligent simplifications of the real interactions at molecular level extracted from available experimental data. However, even if it was possible, at least in theory, to derive the real mechanistic constitutive relations also from first physical principles, the overwhelming number of degrees of freedom involved in the relevant spatial-temporal scales needed for real applications would make this possibility intractable (Xiao and Belytschko, 2004). So, the combination of physical principles and phenomenological models, under well-contrasted hypotheses, has demonstrated to be highly effective to solve complex realistic problems very accurately, while keeping them mathematically tractable. Fig. 2.1 summarises how the information is propagated from lower scales, where first principles hold (i.e., Schrödinger and fundamental interactions) to higher ones, where information is reduced at the expense of the apparition of new closure models. Once the model is established and validated (i.e., it proves to be sufficiently accurate for the purpose of the particular application in hands), the mathematical tools available will allow obtaining every aspect of interest: mathematical structure, relevant parameters and scales, predictions in

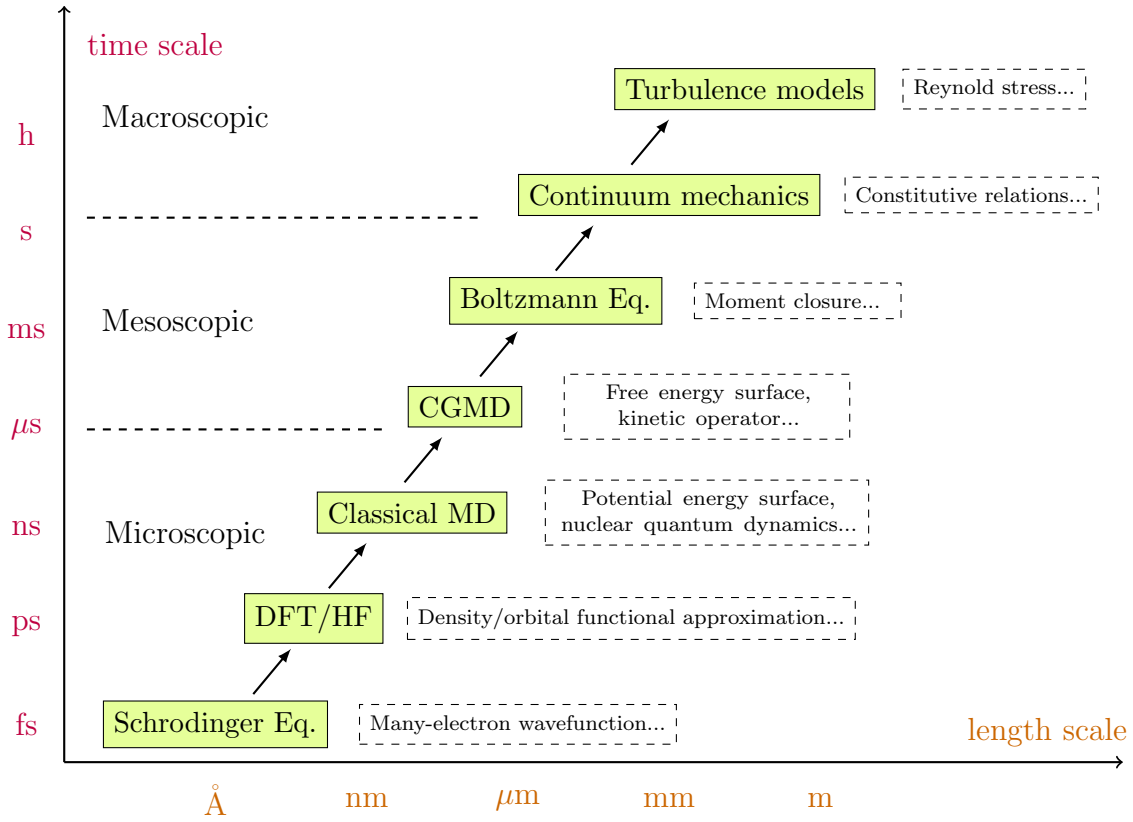


Figure 2.1: Characteristic scales and closure moments in computational Physics. CGMD: Coarse Grain Molecular Dynamics; MD: Molecular dynamics; DFT: Density functional theory approximation. HF: Hybrid functional approximation.

"what if" situations, etc. However, and except for very few examples in which first principles are the only ones used, the ability of any physico-mathematical model to represent reality relies in its phenomenological part and, at the end of the day, in the quality of the simplification hypotheses and of the available experimental data.

Keeping this big picture in mind, Science has historically progressed through the fruitful interaction between theory and experiments, or better, between hypotheses and data. Additionally, since the appearance of computers in the fifties, and accelerated in the nineties of the XXst century, simulation has been progressively recognised as the third pillar of the scientific method (Skuse, 2019). It is clear that Data Analytics techniques would be very useful in SBES at least for three reasons:

1. To obtain more interpretable and reliable models fleeing from the opaque paradigm of conventional AI to the transparent one of explainable artificial intelligence (XAI).
2. To extrapolate the phenomenological submodel, but now constrained with the mathematical expression of first principles. This framework has been coined as Machine Learning and Field Inversion and is brightly discussed in Parish and Duraisamy (2016)
3. To improve the performance of ML methods, which are highly data demanding and strongly depend on data quality.

In this mixed approach, the absence of physical constraints implies recovering the Data Science and ML framework, while total *a priori* parameterisation of experimental data recovers classical SBESs. DDSBESs may be considered then as an *a posteriori* manifold constructor that may be context-dependent. In the same way, PIDS may be considered as an *a priori* manifold restriction to the fulfilment of some physics (usually universal laws) that is also context-dependent. In other words: *let the data tell us which physical variables persist without forcing them* a priori*, except for the ones that we want to.*

In this chapter, our goal is to set a common framework for PIDS methods and to establish the language and notation for the rest of the dissertation. Also, we review the most relevant works in the literature that have played an important role in the origins and the present trends and research direction in PIDS. This review is done according to how, and to what extent, Physics is incorporated in Data Science methods. Finally, we close the chapter putting the different methods presented in perspective and discussing their weaknesses and strengths.

2.2 Concept of Physically-Informed Data Science

To describe a physical phenomenon, one must state the mathematical equations that control the evolution of a set of variables (position, momentum, temperature, entropy, ...) that completely determines the state of the system. That evolution depends upon a set of external stimuli, which are assumed to be known, and upon the state itself. In this context, we distinguish between two kinds of equations: universal physical principles (conservation laws and physical inequalities), and the internal state equations that compile the averaged behaviour of the system from its particular internal structure. The ability of any physical-mathematical model to accurately represent the reality is directly related to the quality of the simplification hypotheses that drive to those state equations and to the available experimental data required to identify the associated parameters.

In the next sections, we nail down a common framework for PIDS that we will use throughout the whole dissertation.

2.2.1 Mathematical formulation

We have seen that we can immediately identify two types of state variables in any averaged theory:

- Observable (measurable) independent variables such as the position $\boldsymbol{x}$, displacement $\boldsymbol{U}$, or temperature θ, that will be denoted in general as $\boldsymbol{u}$, which collect all possible *essential variables* of the problem, that is, the minimum set of independent, in general, time-dependent variables, whose history, together with the appropriate state equations, completely define the state evolution of the system. Each of these variables may have a different tensorial character: scalar, vector or higher-order tensor. Many times, the state functions or state equations are established in terms of time or spatial derivatives of these essential variables, such as velocity $\dot{\boldsymbol{x}}$, strain $\boldsymbol{\varepsilon}$, strain rate $\dot{\boldsymbol{\varepsilon}}$, temperature change rate $\dot{\theta}$, or temperature gradient $\boldsymbol{\nabla}\theta$. These are also easily measurable and will be termed here collectively as $\boldsymbol{\varepsilon}$.

- Internal non-observable (not directly measurable) variables, that are specific to the physical model considered. In general, these internal state variables depend upon the whole time-history of the system, and collect model-specific variables as well as internal changes in the microstructure, which may be reversible or irreversible. Examples are stresses, plastic or viscoplastic strains, damage, etc. They will be denoted in general as $\boldsymbol{v}$, and, again, each has a particular tensorial character.

In the same way, we stated already that there are two types of equations:

- Universal physical laws, valid for any problem and therefore independent of the particular system, such as conservation of mass, linear and angular momenta and energy.
- State equations that depend on the internal particular structure of the system, and, such that, after a set of assumptions, they average the behaviour of the microstructure into a set of internal variables and the associated parameters that are, therefore, macroscopic averaged properties of the system. They are the result of a loss of information that takes place when moving from a lower scale to an upper one. These equations are usually phenomenological so they have to be validated and the associated parameters determined from a sufficient set of experimental tests in different scenarios.

Consequently, any physical system may be formally defined in terms of a manifold $\mathcal{M}$ and a functional $\mathcal{F}$. $\mathcal{M}$ defines the functional space of possible solutions, making explicit, for example, the spatial dimension of the problem domain, the regularity degree requested to the essential variables, and the essential boundary conditions. $\mathcal{F}$ defines the evolution law of the system $\dot{\boldsymbol{u}} = \mathcal{F}(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{f})$ ($\mathcal{F}(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{f}) = \boldsymbol{0}$ in the non-transient case), where $\boldsymbol{v} \in \mathcal{N}$ denotes the internal state variables, each with its specific tensorial character, and $\boldsymbol{f}$ denotes the external stimuli. The set $(\boldsymbol{u}, \boldsymbol{v})$ conforms the set of *state variables*, while the functional space $\mathcal{M} \times \mathcal{E}$ is the so-called *phase space*. In Hamiltonian Mechanics, for instance, the set of state variables is composed only of essential (observable) variables: the position and momentum of each particle, while the stimuli are the forces acting on each particle. $\mathcal{F}$ play the role of the Hamilton equations. In thermodynamics, the state variables are macroscopic representations of the microscopic mechanical state, such as the temperature, pressure, volume, entropy or internal energy, and the stimulus is the heat. Actually, for discrete problems (for instance particle Hamiltonian mechanics), $\mathcal{F}$ is actually a vector function, $\boldsymbol{F}$, acting on a finite-dimensional space. For continuum problems, the state variables are tensor fields, $\boldsymbol{u}(\boldsymbol{x})$ and $\boldsymbol{v}(\boldsymbol{x})$, so the functional $\mathcal{F}$ operates in infinite-dimensional spaces.

Besides, the state equations relate the internal state variables with the essential ones (or their rates or gradients) via appropriate functionals $\mathcal{H}$, $\dot{\boldsymbol{v}} = \mathcal{H}(\boldsymbol{u}, \boldsymbol{v})$ (or $\boldsymbol{v} = \mathcal{H}(\boldsymbol{u})$ in the non-transient case). That is, $\boldsymbol{v}$ is expressed in terms of $\boldsymbol{u}, \boldsymbol{\nabla u}, \dot{\boldsymbol{u}}, \ldots$). Again, once discretised, they may be replaced by a function $\boldsymbol{H}$ between finite-dimensional spaces. These equations and the associated parameters are derived from reasonable assumptions and experimental tests, being, together with the discretisation stage,

the main bottleneck for defining the complete model of any physical system, while the evolution equation $\dot{\boldsymbol{u}} = \boldsymbol{F}(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{f})$ is always the same for a particular theory.

Commonly, the physical systems are defined via ordinary differential equations (ODEs) or partial differential equations (PDEs), depending on whether the system is spatially discrete or continuous. Also, when the spatially-discretised evolution equations are solved via numerical simulation, another discretisation step in time is required to transform the time-continuum problem into a discrete one. This includes the selection of a suitable time integrator (e.g. Euler, Multi-step, Runge-Kutta, among many others).

In the PIDS framework, the state variables of the system are identified with different ML elements. Some of them, the observable variables, are directly related to the input or output of a given procedure, while the rest (non-observable) cannot be identified with input or output variables[2]. Following the formalism above, we can write the rate evolution equation as:

$$\dot{\boldsymbol{u}} = \mathcal{F}(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{f}), \tag{2.1a}$$
$$\dot{\boldsymbol{v}} = \mathcal{H}(\boldsymbol{u}, \boldsymbol{v}), \tag{2.1b}$$

with $\mathcal{F}, \mathcal{H}$ the functionals defining the problem. After discretisation with any appropriate time integrator, we can rewrite:

$$\boldsymbol{u}^{t+\Delta t} = \mathcal{F}(\boldsymbol{u}^t, \boldsymbol{v}^t, \boldsymbol{f}^t), \tag{2.2a}$$
$$\boldsymbol{v}^{t+\Delta t} = \mathcal{H}(\boldsymbol{u}^t.\boldsymbol{v}^t). \tag{2.2b}$$

By identifying $\boldsymbol{u}^{t+1}$ and $\boldsymbol{v}^{t+1}$ with the state variables and $\boldsymbol{u}^t$ and $\boldsymbol{v}^t$, which are known at time t, with the external stimulus, it is possible to find a common framework for both transient and stationary problems using the general equations:

$$\mathcal{F}(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{f}) = 0, \tag{2.3a}$$
$$\mathcal{H}(\boldsymbol{u}, \boldsymbol{v}) = 0. \tag{2.3b}$$

Eqs. (2.3) are the mathematical representation of a system, in general, partial integro-differential equations to be solved by any of the many numerical methods available. As in any realistic problem the phenomena analysed are constrained to a specific region of the space, so Eq. (2.3) must be complemented with appropriate boundary conditions, that are of two kinds:

- **Essential boundary conditions:** They are associated with the knowledge about the essential variables $\boldsymbol{u}$ at some regions of interest.
- **Natural boundary conditions:** They are the expression of universal laws or principles at the domain boundaries.

In any case, identifying by $\boldsymbol{g}$ the data associated with boundary conditions, the boundary conditions may be formulated as:

$$\mathcal{G}(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{g}) = 0, \tag{2.3c}$$

[2] At least, without additional assumptions.

where $\mathcal{G}$ is a new functional.

Note that if there are no internal variables, i.e. the problem is stated only with measurable variables, Eqs. (2.3) write:

$$\mathcal{F}(\boldsymbol{u}, \boldsymbol{f}) = 0, \tag{2.4a}$$
$$\mathcal{G}(\boldsymbol{u}, \boldsymbol{g}) = 0. \tag{2.4b}$$

Also, if the state constitutive equations are assumed to be explicit, Eqs. (2.3) turn into:

$$\mathcal{F}(\boldsymbol{u}, \boldsymbol{v}(\boldsymbol{u}; \boldsymbol{\lambda}), \boldsymbol{f}) = 0, \tag{2.5a}$$
$$\mathcal{G}(\boldsymbol{u}, \boldsymbol{v}(\boldsymbol{u}; \boldsymbol{\lambda}), \boldsymbol{g}) = 0, \tag{2.5b}$$

where $\boldsymbol{\lambda}$ are model parameters, resulting from the substitution of universal laws by empirical equations, after the information loss.

The general case, that is the one that will be discussed throughout this dissertation is:

$$\mathcal{F}(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{f}) = 0, \tag{2.6a}$$
$$\mathcal{G}(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{g}) = 0, \tag{2.6b}$$
$$\mathcal{H}(\boldsymbol{u}, \boldsymbol{v}) = 0. \tag{2.6c}$$

2.2.2 A brief about the discretisation

Let us consider the Hilbert space of physically admissible solutions, $\mathcal{H}$, and a manifold $\mathcal{M} \subset \mathcal{H}$ of solutions dependent on a set of parameters denoted by $\boldsymbol{\mu} \in \mathcal{S}$, being $\emptyset \subseteq \mathcal{S} \subseteq \mathbb{R}^k$ a compact set. These parameters can be model parameters ($\boldsymbol{\lambda}$) or associated with boundary conditions (BC). The elements of $\mathcal{M}$ are denoted as $\boldsymbol{u}_{\boldsymbol{\mu}}$. Considering the solution map:

$$\phi : \mathcal{S} \to \mathcal{H}, \quad \boldsymbol{\mu} \mapsto \boldsymbol{u}_{\boldsymbol{\mu}}, \tag{2.7}$$

it is possible to write $\mathcal{M} = \phi(\mathcal{P})$. When the elements of the solution space are computed numerically, any available discretisation technique (Larsson and Thomee, 2009), such as finite differences (FD) (Langtangen, 1999), finite elements (FE) (Zienkiewicz et al., 1977), or other spectral techniques (Boyd, 2001), the spaces $\mathcal{M}$, $\mathcal{H}$ and the map ϕ are replaced by their discrete analogues, $\mathcal{M}_h$, $\mathcal{H}_h$ and the map ϕ_h, where h denotes some mesh-related parameter. Once the problem is discretised, the state variables and stimuli may be described in terms of the respective finite-dimensional vectors and the resulting problem is:

$$\boldsymbol{F}(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{f}) = 0, \tag{2.8a}$$
$$\boldsymbol{G}(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{g}) = 0, \tag{2.8b}$$
$$\boldsymbol{H}(\boldsymbol{u}, \boldsymbol{v}) = 0. \tag{2.8c}$$

2.2.3 Notation

When referring to the different PIDS, we are going to follow, whenever possible, the following rules:

- Capital calligraphic letters, such as $\mathcal{F}, \mathcal{G}, \mathcal{H}$, will be associated with functionals, sets and manifolds. An exception are linear manifolds (vector spaces), which will be denoted using capital letters.
- Capital bold letters are vector functions, such as $\boldsymbol{F}, \boldsymbol{G}, \boldsymbol{H}$, matrices or higher order tensors, such as $\boldsymbol{\Sigma}, \boldsymbol{M}, \boldsymbol{C}, \boldsymbol{\Upsilon}$ and random vectors, such as $\boldsymbol{X}$.
- Capital letters are manifold points, such as P, Q, and random variables, such as X.
- Lowercase bold letters are vector variables, such as $\boldsymbol{p}, \boldsymbol{x}, \boldsymbol{y}$, and vector fields, such as $\boldsymbol{u}, \boldsymbol{v}$.
- Lowercase regular letters are scalar variables, such as α, β, p and scalar fields, such as u, f.
- Sans-serif capital letters are identified with ANN components or other ML function representation, such as Y, H, and with spatial or time indexation, such as I, J.
- Typewriter style fonts, such as $\mathtt{T}, \mathtt{Y}$, are used to refer to TensorFlow tensor structures, that are multiarrays of different order and dimension.

2.2.4 The goal of Physically-Informed Data Science

The different PIDS strategies address Eqs. (2.6) or Eqs. (2.8) under different situations or hypotheses, using ML methods.

- The first possibility is to assume the whole physics as known. In that case, $\boldsymbol{v} = \mathcal{H}(\boldsymbol{u}; \boldsymbol{\lambda})$ and our aim is to solve a given physical problem, maybe for some different parameters $\boldsymbol{\mu} \in \mathcal{S}$. Therefore, the objective is to derive the relation $\boldsymbol{f}, \boldsymbol{g}, \boldsymbol{\mu} \to \boldsymbol{u}$, that we will call Y, using ML techniques, such that:

$$\boldsymbol{F}(\mathsf{Y}(\boldsymbol{f}, \boldsymbol{g}; \boldsymbol{\mu}), \boldsymbol{H}(\boldsymbol{u}, \boldsymbol{v}; \boldsymbol{\lambda}), \boldsymbol{f}) \equiv 0, \tag{2.9a}$$

$$\boldsymbol{G}(\mathsf{Y}(\boldsymbol{f}, \boldsymbol{g}; \boldsymbol{\mu}), \boldsymbol{H}(\boldsymbol{u}, \boldsymbol{v}; \boldsymbol{\lambda}), \boldsymbol{g}) \equiv 0. \tag{2.9b}$$

  The ML task is, therefore, to build the relationship between $\boldsymbol{x} = (\boldsymbol{f}, \boldsymbol{g})$ and $\boldsymbol{y} = \boldsymbol{u}$[3], and, as a byproduct if it is the case, to determine some physically relevant model parameters $\boldsymbol{\lambda}$. The final PIDS model will be encoded in two sets of parameters, one with a physical prescribed meaning, $\boldsymbol{\lambda}$, and another instrumental one, related to the ML approach selected (for instance the ANN weights and biases), $\boldsymbol{\Lambda}$. The capacity for accurately representing $\boldsymbol{Y}$ is called the **predictive power** of the model. The way in which Physics is treated depends on the method selected, so, in next section, we will discuss the most relevant approaches.

[3]Other approaches are possible, by reversing the role of the input and output, but here we expose only this one for clarity.

- The second possibility is to relax, in one way or another, the physical knowledge about the system. In that case, and depending on the approach, the method can incorporate new features to its predictive power. For instance, one could try to build from the data a model F satisfying:

$$\mathsf{F}(\boldsymbol{u}, \boldsymbol{H}(\boldsymbol{v};\boldsymbol{\lambda}), \boldsymbol{f}) \equiv 0, \tag{2.10a}$$
$$\boldsymbol{G}(\boldsymbol{u}, \boldsymbol{H}(\boldsymbol{v};\boldsymbol{\lambda}), \boldsymbol{g}) = 0. \tag{2.10b}$$
$$\tag{2.10c}$$

  In that case, we would discover the universal laws of the nature. Conversely, one could try to build from the data a model F satisfying:

$$\boldsymbol{F}(\boldsymbol{u}, \mathsf{H}(\boldsymbol{v};\boldsymbol{\lambda}), \boldsymbol{f}) \equiv 0, \tag{2.11a}$$
$$\boldsymbol{G}(\boldsymbol{u}, \mathsf{H}(\boldsymbol{v};\boldsymbol{\lambda}), \boldsymbol{g}) = 0. \tag{2.11b}$$
$$\tag{2.11c}$$

  In that case, we would discover the constitutive equations. This capacity for accurately representing F or H is called the **explanatory power** of the method. Of course, to learn at the same time both universal and constitutive equations is not possible, as this, as illustrated in Fig. 2.1, depends on the information level and selected framework. In other words, there is a certain trade-off between predictive and explanatory power.

- Finally, a third approach is to solve physical problems without knowing totally or partially one of the model components expressed in Eqs. (2.8). This is possible as this ignorance is replaced by data in one way or another, in what we know as DDSBES. Here, there essentially exist two possibilities:
  - The first one is to make the data speak by themselves, that is, to replace a model component by raw data. This is known in the literature as a pure data-driven (DD) approach.
  - The second is to dynamically build, enrich or correct the model by feeding it with data, in a dynamic fashion. This is known as a DDDAS.

Next, we illustrate these three approaches by discussing the most relevant works on the scientific literature. For clarifying the exposition, we group the two last approaches together, as both deal with how Physics is incorporated to the data.

2.3 Taxonomy of Physically-Informed Data Sciences

2.3.1 In terms of the Physics treatment

The different PIDS approaches that have been presented can be categorised based on how Physics is introduced in the computational scheme. Indeed, any ML method may be considered as the combination of three main ingredients:

1. **A ML model.** In pure Data Science approaches, the most common families of models are ANNs (Nielsen, 2015), decision trees (Rokach and Maimon, 2007),

support-vector machine (SVM) (Cortes and Vapnik, 1995), regression analysis (both parametric and non-parametric (Eubank, 1999)), Bayesian networks (Castillo et al., 2012) and genetic algorithms (Mitchell, 1998). The different PIDS approaches start from one of these techniques and introduce some modifications in the model to allow for physics inclusion.

2. **A data-set.** Whether we work with supervised, unsupervised or reinforcement learning methods, a sufficiently big and varied data-set is always required. The ML method will obtain knowledge from these data.
3. **A learning process.** The learning process is the process by which data is transformed into information, encoded in the model parameters or model functions of the ML model. A learning process may also be considered as the combination of several ingredients:
   (a) **A loss function.** It is the function that measures how well the model is able to explain the data. Any supervised or reinforcement learning method includes a loss function, as well as many of the unsupervised learning methods, which may be reframed as the minimisation of a function depending on the model and the data-set.
   (b) **An optimisation algorithm.** The mathematical algorithm by which the loss function is minimised.

Next, we classify the different PIDS methods based on how Physics is incorporated.

2.3.1.1 Physics is included in the data-set

The most straightforward inclusion of Physics in ML methods is via the data-set, that is, applying any ML technique to data that have been generated by means of any physically consistent process. It is clear that all data that surround us are generated following the physical rules governing the universe, but, since these rules (including the associated parameters) are not known, in general, we refer here to data that has been *in silico* generated using some known and prescribed physically-based mathematical equations, using any kind of mathematical procedure:

$$\boldsymbol{y} = \boldsymbol{Y}_{\text{physics}}(\boldsymbol{x}). \tag{2.12}$$

In Eq. (2.12), $\boldsymbol{x}$ is any controlled input variable, $\boldsymbol{y}$ is a piece of data generated, and $\boldsymbol{Y}_{\text{physics}}$ is a mathematical equation or process (that can be solved for instance, by computational simulation), describing some prescribed physics. Hence, the data-set is defined as follows, depending on whether we are in the supervised or unsupervised context:

- Supervised learning: The data-set includes both the input (independent variable) and the output (label or dependent variable):

$$\mathcal{D} = \{(\boldsymbol{x}, \boldsymbol{y}) \,|\, \boldsymbol{y} = \boldsymbol{Y}_{\text{physics}}(\boldsymbol{x}),\ \boldsymbol{x} \in \Omega\}, \tag{2.13}$$

  where Ω is some predefined sampling space.
- Unsupervised learning: During the training process, the inputs are known and classified in some way into different categories. After that, these categories are identified with the outputs.

A posteriori Model Order Reduction

One of the first PIDS tentatives, even though this particular name is much more recent and evokes other techniques, was reduced order models (ROMs) (Lucia et al., 2004). A ROM is a model that, due to its reduced number of degrees of freedom, reduces the computational cost of numerical simulations and may be used in real time applications or online evaluation flows. Considering the solution space $\mathcal{M}_h$ (that may depend on a parametric space $\mathcal{S}$), the idea is to look for a manifold $\mathcal{R}_h$, with $d = \dim(\mathcal{R}_h) \ll \dim(\mathcal{M}_h) = D$ such that, given $\tilde{\boldsymbol{u}}_{\boldsymbol{\mu}} \in \mathcal{R}_h$,

$$\|\tilde{\boldsymbol{u}}_{\boldsymbol{\mu}} - \boldsymbol{u}_{\boldsymbol{\mu}}\| < \epsilon, \quad \forall \boldsymbol{\mu} \in \mathcal{S}, \tag{2.14}$$

where ϵ is a small parameter and $\|\cdot\|$ denotes a suitable norm. This idea was initially introduced in fluid mechanics, due to the high difficulty of solving Navier-Stokes equations in a reasonable time (Berkooz et al., 1993), although it has been applied to a wide variety of areas since then. For instance, and inspired by classical statistical methods such as principal component analysis (PCA), proper orthogonal decomposition (POD) is one of the most notorious techniques to build approximations to physical solutions by means of a linear combination of orthogonal modes (Abdi and Williams, 2010; Jolliffe, 2002; Rathinam and Petzold, 2003):

$$\tilde{\boldsymbol{u}}_{\boldsymbol{\mu}} = \sum_{i=1}^{d} \alpha_i(\boldsymbol{\mu}) \tilde{\boldsymbol{u}}_i. \tag{2.15}$$

A mathematical framework using stochastic processes was developed by Karhunen (1947) and Loève (1978), in what is today known as the Karhunen-Loève decomposition (Graham and Kevrekidis, 1996; Park and Cho, 1996), and the method has been reframed using control theory for controlled dynamical systems (Rowley and Dawson, 2017; Willcox and Peraire, 2002). A more sophisticated technique is the reduced basis method (RBM) (Manzoni et al., 2012; Rozza et al., 2008), which builds a reduced basis using greedy algorithms, and is nowadays a very mature family of methods for analysing parametrised PDEs. In the same vein, similar methods as balanced truncation (Safonov and Chiang, 1989) and rational interpolation (Baur et al., 2011) were investigated, as well as spatially separated representations using geometrical mappings (Ghnatios et al., 2019a).

The same ideas have been extended to nonlinear geometric structures (Mignolet et al., 2013; Nguyen and Peraire, 2008). For instance, the empirical interpolation method (Chaturantabut and Sorensen, 2010; Negri et al., 2015) deals with non-affine parameter dependence. A complete survey on ROM techniques in dynamical systems can be found in Benner et al. (2015). The nonlinear learning capacity offered by ANNs has been largely exploited, particularly with autoencoders (Kramer, 1991) in computational mechanics (CM) contexts (Lee and Carlberg, 2020; Oishi and Yagawa, 2017). Also, other authors have proposed hybrid strategies combining ANNs and ROM, where the ROM is enriched dynamically using a mesh refinement, driven by an ANN that seeks to minimise a model enhancement function (Baiges et al., 2020), or using DL architectures for enhancing POD, as proposed by Fresca and Manzoni (2022).

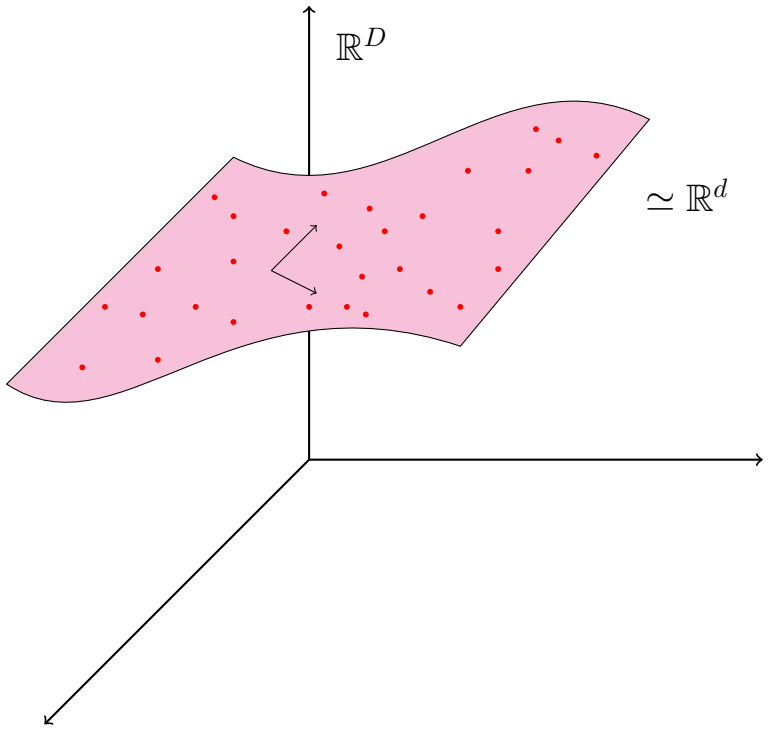


Figure 2.2: Manifold identification for Model Order Reduction. The red dots represent data in the ambient space ($\mathbb{R}^D$). The pink surface is the manifold of hidden latent variables, of dimension $d \ll D$, built using Manifold Learning.

In many occasions, the main goal is oriented to the discovery of the space of latent variables, that is, a simplified representation of the physical structure of the data. Indeed, with the explosion in the 2000s of Manifold Learning, a manifold identification as a reduced space has now grown to become a mature field where much more flexible strategies that overcome linearity have been proposed (LJP and Van Den, 2007; Van Der Maaten et al., 2009). In Fig. 2.2, the geometrical idea behind the method is illustrated. A more detailed description of the different Manifold Learning and ANN methods and their application in PIDS is exposed in Section 2.3.2.4.

Surrogate models (metamodels)

Even if this approach is very similar to ROM techniques, usually we are not really interested in the latent physical structure, but only in the encapsulation of a given $\boldsymbol{x} \rightarrow \boldsymbol{y}$ physical relation, for instance, the one relating the boundary conditions of a PDE (as input) with the solution field (as output) or a given quantity of interest (QoI), or the one relating the solution field (input) to some model parameters (output). This approach is interesting when the aim is to accelerate the computations for uncertainty quantification (UQ) purposes (Tripathy and Bilionis, 2018) or for solving inverse problems (Frangos et al., 2010). The different ML models built using this approach, bypassing conventional simulation, are called metamodels (Ghosh et al., 2018) or surrogate models (Cozad et al., 2014; Forrester and Keane, 2009), as they try to replace, for some convenient reason (usually due to a much lower computational cost in the online evaluation or because of their non-intrusive nature) standard simulation techniques. A very common application of ML and, in particular, of ANNs, is as

a ROM built from FE simulations (Mohammadzadeh and Lejeune, 2021; Phellan et al., 2021) to obtain real time or patient-specific tools.

ANNs have been intensively used in CM since the nineties (Ghaboussi, 2010; Yagawa and Okuda, 1996). The use of ANNs is actually very common for parameter identification or constitutive modelling in CM (Wang et al., 2021a). Some examples, out of the huge amount, are concrete modelling (Ghaboussi et al., 1991), soft tissues (Alekya et al., 2019; Cilla et al., 2018; Liu et al., 2019), hyperelasticity (Liang and Chandrashekhara, 2008; Shen et al., 2005), plasticity (Abueidda et al., 2021; Im et al., 2021), history (Ghavamian and Simone, 2019) and rate dependent materials (Al-Haik et al., 2003; Furukawa and Yagawa, 1998; Jung and Ghaboussi, 2006), granular materials (Ellis et al., 1995; Ghaboussi and Sidarta, 1998; Zhang et al., 2021), polymeric, polycrystalline (Ali et al., 2019; Mangal and Holm, 2018, 2019) and composite materials (Abueidda et al., 2019; Al-Haik et al., 2006; Liu et al., 2020). ANNs have also been used in frameworks such as fracture mechanics, for analysing microcracking, brittle fracture, and crack propagation (Asteris and Plevris, 2017; Gajewski and Sadowski, 2014; Goswami et al., 2020). Other authors have preferred to use Bayesian neural networks (BNNs) (MacKay, 1992). For instance, Geneva and Zabaras (2019) used BNNs for Reynolds-averaged Navier-Stokes (RANS) simulations (Geneva and Zabaras, 2019) and for unravelling the dynamics of a system's evolution (Kohler et al., 2015). Indeed, BNNs may be seen as an extension of hidden Markov models (HMMs) to more complex probabilistic graphical models (Zimmermann and von Hoessle, 2013).

With the advent of ML, and, in particular of ANNs, the compression techniques for features detection have been very prolific. One of the main ANN architectures are autoencoders (Kramer, 1991). An autoencoder with linear activation functions can be shown to be equivalent to the singular value decomposition (SVD), so autoencoders may be viewed as nonlinear generalisations of PCA. Deep autoencoders have been used as surrogate models for the analysis of stochastic partial differential equations (SPDEs) and UQ (Zhu and Zabaras, 2018), particularly in heterogeneous media (Mo et al., 2019; Tripathy and Bilionis, 2018). Autoencoders have also proved to be able to learn the dynamics of nonlinear systems (Gonzalez and Balajewicz, 2018; Milano and Koumoutsakos, 2002), in some cases using sparsity regularisation (Sondak and Protopapas, 2021). Recently, shallow masked autoencoders (Kim et al., 2021) and convolutional autoencoders (Nikolopoulos et al., 2021) have also been used for the surrogate modelling of parametrised time-dependent PDEs.

When the path dependence in evolution processes becomes more important, recurrent neural networks (RNNs) have shown good performance. RNNs have been used for plastic (Borkowski et al., 2022; Im et al., 2021; Mozaffar et al., 2019) and elasto-plastic heterogeneous materials using multiscale approaches (Wu et al., 2020a), for poroplasticity (Wang and Sun, 2018) and for phase-transition theory (de Oca Zapiain et al., 2021).

Recently, DL has impacted CM as a powerful tool to exploit the information contained spatially in the material data-sets (Pfeiffer et al., 2019; Settgast et al., 2020; Teichert et al., 2019; Unger and Könke, 2009), in a similar manner as it is done in medical image (Litjens et al., 2017; Razzak et al., 2018; Shen et al., 2017). Similar approaches

have been proposed for biology and biomedical engineering (Pérez-Aliacar et al., 2021).

The importance of the geometry and its discretisation has also been investigated in relation to ANNs. Jiang et al. (2019) use Spherical ANNs for unstructured grids. A more robust concept is graph convolutional neural networks (GCNNs), very suitable for unstructured meshes and time frames. Since the first ideas developed by Gori et al. (2005) and recovered by Sukhbaatar et al. (2016) in a modern form, GCNNs (Cao et al., 2015; Kipf and Welling, 2016) have arisen as an alternative to convolutional neural networks (CNNs) for non-Euclidean domains, and have demonstrated to be very performant for analysing complex dynamical systems (Sanchez-Gonzalez et al., 2018). These ideas have been generalised in what is known today as geometric deep learning (GDL) (Bronstein et al., 2017), that is, the family of DL methods that conserve some features under the action of the elements of a certain group, something that is closely related to Physics via the Noether's Theorem. In particular, Kutz (2017); Ling et al. (2016) defined Tensor Based Neural Networks, a class of ANNs with embedded rotational invariance, to face RANS turbulence modelling, particularly for anisotropic situations. A similar approach was followed by Vlassis et al. (2020) for solid mechanics.

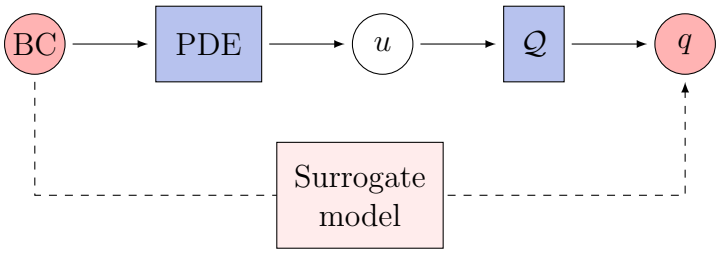
Finally, ML solutions for building surrogate models or response surfaces have been also applied in multiscale mechanics (Alber et al., 2019; Rocha et al., 2021; Wirtz et al., 2015), with the aim of accelerating the computations. Some examples are elasto-plastic materials (Wu et al., 2020a), poroelasticity (Wang and Sun, 2018), soft materials (Jackson et al., 2019), or composites (Yan et al., 2020). A closely related approach is the use of ANNs to design new materials taking as input microscopic structural parameters (Bessa et al., 2017; Chen and Gu, 2020; Le et al., 2015).

Modern approach to Gaussian Processes as surrogates

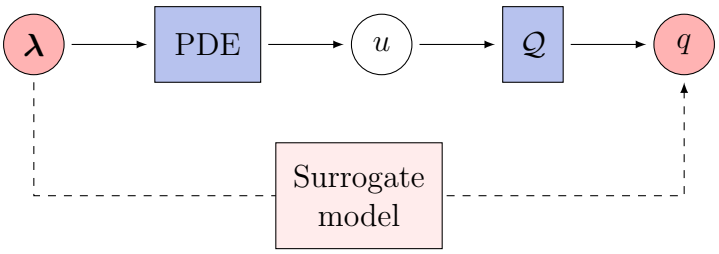
Among Bayesian techniques, and due to their flexible nature, Gaussian Processes (GP) (MacKay et al., 1998; Williams and Rasmussen, 2006) have been very popular as surrogates for reproducing complex models. A GP[4] may be seen as a BNN with a single hidden layer with infinite units (Neal, 2012). Predicting the output of expensive simulations using GP was explored by Kennedy and O'Hagan (2000) many years ago (Kennedy and O'Hagan, 2000; Santner et al., 2003). It has been widely used within science and technology (Gramacy, 2020; Pourmohamad, 2021). As an example, multi-output GP have been used to represent material properties (Bilionis et al., 2013), and the spatial structure (Atkinson and Zabaras, 2019) of some physical relations. Here we review the most modern approaches in the context of PIDS.

Inspired by the use of radial basis function (RBF) for approximating PDEs (Fasshauer and Ye, 2013; Franke and Schaback, 1998; Owhadi, 2015) (see next sections), Raissi et al. (2017a,b) proposed to use GP to learn both the solution and the hidden parameters of a PDE. From the perspective of Latent Variable Models (Everett, 2013), that is, using Gaussian process latent variable models (GPLVMs) (Lawrence and Hyvärinen, 2005), M. Raissi and co-workers extended the work of Alvarez et al.

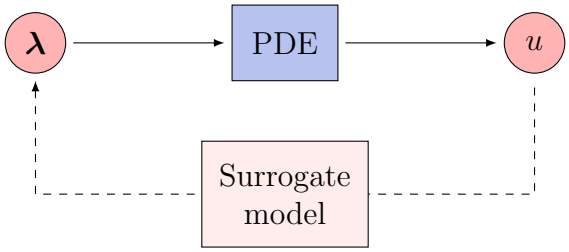
[4]Regression using GP is also known as Kriging.



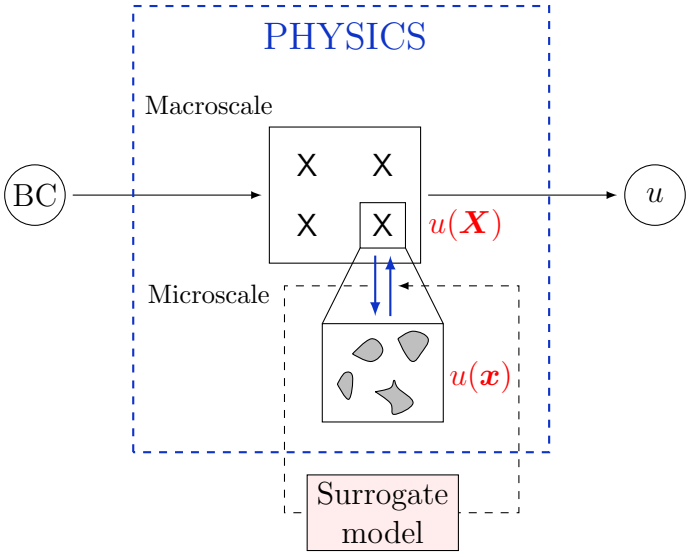
(a) QoI estimation and UQ (boundary conditions).



(b) QoI estimation and UQ (model parameters).



(c) Parameter identification.



(d) Multiscale constitutive modelling.

Figure 2.3: Surrogate models in SBESs. Physics is included in the data-set nature and is exogenous to the ML technique.

(2013) for estimating the driving term of a dynamic equation to a linear PDE having the form:

$$\mathcal{D}_{\boldsymbol{\lambda}}(u) = f, \tag{2.16}$$

where $\mathcal{D}_{\boldsymbol{\lambda}}$ is a certain differential operator depending on some model parameters $\boldsymbol{\lambda}$. They assume that the solution field is a GP:

$$u \sim \mathcal{GP}\left(0, k_{uu}(x, x'; \boldsymbol{\theta})\right), \tag{2.17}$$

where k_{uu} is the kernel function defined in terms of some hyperparameters $\boldsymbol{\theta}$. By linearity, f will be also a GP and the two kernels will be related by (Graepel, 2003; Särkkä, 2011):

$$k_{ff}(x, x'; \boldsymbol{\theta}, \boldsymbol{\lambda}) = \left(\mathcal{D}_{\boldsymbol{\lambda}} \circ \mathcal{D}'_{\boldsymbol{\lambda}}\right)\left(k_{uu}(x, x'; \boldsymbol{\theta})\right). \tag{2.18a}$$

$$k_{uf}(x, x'; \boldsymbol{\theta}, \boldsymbol{\lambda}) = \mathcal{D}'_{\boldsymbol{\lambda}}(k_{uu}(x, x'; \boldsymbol{\theta})). \tag{2.18b}$$

$$k_{fu}(x, x'; \boldsymbol{\theta}, \boldsymbol{\lambda}) = \mathcal{D}_{\boldsymbol{\lambda}}(k_{uu}(x, x'; \boldsymbol{\theta})). \tag{2.18c}$$

They demonstrate that it is possible to recover the solution of some integro-differential equations (Raissi et al., 2017a) and to accurately estimate the model parameters of fractional differential equations (Gulian et al., 2019; Raissi et al., 2017b). Their work has been extended to nonlinear time-dependent PDEs (Raissi and Karniadakis, 2018; Raissi et al., 2018b) using multi-step or Runge-Kutta integrators (Lambert, 1991) and substituting some terms by posterior mean estimations. Among the applications, we can cite the Laplace equation (Albert and Rath, 2020), heat equation (Raissi et al., 2018b), advection-diffusion-reaction systems (Raissi et al., 2017b), Burguer's and wave equation (Raissi et al., 2018b) or geosciences (Karimpouli and Tahmasebi, 2020). The method has also been adapted to include explicit constraints (Gulian et al., 2022).

2.3.1.2 Physics is included in the machine learning model

A priori Model Order Reduction

In *a priori* ROM methods, the reduced model is built without data, so no results or observations of the system are necessary to construct the ROM approximation to the unknown field $\boldsymbol{u}_{\boldsymbol{\mu}}$. In other words, Physics is introduced during the ROM construction.

One of the most prominent *a priori* methods is proper generalized decomposition (PGD) (Chinesta et al., 2010b, 2011; Ghnatios et al., 2012). PGD is formulated in two stages: an offline step where the computation of the ROM approximation is performed; and an online phase where the parametric solution is evaluated, under real-time constraints (Cueto et al., 2016). Mathematically, PGD assumes a separate representation[5] of the unknown solution field:

$$\boldsymbol{u}_{\boldsymbol{\mu}} \simeq \sum_{i=1}^{d} \alpha_i(\boldsymbol{\mu}) \boldsymbol{u}_i. \tag{2.19}$$

[5]In this example, we have considered the separability for the parametric dependence, although the separability may be considered for every single parameter or coordinate, something that will be more or less successful for approximating the solution depending on the geometry and nature of $\mathcal{M}_h$.

For determining the different modes and coefficients, the separate expression of the solution is introduced in the problem's physical equation. For instance, if the physics is formulated by means of the weak form of a PDE:

$$[A_{\mathcal{D}}(u)](v) = [B(u)](v), \quad \forall v \in \mathcal{H}_0, \tag{2.20}$$

where $A_{\mathcal{D}}(u)$ is a nonlinear functional associated with a differential operator $\mathcal{D}$ and $B(u)$ another functional, the separation modes are computed by solving the problem $[A_{\mathcal{D}}(\tilde{u})](v) = [B(\tilde{u})](v)$, where $\tilde{u}$ is the separate representation of Eq. (2.19). For linear problems, $[A_{\mathcal{D}}(u)](v) = a(u, v)$ and $[B(u)](v) = (u, v)$, where $a(\cdot, \cdot)$ is a bilinear form and $(\cdot, \cdot)$ denotes the scalar product at the Hilbert space. The precise form of the functional, ROM approximation is determined by means of a greedy algorithm (Chinesta et al., 2011). Within each sum of the separate representation, each "mode" is determined by solving the resulting problem obtained when we introduce the solution $\tilde{u}$. Different methods can be used, being alternating direction strategy the most common one (Chinesta et al., 2011). The ROM construction stops under a certain criterion based on error estimation (Alfaro et al., 2015; Ammar et al., 2010; de Almeida, 2013; Ladeveze and Chamoin, 2011). PGD has been widely used in many multiscale (Chinesta et al., 2010a) and multiphysics (Néron and Ladevèze, 2010) problems and is nowadays a mature field for solving physical problems in real time (Chinesta et al., 2013b; Niroomandi et al., 2013) and for design, optimisation, control and UQ (Chinesta et al., 2013c). Also, optimised sampling strategies, such as Sparse Subspace Learning (Borzacchiello et al., 2019) have been considered to circumvent the need of affinity in the separate representation, leveraging the Galerkin projection.

In addition to PGD, there are other *a priori* ROM methods. Ryckelynck et al. (2005, 2006) proposed the *a priori* Hyper-Reduction method, which is an incremental Karhunen-Loève decomposition built from the system equations. Similar approaches deal with nonlinear problems (Allery et al., 2011; Naets et al., 2019).

Structure preserving machine learning tools

A first natural approach is to define the ANN according to the process we want to learn. In that direction, some authors suggest using hybrid linear-nonlinear networks (Haesloop and Holt, 1990) and hierarchical neural networks (Mavrovouniotis and Chang, 1992), incorporating the structure of the relations we want to learn (see Fig. 2.4a). However, these approaches are based on empirical bias when assuming a given model structure and parametric dependence. The inductive bias given by parametric state modelling is smartly exploited by Psichogios and Ungar (1992), in a first attempt to open the "black-box" of ANNs by defining hybrid ANNs, including the universal principles in a network component, leaving another for the model discovery. In this work, a similar approach to Kalman filters (Kalman, 1960) is adopted, and the equation:

$$\frac{d\boldsymbol{u}}{dt} = \boldsymbol{F}(\boldsymbol{u}, \boldsymbol{f}), \tag{2.21}$$

where $\boldsymbol{u}$ is the state variable and $\boldsymbol{f}$ the control variable, is split in:

$$\frac{d\boldsymbol{u}}{dt} = \boldsymbol{F}(\boldsymbol{u}, \boldsymbol{f}, \boldsymbol{p}), \tag{2.22a}$$

$$\boldsymbol{p} = \boldsymbol{H}(\boldsymbol{u}, \boldsymbol{f}), \tag{2.22b}$$

so that $\boldsymbol{F}$ represents the universal physical evolution and $\boldsymbol{p} = \boldsymbol{H}(\boldsymbol{u}, \boldsymbol{f})$ a model state equation that is replaced by a network that learns the relationship $(\boldsymbol{u}, \boldsymbol{f}) \rightarrow \boldsymbol{p}$. Actually, the hybrid network represented in Fig. 2.4b learns the discrete relationship $(\boldsymbol{u}_k, \boldsymbol{f}_k) \rightarrow \boldsymbol{u}_{k+1}$. This approach was applied to more complex processes accounting for more flexible components (Thompson and Kramer, 1994; Wu et al., 2020b) and for parameter identification (Dua, 2011; González-García et al., 1998).

More recently, specific ANNs have arisen with the purpose of preserving the structure of some specific problems. In that sense, Xu et al. (2021) use the so-called symmetric positive definite neural networks (SPD-NN), specially prepared for learning constitutive relationships in elliptic PDEs problems. They illustrate the method for the hyperelastic, elastoplastic and multiscale problems in CM, where the goal is to unravel the constitutive relation $\boldsymbol{\varepsilon} \rightarrow \boldsymbol{\sigma}$. Vlassis and Sun (2021) suggested to modify the training process to include the higher derivatives of the strain energy function Ψ (Sobolev training) to ensure thermodynamics in elasto-plasticity problems, formulated in terms of a Hamilton-Jacobi evolution equation. Another specific network architecture is the so-called Thermodynamic-Aware neural network (Masi et al., 2021), where the stress increment $\Delta\sigma$ is computed from the state variables at the previous step (including strain, stress, temperature and other internal variables state, as well as strain increment), using the dissipation function and first and second laws of thermodynamics (local energy conservation and Clausius-Duhem inequality). Also, Tensor Basis Neural Networks (Fang et al., 2020; Ling et al., 2016; Milani et al., 2021) have been designed for facing rotational invariance in the turbulence closure models of Fluid Mechanics, in the same vein as SO(3)-informed neural networks for elasto-plastic anisotropic responses (Heider et al., 2020).

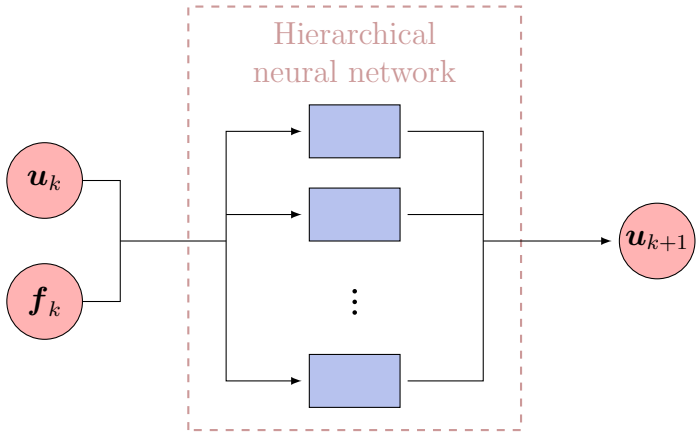
A new generation of structure preserving ML tools has recently emerged, relaxing the structure of the different operators for adapting them to arbitrary systems (satisfying the laws of thermodynamics), incorporating the general equation for the non-equilibrium reversible-irreversible coupling (GENERIC) formalism (Grmela, 2010). The GENERIC formalism establishes the equations of the dynamics of a system under thermodynamic reversible and irreversible conditions (also called metripletic structure), expressed by the evolution of energy E and entropy S, respectively (Grmela and Öttinger, 1997; Öttinger, 2005). The evolution equations for such a system are:

$$\dot{\boldsymbol{u}} = \boldsymbol{L}(\boldsymbol{u})\frac{\partial E}{\partial \boldsymbol{u}} + \boldsymbol{M}(\boldsymbol{u})\frac{\partial S}{\partial \boldsymbol{u}}, \quad \boldsymbol{u}(0) = \boldsymbol{u}_0. \tag{2.23}$$

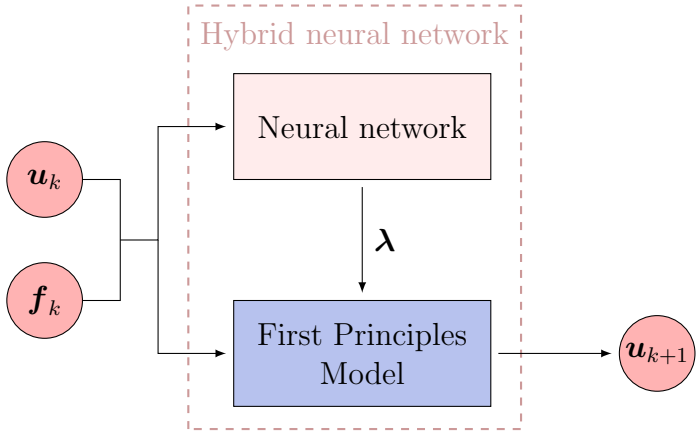
The term $\boldsymbol{L}$ is the so-called Poisson operator matrix and is responsible for the reversible part of the evolution system (according to Hamiltonian dynamics). $E = E(\boldsymbol{u})$ is the energy of the system, as a function of its state $\boldsymbol{u}$. In turn, $\boldsymbol{M}$ represents the dissipation matrix, responsible for the system irreversibility. $S = S(\boldsymbol{u})$ is the entropy of the system, that is also dependent on the state variables $\boldsymbol{u}$ (González et al., 2019). Eq. (2.23) must be completed with the complementary degeneracy conditions:

$$\boldsymbol{L} \cdot \frac{\partial S}{\partial \boldsymbol{u}} = \boldsymbol{0}, \tag{2.24a}$$

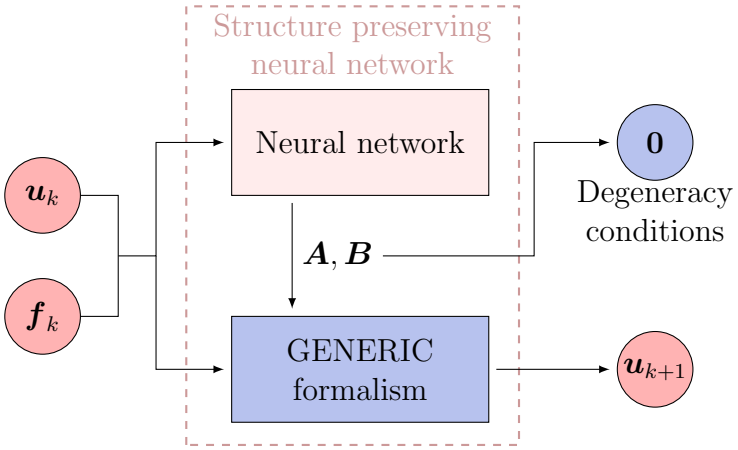
$$\boldsymbol{M} \cdot \frac{\partial E}{\partial \boldsymbol{u}} = \boldsymbol{0}. \tag{2.24b}$$



(a) Hierarchical ANN



(b) Hybrid ANN.



(c) Structure Preserving ANN.

Figure 2.4: Different families of ANNs preserving the physical structure. The network architecture, as well as the different nonlinear activation functions, are defined cleverly, according to the problem Physics.

If we choose $\boldsymbol{L}$ to be skew-symmetric and $\boldsymbol{M}$ symmetric and positive semi-definite, one ensures using the degeneracy conditions that $\dot{E} = 0$ and $\dot{S} \geq 0$, so the first and second principles of thermodynamics are satisfied. These ideas were adapted by Romero (2009, 2010a,b) to discrete numerical algorithms sharing the same symmetry properties. A first approach proposed by Ghnatios et al. (2019b); González et al. (2019) is to directly estimate $\boldsymbol{L}$ and $\boldsymbol{M}$ as well as the discrete versions of $\frac{\partial E}{\partial \boldsymbol{u}}$ and $\frac{\partial S}{\partial \boldsymbol{u}}$, that are expressed as $\boldsymbol{Au}$ and $\boldsymbol{Bu}$ respectively. The combination of this approach with ROM techniques has demonstrated to be very effective for analysing problems with a lot of degrees of freedom (Moya et al., 2019) that require real time evaluation, adapted for real scenarios of mixed and augmented reality (Moya et al., 2020a) and for dealing with noisy data-sets from biomechanics (González et al., 2020). Also, very recently, González et al. (2021) have adapted the methodology to incorporate non-observable (hidden) state variables, using kernel principal component analysis (kPCA) for unveiling the intrinsic variables of the dynamical evolution.

The GENERIC-adapted version of an ANN is called structure preserving neural network (SPNN) and was developed by Hernández et al. (2021). In a SPNN, a neural network block (for instance, a multilayer perceptron (MLP)) is used for estimating $\boldsymbol{A} = \boldsymbol{A}(\boldsymbol{u})$ and $\boldsymbol{B} = \boldsymbol{B}(\boldsymbol{u})$ as a function of the state variables $\boldsymbol{u}$. The degeneracy conditions are ensured by adding a regularisation term to the loss function of the ANN, as illustrated in Fig. 2.4c. The work has been adapted to large-scale discretised problems with a lot of degrees of freedom (Hernandez et al., 2021). In that case, the SPNN acts at a reduced space of latent variables that is built using sparse autoencoders.

Methods and architectures for specific PDEs

Another approach to use ML techniques in Physics is to design ML methods adapted to the structure of the PDE or the stochastic process involved. Some very preliminar ideas using RBFs were exposed by Mai-Duy and Tran-Cong (2001) (available data) and Jianyu et al. (2003); Mai-Duy and Tran-Cong (2002) (non available data) for elliptic PDEs that may be seen as some kind of filtering processes. Indeed, the RBF approach seems to be suitable for a meshless approach to the problem (Fasshauer and Ye, 2013; Franke and Schaback, 1998) (particularly for irregular domains (Mai-Duy and Tran-Cong, 2007)), including finite volume method (FVM) (Kansa et al., 2004), high-order ODEs (Mai-Duy, 2005), integro-differential equations (Golbabai and Seifollahi, 2007) and nonlinear integral equations Golbabai et al. (2009). Another direction was adopted using MLPs for specific classes of PDEs by Aarts and Van Der Veer (2001); He et al. (2000). Also, some works describe iterative procedures for enriching the network architecture (Shirvany et al., 2008; Tsoulos et al., 2009).

More recently, some authors have exploited the relation between PDEs and stochastic processes for exploiting different ML approaches. Indeed, the link between stochastic processes and PDEs is provided by Itô calculus (Durrett, 2018; Klebaner, 2012). Suppose a stochastic process defined with the SPDE:

$$dX_t = \mu(t, X_t)dt + \sigma(t, X_t)dW_t, \tag{2.25}$$

where μ and σ are regular functions and W_t denotes a Wiener process. Then, the probability density function (PDF) of the random process X_t, f satisfies the following Fokker-Planck equation:

$$\frac{\partial f}{\partial t} = -\frac{\partial}{\partial x}\left(\mu(x,t)\right) + \frac{1}{2}\frac{\partial^2}{\partial x^2}\left(\sigma^2(t,x)f\right). \tag{2.26}$$

This fundamental result, and some extensions to nonlinear equations using the Feynman-Kac formula (Bertini and Cancrini, 1995), were exploited by Weinan et al. (2017) in its seminal work relating backward stochastic differential equations (BSDEs) and parabolic PDEs. For a fairly general class of nonlinear parabolic PDEs, the equation is formulated as a BSDE using the Feynman-Kac formula, (Pardoux and Peng, 1990), and next interpreted as a stochastic control problem, with the gradient of the solution being the policy function. Finally, this stochastic control problem is interpreted as a reinforcement learning problem, where the policy function, that is, the gradient, is approximated by means of a DL network. This approach demonstrated to be particularly suitable for high dimensional problems, affected by the curse of dimensionality, such as the Black-Scholes model, Allen-Cahn equation and Hamilton-Jacobi-Bellman equation (Han et al., 2018). Beck et al. (2019) extended the approach to fully nonlinear parabolic PDEs and Chan-Wai-Nam et al. (2019) investigated different network architectures. Besides, some authors developed related schemes that approximate also the solution using DL (Huré et al., 2019) and improved the algorithm by including an asymptotic expansion as a prior estimation of the solution, accelerating the speed of convergence of the ANN (Fujii et al., 2019). Similar approaches have been investigated for the Fokker-Planck and Langevin equations (see, for instance, Khoo et al. (2019); Vanden-Eijnden et al. (2010), among a vast literature).

Parabolic PDEs are not the only ones for which specific methods for overcoming the curse of dimensionality using DL have been developed. Due to its structure, the residue network (ResNet) may be seen as the basic building block for approximating nonlinear ODEs, using multistep integrators (Qin et al., 2019). Due to its relevance in classical and quantum mechanics and optics (Courant and Hilbert, 2008; Goldstein, 2011), imaging science (Darbon, 2015), game theory (Evans and Souganidis, 1984) and optimal control (Bardi et al., 1997), the Hamilton-Jacobi equation has attracted much attention from researchers. In that sense, Darbon et al. (2020) have developed a method for obtaining viscosity (weak) solutions for the Hamilton-Jacobi equation. They propose an ANN architecture that exactly represents viscosity solutions of the Hamilton-Jacobi equation. A specific ANN architecture has also been proposed for solving the Schrödinger equation for many-electron systems (Han et al., 2019) and the many-body problem (Carleo and Troyer, 2017), this latter using a reinforcement learning scheme.

2.3.1.3 Physics is included in the training process

Sometimes, it is interesting to incorporate part of the Physics of the system to the different ML methods during the training process. This could be interesting at least for three reasons:

- For obtaining more interpretable results in the physical sense.

- For reducing the data needs in order to build accurate ML models.
- For obtaining more robust and stable numerical simulation schemes, avoiding pathological behaviours such as energy blow-ups.

Many authors have tried to incorporate physical knowledge to the different statistical and ML methods, via regularisation or explicitly introducing the constraints in the equations defining the latent structure of the ML model. In the first case, the ML loss $\mathcal{L}$ is adapted in order to incorporate penalty terms representing the discrepancy between the ML method performance and its ability to incorporate the physical knowledge. In other words, if $\mathcal{L}$ is the loss function of a given ML method, it is modified to $\mathcal{L}'$:

$$\mathcal{L}' = \mathcal{L} + \mathcal{L}_{\text{reg}}, \tag{2.27}$$

where $\mathcal{L}_{\text{reg}}$ will depend on how Physics is incorporated.

In the second case, the inclusion of specific constraints will lead to a different optimisation problem. For instance, let suppose that a certain ML method looks for the solution of a certain problem formulated as:

$$\boldsymbol{F}(\boldsymbol{x}; \boldsymbol{\Lambda}) = \boldsymbol{y}, \tag{2.28}$$

where $\boldsymbol{x}$ is the unknown solution, $\boldsymbol{y}$ the data or physical knowledge and $\boldsymbol{F}$ is a model, with model parameters $\boldsymbol{\Lambda}$. Then, except for very particular cases where $\boldsymbol{F}$ may be inverted, the solution is obtained by the minimisation of the residual $\mathcal{R}(\boldsymbol{x}) = \|\boldsymbol{F}(\boldsymbol{x}; \boldsymbol{\Lambda}) - \boldsymbol{y}\|$, that is, by solving the problem:

$$\boldsymbol{\Lambda}^* = \operatorname{argmin} \|\boldsymbol{F}(\boldsymbol{x}; \boldsymbol{\Lambda}) - \boldsymbol{y}\|. \tag{2.29}$$

If we know that the physics of the problem is expressed by the relation $\boldsymbol{R}(\boldsymbol{x}, \boldsymbol{y}) = \boldsymbol{0}$, the problem to be solved becomes:

$$\begin{aligned} \boldsymbol{\Lambda}^* = \underset{\boldsymbol{\Lambda} \in \mathcal{S}}{\operatorname{argmin}} \quad & \|\boldsymbol{F}(\boldsymbol{x}; \boldsymbol{\Lambda}) - \boldsymbol{y}\| \\ \text{subject to} \quad & \\ & \boldsymbol{R}(\boldsymbol{x}, \boldsymbol{y}) = \boldsymbol{0}. \end{aligned} \tag{2.30}$$

Loss regularisation

The most straightforward approach for including Physics during the training process is by including/adding regularisation terms in the loss function. Many regularisation techniques have been explored in the Data Science community to enforce different degrees of model complexity. Minimising the L_p norm of model parameters has been extensively used for obtaining different effects of regularisation in parametric model learning. The L_2 norm has been used to avoid overly large parameter values in Ridge regression (Hoerl and Kennard, 1970; Saleh et al., 2019) (also known as Tikhonov regularisation) and SVM. On the other hand, minimising the L_1 norm results in the Lasso formulation (Tibshirani, 1996) and the Dantzig selector (Candes and Tao, 2007; James et al., 2009), both of which encode sparsity in the model parameters. Whatever it is the ML approach to the problem, the loss function is formulated as:

$$\mathcal{L} = \alpha \mathcal{L}_{\text{DATA}} + \beta \mathcal{L}_{\text{PHYS}}, \tag{2.31}$$

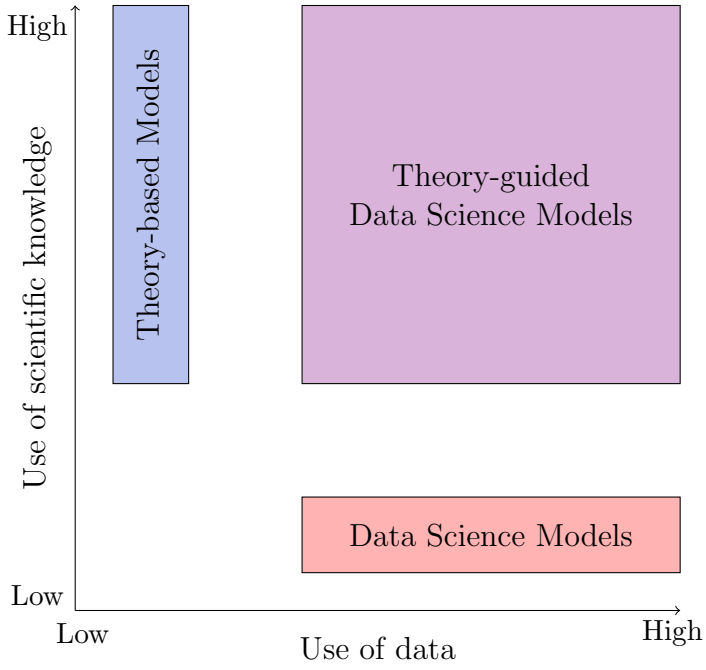


Figure 2.5: Theory-Guided Data Sciences. Theory-Guided Data Sciences are in the middle of Data Sciences and Theory-Based Models, where both physical knowledge and data are combined. Source: adapted from Karpatne et al. (2017b).

where $\mathcal{L}_{\text{DATA}}$ accounts for the knowledge about the system by means of labelled/measured data and $\mathcal{L}_{\text{PHYS}}$ incorporates, in one way or another, the physics of the system. This is what Karpatne et al. (2017b) defined as Theory-Guided Data Science, and was applied for predicting lake temperature profiles (Karpatne et al., 2017c) using what they called physically-guided neural networks (PGNN) (other works exist for referring to the same approach, such as Domain Adapted Neural Networks (Muralidhar et al., 2018; Stewart and Ermon, 2017)). If $\alpha = 0$, the ML method is unsupervised and only the physics of the system guide the learning process, whereas if $\beta = 0$ we recover pure Data Science ML methods. The important point is that hybrid formulations are possible, as illustrated in Fig. 2.5. Their work was extended using RNNs (Jia et al., 2019). The same approach was followed by Magiera et al. (2020), in what they defined as constraint-aware neural networks, for imposing the Rankine-Hugoniot condition between the input and output layers, identified with the states on both sides of a shock wave in hyperbolic PDEs.

Other authors include the physical constraints during the model selection process of a given ML method (Wagner and Rondinelli, 2016), such as decision trees, data clustering (Chatterjee et al., 2014) or PCA, in the so-called Theory-Guided ML methods. They show that not including the guidance of domain knowledge can lead to both quantitatively and qualitatively incorrect predictive models when applied to material sciences. In this line, a family of methods coined as Group Lasso (Simon et al., 2013; Yuan and Lin, 2006) try to group variables, as being the associated with a common physical feature. This is, for instance, typical of genetics (Liu et al., 2013) where some knowledge about a common linkage may be incorporated exogenously.

Solving Partial Differential Equations

A particular strategy for including Physics as a loss regularisation is via the residual of the PDE governing the problem physics. Many approaches have been proposed in the last decade, as summarised in Yadav et al. (2015). Maybe one of the first tentatives to use of ANN methods for solving ODEs was the one proposed by Lee and Kang (1990), where the minimisation algorithms used for ANNs were employed for solving the equations coming from the ODE discretisation. The idea was adapted by Meade Jr and Fernandez (1994a,b) to Petrov-Galerkin schemes and was also investigated by Malek and Beidokhti (2006) using a hybrid ANN-optimisation method. The idea was adapted for PDEs by Lagaris et al. (1998); van Milligen et al. (1995), whose main claim was to reduce the computational cost of standard simulation methods. Since then, many authors have adapted the ideas to richer problems (Alli et al., 2003; Effati and Pakdaman, 2010; Lagaris et al., 2000; McFall and Mahan, 2009). The main idea is to design an ANN able to infer the relation $(\boldsymbol{x}, t) \rightarrow \boldsymbol{u}$, something guaranteed by the universal approximation theorem (Cybenko, 1989; Hanin, 2017; Hornik, 1991; Lu et al., 2017; Pinkus, 1999). Hence, if the PDE is expressed as:

$$\mathcal{D}(u) = f, \quad \boldsymbol{x} \in \Omega, \tag{2.32a}$$
$$\mathcal{B}(u) = g, \quad \boldsymbol{x} \in \partial\Omega, \tag{2.32b}$$

where $\mathcal{D}$ is a differential operator, with some specific boundary conditions given by the operator $\mathcal{B}$, the solution u is sought using the decomposition:

$$u(\boldsymbol{x}) = u_{\text{BC}}(\boldsymbol{x}) + u_{\text{ANN}}(\boldsymbol{x}; \boldsymbol{\Lambda}), \tag{2.33}$$

with u_{BC} being a function satisfying boundary conditions and u_{ANN} the output of an ANN representing the relation $(\boldsymbol{x}, t) \rightarrow \boldsymbol{u}$ and depending on the network parameters $\boldsymbol{\Lambda}$. The network convergence is then obtained by defining the loss function in an appropriate manner, usually related to the method used for the PDE discretisation.

For instance, Dissanayake and Phan-Thien (1994) proposed to express the residual of the PDE and the boundary conditions as:

$$\mathcal{R}_{\text{PDE}}(u) = \mathcal{D}(u) - f, \tag{2.34a}$$
$$\mathcal{R}_{\text{BC}}(u) = \mathcal{B}(u) - g, \tag{2.34b}$$

and to solve the minimisation of the total residual:

$$R = \int_{\Omega} \|\mathcal{R}_{\text{PDE}}(u)\|^2 \, dV + \int_{\partial\Omega} \|\mathcal{R}_{\text{BC}}(u)\|^2 \, dS, \tag{2.35}$$

by using a quadrature method. Another possibility is Montecarlo sampling for the numerical integration. In this line, Sirignano and Spiliopoulos (2018), in addition to derive some mathematical results for quasilinear parabolic PDEs, established what they call the Deep Galerkin Method, for high dimensional free boundary PDEs, including extra dimensions for the boundary condition parametrisation. This method is nothing more than a meshfree evaluation of Eq. (2.35) by random sampling, combined with DL architectures for the networks considered, instead of shallow MLPs, and a new computational scheme for the efficient computation of ANN

gradients arising from the second derivatives of high-dimensional PDEs. In the same vein, (Berg and Nyström, 2018) adapted the method for complex geometries where the sampling strategy is a problem.

Similarly, Beidokhti and Malek (2009); Rudd and Ferrari (2015); Rudd et al. (2013) adapted some of these ideas to spectral methods and defined the Constrained Integration approach for pseudo-spectral (collocation) methods. They use RBF for approximating the function close to the collocation points, and minimise the residual using backpropagation techniques. A broad family of works use similar ideas, in what they call the Deep Collocation Method, for analysing problems of solid and structural mechanics (Abueidda et al., 2020; Guo et al., 2021) or heat transfer (Lin et al., 2020) and for adaptability strategies depending on the residual estimation (Anitescu et al., 2019). Indeed, sparsity has been exploited by Schaeffer et al. (2013) in combination with spectral methods. It is possible to consider a spectral expansion of the solution in terms of a complete basis $\{\Phi_i\}_{i=1,\dots,N}$:

$$u = \sum_{i=1}^{N} u_i \phi_i, \tag{2.36}$$

and try to build u from the spectral coefficients u_i, $i = 1, \dots, N$.

In another direction, and inspired by variational calculus, Yu et al. (2017) developed the Deep Ritz Method, which consists in the variational expression of the PDE, that is, in the solution of a problem with the structure:

$$\min_{u \in \mathcal{U}} \mathcal{I}(u), \tag{2.37}$$

being $\mathcal{I}$ a functional, such as $\mathcal{I}(u) = \int_\Omega \left(\frac{1}{2} \|\boldsymbol{\nabla} u\|^2 - f(x)u(x) \right) dx$ for Poisson equation $\Delta u = f$. By expressing $u = u(x; \boldsymbol{\Lambda})$, being $\boldsymbol{\Lambda}$ the network parameters, and identifying the numerical integration with a finite sum, it is possible to reformulate the stochastic gradient descent (SGD) method easily, using any quadrature rule for the integral evaluation, and to adapt it for transfer learning. This method has been adapted for many problems coming from CM when the minimisation of a total energy is sought, in what we call the Deep Energy Method (Samaniego et al., 2020), and for structural elements such as plates (Li et al., 2021), finite deformation hyperelasticity (Nguyen-Thanh et al., 2020), as well as for more flexible and general conditions (Liao and Ming, 2019; Lu et al., 2021a).

The rise of ANN specific software, incorporating powerful and easy-to-use automatic differentiation tools triggered a new impulse in this approach. One of the most popular works was the one developed by Raissi et al. (2019), including a first attempt in model explanation, by introducing some extra parameters to be learned, related with some of the physical phenomena. The idea behind these ANNs, baptised as physics-informed neural networks (PINNs), is illustrated in Fig. 2.6 both for unsupervised (forward) and supervised (inverse) problems: given a PDE, the ANN loss $\mathcal{L}$ is formulated in terms of the residual $\mathcal{R}$ of the equation and its boundary conditions. The parametric explanatory capacity is obtained by including some learning parameters at the residual and providing a data-set, such that $\mathcal{L} = \mathcal{L}_{\text{DATA}} + \mathcal{L}_{\text{PHYS}}$ accounts both for the physical structure and for the data. In any case, the inclusion of the physical

term may be intended merely for regularisation purposes (Nabian and Meidani, 2020) and has demonstrated to have some generalisation capability (Magill et al., 2018). This family of ANNs has been exploited in many contexts, such as fluid mechanics (Mao et al., 2020; Raissi and Karniadakis, 2018; Sun et al., 2020), where they have been particularly prolific due to the turbulence closure problem (Yang et al., 2019) and complex fluid behaviour (Reyes et al., 2021), solid and structural mechanics (Haghighat et al., 2021; Li et al., 2021), fractional calculus equations (Pang et al., 2019), biology and biomedical engineering (Lagergren et al., 2020; Sahli Costabal et al., 2020; Yazdani et al., 2020), epidemiology (Shaier et al., 2021), electromagnetism (Noakoasteen et al., 2020), optics (Chen et al., 2020b), power systems (Misyris et al., 2020) and chemical engineering (Ji et al., 2021). The methodology has been adapted for incorporating other theoretical knowledge about the system (theory-guided neural network (TGNN) (Wang et al., 2020)), for different network architectures (see for instance (Wang et al., 2021b) for Theory-Guided Autoencoders), for allowing domain decomposition with jump discontinuities (Jagtap et al., 2020), extended to Bayesian frameworks for a data noise incorporation (Yang et al., 2021) and to UQ using adversarial strategies (Yang and Perdikaris, 2019). Very recently, some theoretical works about PINNs convergence and performance are starting to emerge (see for instance Hutzenthaler et al. (2020) for the semilinear heat equation and Kutyniok et al. (2021) for general convergence results and a discussion on their impact and extensions), along with adaptative training strategies (Wang et al., 2021c, 2022) and more sophisticated training processes, combining high-fidelity and low-fidelity estimators (Liu and Wang, 2019). Also, Nabian and Meidani (2018) adapted the method for random PDEs, both in the strong and weak (variational) form, using a polynomial chaos expansion (Xiu and Karniadakis, 2002) and Zhang et al. (2020) for SPDEs using modal decomposition. Nowadays, PINNs are already in a mature phase and specific software libraries exist to deal with physical problems using this approach (Lu et al., 2021b).

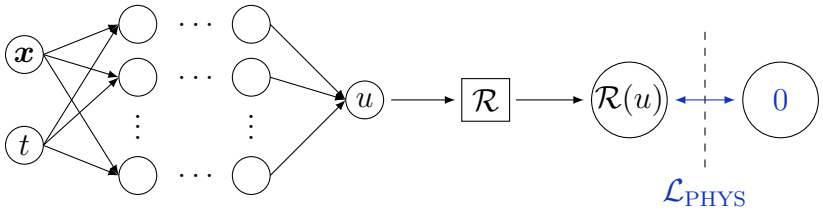
Very recently, (Khoo et al., 2021) changed the approach and considered as the input of the network the (discretised representation of a given) field intervening in the PDE in hands. More in detail, given a parametric field $k(x)$, they consider equations defined as:

$$\mathcal{D}(u, k) = f. \tag{2.38}$$

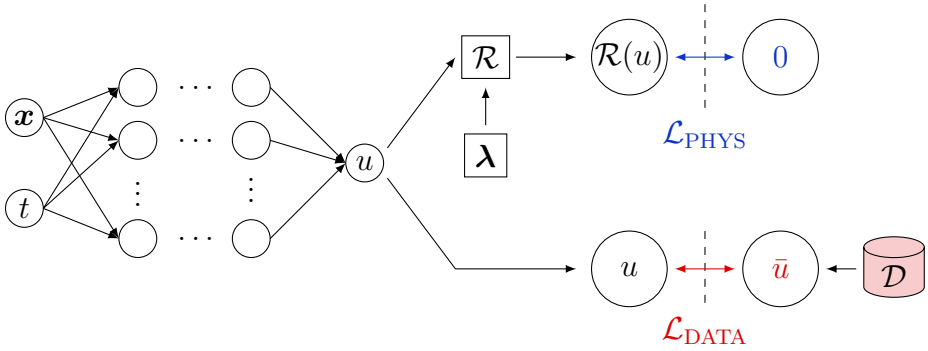
A natural example is the heterogeneous heat equation $u_t + \boldsymbol{\nabla} \cdot (k(x)\boldsymbol{\nabla} u) = f(x)$. The goal is to predict the value of a given QoI, $q = \mathcal{Q}(u)$ form the field $k(x)$. For such purpose, they define an ANN for learning the relationship $\boldsymbol{k} \to q$, where $\boldsymbol{k}$ is a discrete representation of the field k. This approach was extended by Zhu et al. (2019) for predicting the relationship $\boldsymbol{k} \to \boldsymbol{u}$, being $\boldsymbol{u}$ a discrete representation of u, and adapted to stochastic frameworks, using Kullback-Leibler divergence (Hall, 1987).

Including physical constraints

Lastly, the numerical methods employed for discretising and solving the equations or ROM do not exactly preserve some physical rules (this is the case, for instance, of Galerkin projection-based methods (Noack et al., 2003; Semaan et al., 2016), such as finite element method (FEM) for PDE solving, or POD for ROM, unlike



(a) Unsupervised approach for forward problems.



(b) Supervised approach for inverse problems.

Figure 2.6: Scheme of the PINN structure. The PINN is able to learn the shape of the solution field u and, with the inclusion of a data-set, the value of some learning parameters $\boldsymbol{\lambda}$, acquiring some explanatory capacity.

other methods that by construction conserve universal balances, such as FVM). For stability purposes, it is occasionally convenient to explicitly include these physical constraints (Schlegel and Noack, 2015). Carlberg et al. (2015) discuss a method for preserving Lagrangian dynamics explicitly, and a similar approach was taken by Balajewicz et al. (2013) for constraints coming from Navier-Stokes equations (Balajewicz et al., 2013), adding a power balance constraint included at the modal level, achieving more stable simulations. Very recently, the methods have been exported from fluid mechanics to magnetohydrodynamics (Kaptanoglu et al., 2021).

2.3.2 In terms of the knowledge available

So far, we have classified the different PIDS methods in terms of how, from an implementation point of view, Physics is treated or included in the computations. Now we change slightly the approach and analyse to what extent Physics is incorporated. In all the methods described above, we assume that, when some features of the physical knowledge are incorporated to the ML method, these are included as full knowledge, except, at most, for some learning parameters. It is important to insist on this:

1. When Physics is included in the data-set, such data-set is assumed to fulfil the physics we want to learn, except for the fact that it might be noisy data.
2. When Physics is included in the ML method, the method itself is designed for such purpose, so it respects the physical structure (for instance, PGD works with the exact equations of the system, hybrid ANNs or SPNNs enforce conservation laws or thermodynamics, and the methods designed for specific PDEs are designed

according to their structure). Some of these components, of course, may include some model parameters (such as the network weights and biases, $\mathbf{\Lambda}$) but they are put in specific components of the ML method.

3. When Physics is included in the learning process, it is by means of the exact knowledge in terms of a given equation representing the system physics. This equation can be, of course, parametrised in terms of some model parameters, as it is done in PINNs, but these could be later interpreted as extra model parameters $\mathbf{\Lambda}$.

In what follows, we present other methods, where the aim is to build the physical knowledge from data, leveraging the different levels of knowledge on the system:

1. All the physics is intended to be discovered: learning physics from data.
2. The system is able to assimilate some of the physics from data, or to correct it dynamically: DDDAS.
3. Some of the physics is fully known, and the other is replaced by the data: pure DD approaches.
4. Some of the physics is fully known, and the other is inferred from the data: building constitutive manifolds.

2.3.2.1 Learning physics from data

For centuries, researchers and scientists have attempted to unveil the mathematical laws underlying the different physical phenomena in Nature. Despite the prevalence of computing power, the process of finding the mathematical equations of natural laws has resisted automation. The first modern approaches consist of standard parametric fitting applied to the discretised version of a PDE using numerical differentiation (Bär et al., 1999; Voss et al., 1998). The thoughtful works by Bongard and Lipson (2007); Schmidt and Lipson (2009) signalled the start of a new era for unveiling natural laws from data, demonstrating the ability to unravel the conservation of Hamiltonians and Lagrangians, among other laws of geometric and momentum conservation.

Sparse identification of Nonlinear Dynamics

Despite the contemporary relevant attempts from Daniels and Nemenman (2015); Peherstorfer and Willcox (2016); Quade et al. (2016); Schaeffer and McCalla (2017), maybe one of the most prominent works is the one by Brunton et al. (2016), where sparse identification was used to discover the hidden physics of a dynamical system. Consider a dynamical system given by:

$$\dot{\boldsymbol{u}} = \boldsymbol{F}(\boldsymbol{u}), \tag{2.39}$$

where $\boldsymbol{F}$ is an (unknown) flow function. Their method uses a library of nonlinear functions of the state $\boldsymbol{u}$ to approximate $\boldsymbol{F}$. A snapshot matrix is built from data collected at different time frames, $\boldsymbol{U}$, with $U_{ij} = u_j(t = t_i)$, and the library is

expressed as $\boldsymbol{\Theta}(\boldsymbol{U})$, formed by the horizontal concatenation of the candidate nonlinear functions. Then, the solution is sought to be:

$$\dot{\boldsymbol{U}} = \boldsymbol{\Theta}(\boldsymbol{U})\boldsymbol{\Xi}, \tag{2.40}$$

where $\boldsymbol{\Xi}$ is a sparse coefficient matrix, whose values are obtained by sparse regression. The overall process is illustrated in Fig. 2.7 for the Lorentz system. This strategy provides "models (that) are parsimonious, balancing model complexity with descriptive ability while avoiding overfitting". Since their work, *parsimony* has become one of the main desired properties of ML methods seeking to discover the hidden physics of a system. The method was extended to incorporate explicit constraints, such as energy conservation or spatial symmetries (Loiseau and Brunton, 2018), and to coordinate identification by means of an autoencoder structure for latent variable identification (Champion et al., 2019a). Other studies tried to improve these strategies for noisy data, adding non-convex features to the optimisation algorithm (Schaeffer and McCalla, 2017), corrupted data, incorporating compressing sensing (Tran and Ward, 2017) or limited data, by leveraging the sparse structure and including a sampling strategy (Schaeffer et al., 2018). The same procedure has been also adapted to PDEs (Rudy et al., 2017; Schaeffer, 2017), including the ones with time-dependent parameters (Rudy et al., 2019a), and to multiscale contexts Brunton and Kutz (2019); Champion et al. (2019b). Also, some theoretical results on the method convergence have been derived (Zhang and Schaeffer, 2019).

Other variations of the approach have been proposed to combine Sparse Identification with different techniques for dynamical system identification. Mangan et al. (2017, 2019) suggested to combine the method with the Akaike Information Criterion (Akaike, 1998) for accelerating the model selection. Brunton et al. (2017); Lusch et al. (2018) incorporate Koopman operator theory (Koopman, 1931; Mauroy et al., 2020) elements and Nguyen et al. (2019) elements of expectation-maximisation (Lusch et al., 2018). Finally, Loiseau et al. (2018) place sparse identification in the context of ROM for full-state reconstruction from sensor data.

Some examples of application of sparse identification are fluid flows (Loiseau and Brunton, 2018; Loiseau et al., 2018), structural and mechanical systems Lai and Nagarajaiah (2019); Li et al. (2019), biological and chemical systems Bhadriraju et al. (2019); Hoffmann et al. (2019); Mangan et al. (2016), material sciences (Brunton and Kutz, 2019) or population dynamics Dam et al. (2017); Kaiser et al. (2018), to cite only a few.

Partial differential equations discovery and ML

Raissi (2018) extended the work for PINNs to learn the structure itself of PDEs by replacing the residual of the PDE by a neural network. Indeed, let us consider the PDE:

$$u_t = \mathcal{D}(u), \tag{2.41}$$

where $\mathcal{D}$ is a certain differential operator. PINNs were formulated by describing the residual as:

$$\mathcal{R}(u) = u_t - \mathcal{D}(u). \tag{2.42}$$

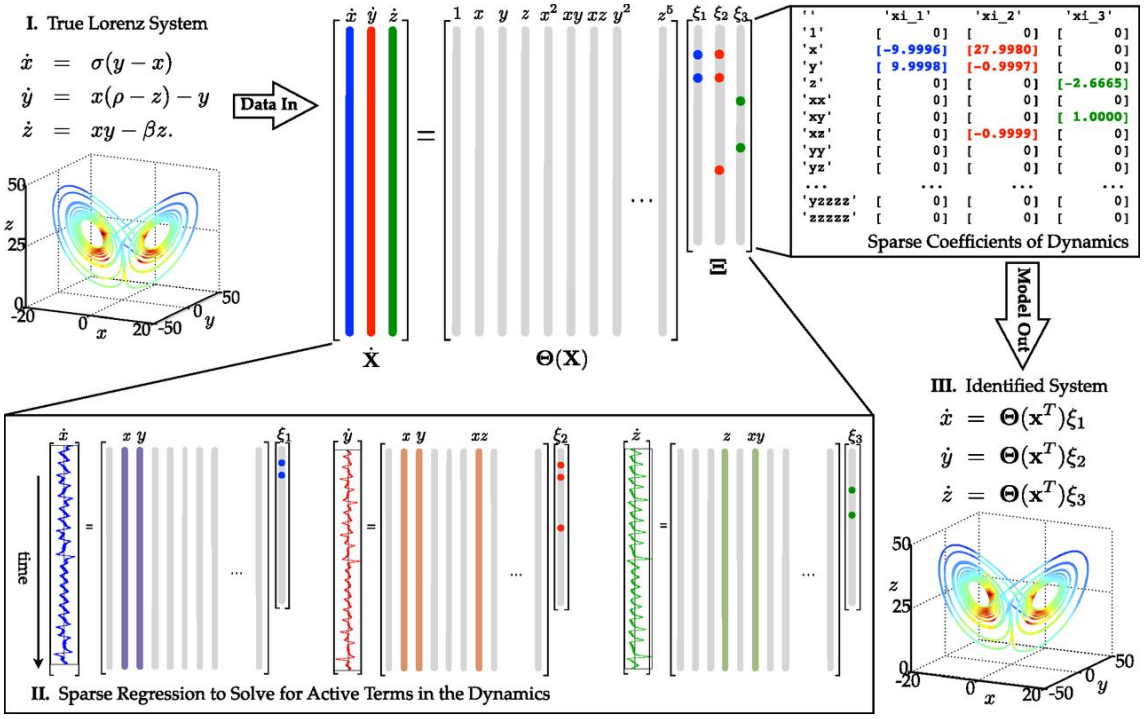


Figure 2.7: Sparse identification of Physical systems. Schematic of the Sparse Identification of Nonlinear Dynamics demonstrated on the Lorenz equations. Data are collected from the system, including a time history of the states and derivatives. A library of nonlinear functions of the states is constructed. This nonlinear feature library is used to find the fewest terms needed to satisfy the snapshot dynamical equation. The few entries in the vectors of parameters, solved for by sparse regression, denote the relevant terms on the right-hand side of the dynamics. The trajectory on the Lorenz attractor is coloured by the adaptive time step required, with red indicating a smaller time step. Source: Brunton et al. (2016), so the variable notation comes from the original work.

Now, the idea is to replace the operator $\mathcal{D}(u)$ by another ANN able to discover the relation:

$$\mathcal{D}(u) = \mathsf{Y}(t, x, u, \boldsymbol{\partial} u, \boldsymbol{\partial}^2 u, \ldots), \tag{2.43}$$

where $\boldsymbol{\partial} u$ denotes first partial derivatives, $\boldsymbol{\partial}^2 u$ second partial derivatives and so on. Hence, the residual is expressed by:

$$\mathcal{R}(u) = u_t - \mathsf{Y}(t, x, u, \boldsymbol{\partial} u, \boldsymbol{\partial}^2 u, \ldots), \tag{2.44}$$

so the method looks for simultaneously fitting the two ANNs:

$$(t, x) \to u, \tag{2.45a}$$

$$(t, x, u, \boldsymbol{\partial} u, \boldsymbol{\partial}^2 u, \ldots) \to \mathcal{D}(u). \tag{2.45b}$$

A similar approach was proposed for ODEs based on multistep integrators (Raissi et al., 2018a). The problem with this approach is that it is still a "black-box" as it cannot recover the structural form of the learned equations. To overcome this limitation, Both et al. (2021) used the same ideas but including sparsity by considering a Lasso approach to ANN training:

$$\mathcal{L} = \mathcal{L}_{\text{DATA}} + \mathcal{L}_{\text{PHYS}} + \mathcal{L}_{L_1}, \tag{2.46}$$

where $\mathcal{L}_{\text{PHYS}}$ accounts for a sparse reconstruction of the differential operator and $\mathcal{L}_{L_1}$ for Lasso regularisation. More recently, Chen et al. (2021) adapted the PINN framework for scarce data using sparse regression (PINN-SR) and adapted the approach for data coming from multiple boundary and initial conditions, combining the best features of the sparse identification and PDE discovery approaches. Also, Berg and Nyström (2019) solved this problem by splitting the process into two steps, one in which the relation $(\boldsymbol{x}, t) \to \boldsymbol{u}$ is learned and another for the operator $\mathcal{D}$ discovery.

In a similar vein, Long et al. (2018, 2019) developed PDE-Nets. This family of ANNs use the most recent ideas of ANNs and DL. They take advantage of the parallelism between convolution and differentiation (Cai et al., 2012; Dong et al., 2017) and accordingly they establish general rules for building the approximate solution. The differential equation is expressed as:

$$u_t = \boldsymbol{F}(\boldsymbol{x}, u, \boldsymbol{\partial} u, \boldsymbol{\partial}^2 u, \ldots). \tag{2.47}$$

That is, the differentiation order of the equation is specified, as well as the response function F as a vectorial function of the derivative fields up to the differentiation order. Next, the differential operators are replaced by spatial filters, and the response $\boldsymbol{F}$ is expressed as a nonlinear function of the filter's output, using other ANN structures. In the discrete representation, the solution is sought as:

$$\hat{\boldsymbol{u}}_{k+1} = \boldsymbol{u}_k + (\Delta t)\mathsf{F}(\boldsymbol{x}, \boldsymbol{u}_k, \boldsymbol{D}_{I_1}(\boldsymbol{u}_k), \boldsymbol{D}_{I_2}(\boldsymbol{u}_k), \ldots), \tag{2.48}$$

where $\boldsymbol{u}_k$ are discrete fields treated as images, $\boldsymbol{D}_{I_\nu}$, $I_\nu = (i_\nu, j_\nu, k_\nu)$ are appropriate CNNs representing $\partial_{I_\nu} = \frac{\partial^{i+j+k}}{\partial^i x, \partial^j y \partial^k z}$, and F is the approximation of $\boldsymbol{F}$ using any ANN regression technique. The PDE structure is learned using the loss function

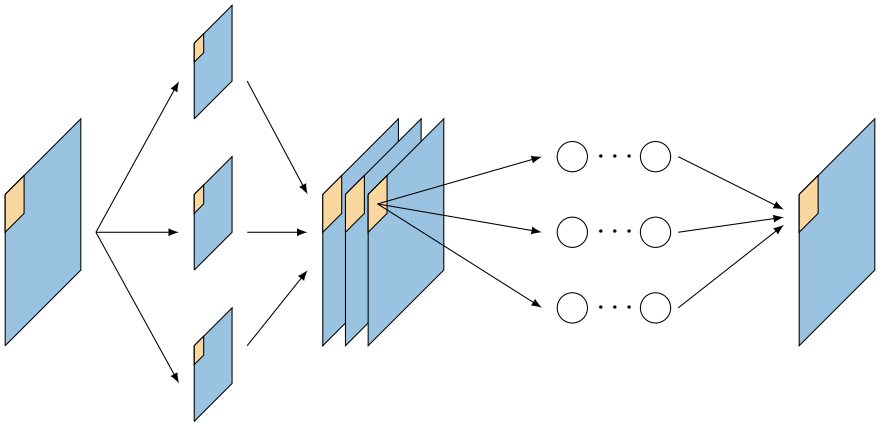


Figure 2.8: Structure of a PDE-Net. From the known field u we define prescribed convolutional filters that act as discrete differential operators, and then any ML regressor (e.g. a MLP) expresses $\boldsymbol{F}$ as a function of the field derivatives. The network output variable is then computed using Eq. (2.48).

$\mathcal{L} = \sum_{k=1}^{m} \|\hat{\boldsymbol{u}}_{n+k} - \boldsymbol{u}_{n+k}\|^2$, that is, the prediction error during m steps. The overall process for estimating $\boldsymbol{F}$ is illustrated in Fig. 2.8. Due to the network structure, the method has both predictive and explanatory capacity as it can predict the system evolution and discover the structure of the response function $\boldsymbol{F}$.

There are other recent approaches for PDE discovery using ML techniques. For instance, Wang et al. (2019) suggest to express symbolically a broad class of PDEs as:

$$\frac{\partial u}{\partial t} - \boldsymbol{\chi}^{\intercal}\boldsymbol{\omega} = 0, \tag{2.49}$$

where $\boldsymbol{\chi} = (1, u, u^2, \ldots, \boldsymbol{\partial u}, \ldots)^{\intercal}$ and $\boldsymbol{\omega}$ is a vector of coefficients (pre-factors). The form of the equations is determined by expressing Eq. (2.49) in the weak form and then solving a minimisation problem fed by data.

2.3.2.2 Data-Driven Assimilation Systems

The second family of methods is the one where data is, in one way or another, explicitly incorporated to the system knowledge. In DDDAS (Darema, 2004), the idea is to provide both predictive and learning capabilities to the control system from data acquired from a sufficient set of sensors. This paradigm was settled down by Kalman (1960) in the sixties with his groundbreaking filter and is still nowadays a hot topic of research opening up a huge range of possibilities.

Data assimilation, reduced models and model corrections

For instance, Peherstorfer and Willcox (2015) adapted the ROM philosophy to dynamical contexts. Let us express an evolving system (obtained, for instance, from the discretisation of a PDE):

$$\boldsymbol{A}_{\boldsymbol{\eta}}(\boldsymbol{\mu})\boldsymbol{u}_{\boldsymbol{\eta}}(\boldsymbol{\mu}) = \boldsymbol{f}(\boldsymbol{\mu}), \tag{2.50}$$

where $\boldsymbol{\eta}$ and $\boldsymbol{\mu}$ are the observable and hidden parameters, respectively. Suppose that the operators $\boldsymbol{A}_{\boldsymbol{\eta}}$ and $\boldsymbol{f}$ have an affine representation in terms of parameters $\boldsymbol{\mu}$. The

observable parameters $\boldsymbol{\mu}$ are the ones that are fed by data, and at each measurement, a reduced based built using POD is enriched using low-rank updates and fitting the new coefficients solving a minimisation problem (something that is not expensive in a low-dimensional space), and so it is for the reduced operators. Also, dynamic mode decomposition (DMD) (Proctor et al., 2016; Schmid, 2011; Tu, 2013) and similar approaches (Peherstorfer and Willcox, 2016) have been used for ROM in dynamic contexts and (Rowley and Dawson, 2017) revisited the analysis of the use of ROM techniques and DDDAS for flow problems using control theory and the Koopman operator. González et al. (2017) adapted the Kalman filter ideas to nonlinear models by using ROM techniques.

With the advent of the GENERIC framework, new DDDAS have been formulated for learning the discrepancy between a model and real data, using a thermodynamic consistent approach. (González et al., 2019) demonstrate that it is possible to learn hyperelastic models from data. Indeed, by considering an additive decomposition $\boldsymbol{u}^{\text{exp}} = \boldsymbol{u}^{\text{m}} + \boldsymbol{u}^{\text{corr}}$ and assuming the GENERIC metriplectic evolution equation for the corrections, they obtained a new metriplectic evolution for the measured variable:

$$\dot{\boldsymbol{u}}^{\text{exp}} = \boldsymbol{L}\left(\frac{\partial E}{\partial \boldsymbol{u}}(\boldsymbol{u}^{\text{mod}}) + \frac{\partial E}{\partial \boldsymbol{u}}(\boldsymbol{u}^{\text{corr}})\right) + \boldsymbol{M}(\boldsymbol{u}^{\text{corr}})\frac{\partial S}{\partial \boldsymbol{u}}(\boldsymbol{u}^{\text{corr}}). \tag{2.51}$$

where the superscripts "exp", "mod" and "corr" denote the experimental measurements, the model and the corrections. With this approach, it is possible to detect the dissipative correction to the model (that, if hyperelastic, is zero by definition) and even to correct the hyperelastic modelling error.

Another approach to correct hyperelastic models was proposed by Ibáñez et al. (2019), using a response surface for the error approximation, and similarly by Lam et al. (2017), who expressed a model as the result of three interacting features: fidelity (related with the error), virtues (desirable properties of the model) and structure (a parametric equation). Recently, Rudy et al. (2019b) developed a method for learning both the underlying model and the structure of the noise component using DL methods. Finally, de Silva et al. (2020) defined an approach to the discrepancy understanding using the Sparse Identification framework.

Hybrid twins and augmented reality

DDDAS were combined with the concept of Digital Twin, to incorporate the user interaction in what was coined as the Hybrid Twin (Chinesta et al., 2020; Moya et al., 2020b). The Hybrid Twin may be mathematically formulated by expressing the evolution equation as:

$$\dot{\boldsymbol{u}} = \boldsymbol{A}^{\text{m}}(\boldsymbol{u}, t, \boldsymbol{\lambda}) + \boldsymbol{A}^{\text{corr}}(\boldsymbol{u}, t) + \boldsymbol{F}(t) + \boldsymbol{\varepsilon}(t) \tag{2.52}$$

where:

- $\boldsymbol{A}^{\text{m}}$ is the system model, depending on some parameters $\boldsymbol{\mu}$.
- $\boldsymbol{A}^{\text{corr}}$ is the model correction, or, as expressed by Chinesta et al. (2020), the "ignorance".

- $\boldsymbol{F}(t)$ are the user interactions that may force the system to evolve in one way or another.
- $\boldsymbol{\varepsilon}(t)$ is the error term.

The concept of Hybrid Twins has been fruitfully exploited in real-interactive and augmented reality platforms. For instance, Badías et al. (2019) developed an augmented reality platform for aerodynamics design, Moya et al. (2020a) exploited the digital twin concept in slosh dynamics and Sancarlos et al. (2021) in electrochemistry for lithium-ion batteries. Also, Singh et al. (2017); Wang et al. (2017); Xiao et al. (2016) applied different ML methods for quantifying, reducing and learning the discrepancies between the data and a prescribed model for reconstructing Reynolds stress tensor in turbulent flows. Finally, combining the Hybrid Twin concept with ROM achieved real-time user interactions in mixed reality platforms (Badías et al., 2020).

2.3.2.3 Raw data approaches: Data-Driven techniques and interpolation

Another approach to PIDS is to split the problem physics in the two components described in Section 2.2: universal laws (and boundary conditions), that were denoted by $\boldsymbol{F}$ and $\boldsymbol{G}$, and constitutive equations, denoted by $\boldsymbol{H}$, relating some of the variables of the system. In terms of the set of state variables of the system, $(\boldsymbol{u}, \boldsymbol{v})$, where $\boldsymbol{u}$ are the measurable state variables and $\boldsymbol{v}$ the non-measurable hidden ones, the manifold of admissible states is:

$$\mathcal{P} = \{(\boldsymbol{u}, \boldsymbol{v}) \in \mathbb{R}^M | \boldsymbol{F}(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{f}) = \boldsymbol{0}, \boldsymbol{G}(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{g}) = \boldsymbol{0}, \boldsymbol{H}(\boldsymbol{u}, \boldsymbol{v}) = \boldsymbol{0}\}. \tag{2.53}$$

However we may leverage the degree of knowledge about the system and merely consider the universal laws:

$$\mathcal{P} = \{(\boldsymbol{u}, \boldsymbol{v}) \in \mathbb{R}^M | \boldsymbol{F}(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{f}) = \boldsymbol{0}, \boldsymbol{G}(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{g}) = \boldsymbol{0}\}. \tag{2.54}$$

In that case, however, the problem remains undetermined and a data-set has to be provided in order to make good predictions about the system state. Here is where Data Science, and in particular, CM, comes to the rescue. Indeed, Material Informatics (Agrawal and Choudhary, 2016; Broderick and Rajan, 2015; Rajan, 2005, 2015) is a growing, although recent, field that aims to characterise material properties from the Big Data perspective.

Data-Driven Computational Mechanics

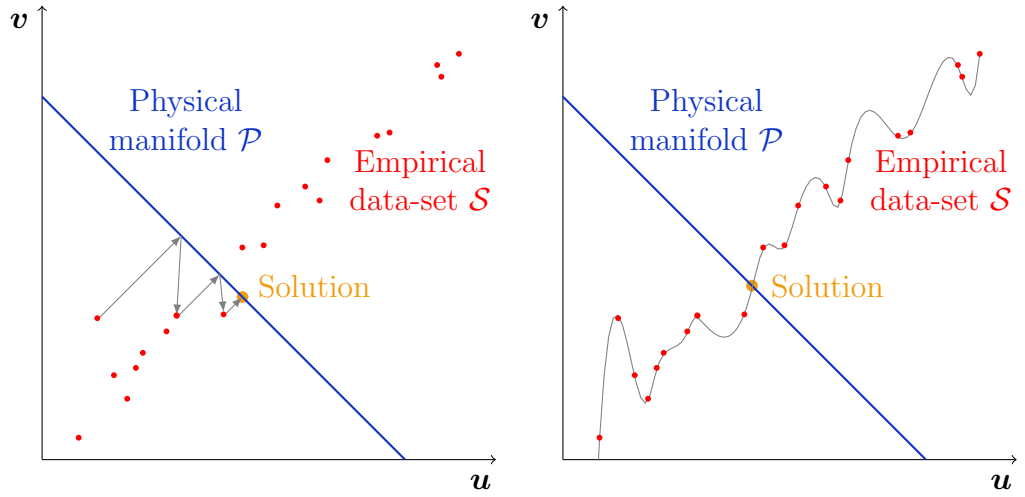
In this approach, the raw data-set plays the role of the constitutive equations, except for the fact that no necessary extra structure has to be considered. Hence, the data-set to be supplied has the form:

$$\mathcal{D} = \{(\boldsymbol{u}^i, \boldsymbol{v}^i),\ i = 1, \dots, N\}. \tag{2.55}$$

These are the main ingredients of what Kirchdoerfer and Ortiz (2016b) coined as DD CM in their seminal work for the elastic problem. If $P = (\boldsymbol{u}, \boldsymbol{v}) \in \mathcal{P}$ and $P' \in \mathcal{D}$, the essential idea is to consider the actual state as the solution of the minimisation problem:

$$\min_{P' \in \mathcal{D}, P \in \mathcal{P}} d(P, P'). \tag{2.56}$$

Due to the discrete nature of $\mathcal{D}$, the problem is solved iteratively by combining one local search to look for the closest point at the material data-set $\mathcal{D}$, using k-nearest neighbours' algorithm, and a projection step to the physical manifold $\mathcal{P}$. The idea and the algorithmic procedure are illustrated in Fig. 2.9a. Their ideas were later extended fruitfully to dynamics, using the Newmark integration algorithm (Kirchdoerfer and Ortiz, 2018), inelastic problems (viscoelasticity and plasticity), using the internal variables formalism (Eggersmann et al., 2019) or linear plasticity theory (Ladevèze et al., 2019), noisy data-sets, by means of clustering analysis and maximum - entropy estimation (Kirchdoerfer and Ortiz, 2017), kernel regression (Kanno, 2018), or locally convex reconstruction (He and Chen, 2020)[6], finite strains theory and large deformations (Nguyen and Keip, 2018; Platzer et al., 2021), multiscale modelling (Karapiperis et al., 2021a), fracture mechanics (Carrara et al., 2020) and nonlocal mechanics (Karapiperis et al., 2021b). From the computational point of view, Eggersmann et al. (2021) investigated efficient data structures and algorithms for the local search, surpassing the standard k-nearest neighbours' algorithm and enhanced the performance by building locally linear tangent spaces using tensor voting (Mordohai and Medioni, 2010). On the other hand, (Nguyen et al., 2020) formulated the problem using an abstract variational framework and Amir Siddiq (2021) presented a FEM practical implementation. DD procedures have also been investigated from the mathematical point of view, both for infinitesimal (Conti et al., 2018) and finite theory (Conti et al., 2020). The method performance has been evaluated in soft tissue biomechanics (He et al., 2021a). Also, the ideas have started to be exported to other fields of continuum Physics such as electrostatics or magnetostatics (Galetzka et al., 2021a,b).



(a) DD procedure. **(b)** Instrumental interpolation.

Figure 2.9: Different raw data approaches to PIDS. We show the ideas behind the method and an illustration of the numerical algorithm.

Using the same ideas, but rethinking them in the opposite direction, Leygue et al. (2018) described DD identification, where now we look for the material response

[6]Strictly speaking, this enhancement of the DD procedure belongs to the family of Manifold Learning procedures, which will be discussed later, although we have considered to place it here since it is born from the necessity of improving raw DD approaches.

(that is, its constitutive equation), rather than the global stress-deformation state. From the kinematic global response of a material obtained from measurements using, for example, digital image correlation (DIC) (Sztefek et al., 2010) and the external forces, the DD solver is then "inverted" to obtain the stress state, therefore obtaining the pairs (ε, σ), that is, the geometry of the state space:

$$\mathcal{S} = \{(\boldsymbol{u}, \boldsymbol{v}),\ \boldsymbol{H}(\boldsymbol{u}, \boldsymbol{v}) = \boldsymbol{0}\}. \tag{2.57}$$

In other words, DD identification is able to measure stress fields without constitutive equations (Dalémat et al., 2019). The work was extended to non-linear elasticity, plasticity and dynamics (Leygue et al., 2019). The sequential combination of both approaches has yielded good results as a sound and consistent workflow in CM (Stainier et al., 2019).

Instrumental interpolation

Another approach to overcome constitutive modelling without the need of state equations (or, what is the same, the definition of a global strain energy function) is to fit the material data locally using piece-wise spline interpolations of stress-strain data obtained from tension-compression tests. The constitutive equation is computed, but is merely instrumental and for computational purposes (see Fig. 2.9b). This is the approach followed by Sussman and Bathe (2009), using the inversion formula (Kearsley and Zapas, 1980). The work was extended to isotropic hyperelasticity (Latorre and Montáns, 2013), and to orthotropic Latorre and Montáns (2014) and compressible materials (Crespo et al., 2017), in what was called the what you prescribe is what you get (WYPIWYG) approach, in which the splines interpolations are computed for both the energy and stress–strain data. The method has been improved to properly assimilate fuzzy experimental data by means of curvature smoothing (Latorre and Montáns, 2020). WYPIWYG has demonstrated to be able to capture the complexity of real soft tissues (Latorre and Montáns, 2017a,b; Latorre et al., 2017; Romero et al., 2017). Finally, the work has also been extended to damage mechanics (Miñano and Montáns, 2018).

2.3.2.4 Building the constitutive manifold

In this approach, the problem physics is split into the two same components but now, the aim is to reconstruct a manifold from the sampled data at the space $\mathcal{D}$. The manifold of admissible values is the one given by Eq. (2.54), but now, our aim is to build (we will see later how) a constitutive continuous manifold $\mathcal{E}$ such that the actual state of the problem is found by computing $\mathcal{P} \cap \mathcal{E}$. The problem is therefore reduced, up to computational considerations for evaluating the manifold intersection, to obtain from the (possibly noisy) data $\mathcal{D}$, the constitutive manifold $\mathcal{E}$.

Manifold Learning and kernel regression

Among the methods that have been proposed for building the underlying manifold from data, we can highlight kPCA (Schölkopf et al., 1998), self-organizing map (SOM) (Kohonen, 1990), locally linear embedding (LLE) (Roweis and Saul, 2000), Isomap (Tenenbaum et al., 2000), Laplacian Eigenmap (Belkin and Niyogi, 2003),

t-distributed stochastic Neighbour embedding (t-SNE) (Maaten and Hinton, 2008) or the many techniques related to topological data analysis (TDA) (Carlsson, 2009). The application of these methods for underlying manifolds with physical structure has been prolific. For instance, Ibanez et al. (2018), used LLE to build the constitutive manifold, and then investigated two extra different possibilities for solving the problems formulated in the weak form discretised by FEM, in addition to the pure DD approach at the reduced space: to approximate the space using the tangent or secant method from the reduced space. Then, the problem is again solved by combining a local step, where the point is projected to $\mathcal{E}$ and a global step, where a search direction in the stress-strain space is selected, which could be adapted at each iteration. If the search direction is assumed as constant, we recover the LaTIn linearisation technique (Ladevèze, 1989). The approach has been demonstrated to be effective in Materials Engineering, where material heterogeneity increases the problem complexity, and remaining in low order manifolds may considerably speed up computations (Lopez et al., 2018), and since then, other works have tried to lighten the need for data of common manifold learning algorithms (Ibañez et al., 2017) and have extended the methodology to plasticity (Ibáñez et al., 2019). Despite the effectiveness of Manifold Learning algorithms in providing low-dimensional representations for high-dimensional data-sets, their main drawback stems from the fact that they do not provide a map for transforming the data from the latent space to their high-dimensional representations in the original space. This problem is known in the literature as the pre-image problem (Kwok and Tsang, 2004). In response to this problem, He et al. (2021b) suggest the use of deep autoencoders in combination with DD schemes as a successful representation of the constitutive manifold in a low dimensional space.

Constitutive equations in PDEs

Previous works to the use of ML methods in PDE discovery from data are forward sensitivity analysis (FSA) and adjoint sensitivity analysis (ASA) (Tarantola, 2005). These methods result in parameter estimation of some specific PDEs with fixed mathematical structure (Cao et al., 2002, 2003; Maly and Petzold, 1996). FSA computes the model sensitivity along with the model evaluation, and ASA obtains the gradient of the mismatch error by solving the adjoint linearised PDE. Zhao et al. (2020) adapted FSA for learning the physics of pattern formation in nonlinear systems from images (applied to the Cahn-Hilliard and Allen-Cahn equations and dynamical density functional theory), using a spectral expansion of the functions involving the constitutive equation. In particular, they demonstrated that it is possible to learn constitutive relationships in phase transition systems, although they are limited to a parametric spectral expansion.

In another direction, Tartakovsky et al. (2018) developed an approach for learning heterogeneous constitutive laws. Starting from the diffusion equation:

$$\boldsymbol{\nabla} \cdot (K(\boldsymbol{x}, u)\boldsymbol{\nabla} u) = 0, \tag{2.58}$$

they defined the two ANNs $(\boldsymbol{x}, u) \rightarrow \boldsymbol{K}$ and $(\boldsymbol{x}) \rightarrow \boldsymbol{u}$, respectively K and U, and then expressed the loss function of the combined networks as:

$$\mathcal{L} = \frac{1}{N_K}\sum_{i=1}^{N_K}(\mathsf{K}(\boldsymbol{x}_i, \mathsf{U}(\boldsymbol{x}_i)) - K_i)^2 + \frac{1}{N_u}\sum_{i=1}^{N_u}(\mathsf{U}(\boldsymbol{x}_i) - u_i)^2 + \frac{1}{N_D}\sum_{i=1}^{N_D}\left(\mathsf{U}(\boldsymbol{x}_i^D) - u_i^*\right)^2 + \frac{1}{N_N}\sum_{i=1}^{N_N}\left(f_N(\boldsymbol{x}_i^N) - g_i^*\right)^2 + \frac{1}{N_c}\sum_{i=1}^{N_c} f(\boldsymbol{x}_i)^2. \quad (2.59)$$

where $f_N(\boldsymbol{x}) = \boldsymbol{n} \cdot (\mathsf{K}(\boldsymbol{x}, u)\boldsymbol{\nabla} u)$ and $f(\boldsymbol{x}) = \boldsymbol{\nabla} \cdot (\mathsf{K}(\boldsymbol{x}, u)\boldsymbol{\nabla} u)$ and N_K, N_u, N_D, N_N and N_c are the number of K measurements, u measurements, Dirichlet boundary points, Neumann boundary points and collocation points respectively. However, note that this method requires the measurement of the field K at certain points. This condition was relaxed by Huang et al. (2020), although at the cost of losing some interpretability of the constitutive relation.

In solid mechanics, a recent work by Flaschel et al. (2021) explores the discovery of constitutive equations without the need of data in the state space $(\boldsymbol{u}, \boldsymbol{v}) \in \mathcal{S}$, that is, using only measurable variables (this is the reason why the authors use the word unsupervised). This method is illustrated in Fig. 2.10. It incorporates the ideas from Sparse Identification to build a library dependent on the strain field values, from which the stress field is recovered.

In Fig. 2.11 we place the most important discussed methods in terms of their predictive/explanatory capacity and their use of data/physics.

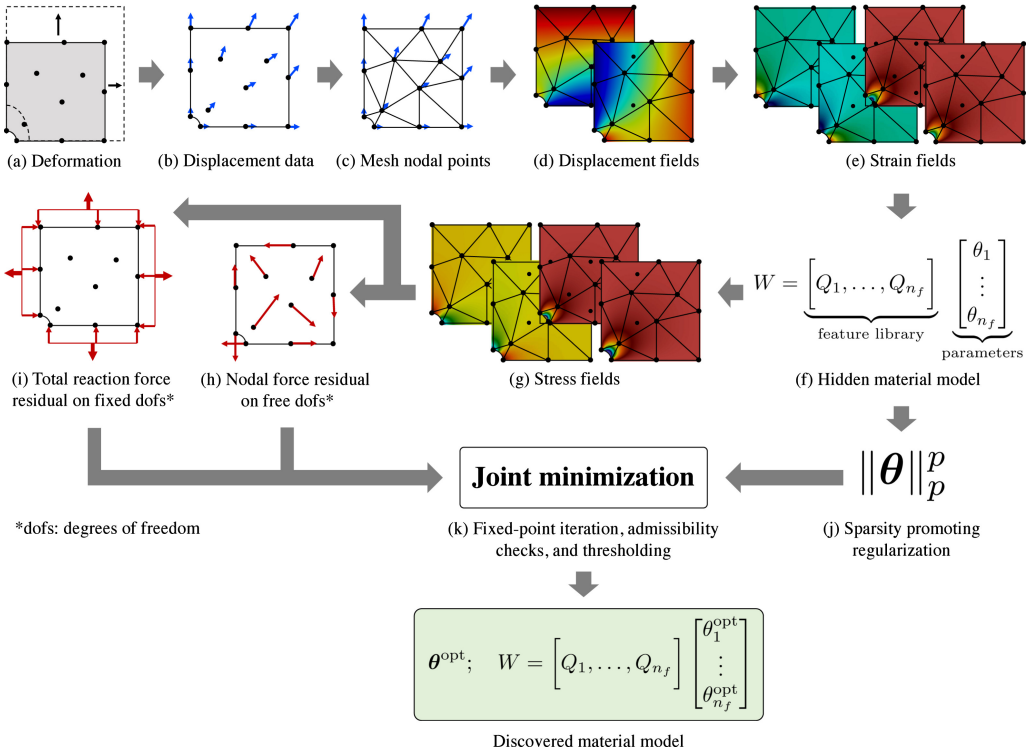


Figure 2.10: Constitutive equations identification. Starting from a deformed body under loading (a), the point-wise displacement measurements (b) are interpolated by constructing a FE mesh (c). The resulting continuous displacement field (d) is differentiated to obtain the strain field (e). The material strain energy density Ψ is expressed as a combination of nonlinear features (f). The stress field is computed as the derivative of the strain energy density with respect to the strain field (g). The joint optimization problem (k) is formulated to find the unknown material parameters such that the weak form of the linear momentum balance in the bulk material (h) and the reaction force balance on the Dirichlet boundaries (i) are satisfied. The joint optimization problem (k) also includes a sparsity-promoting regularization (j) to yield a parsimonious and interpretable material model. Source: Flaschel et al. (2021).

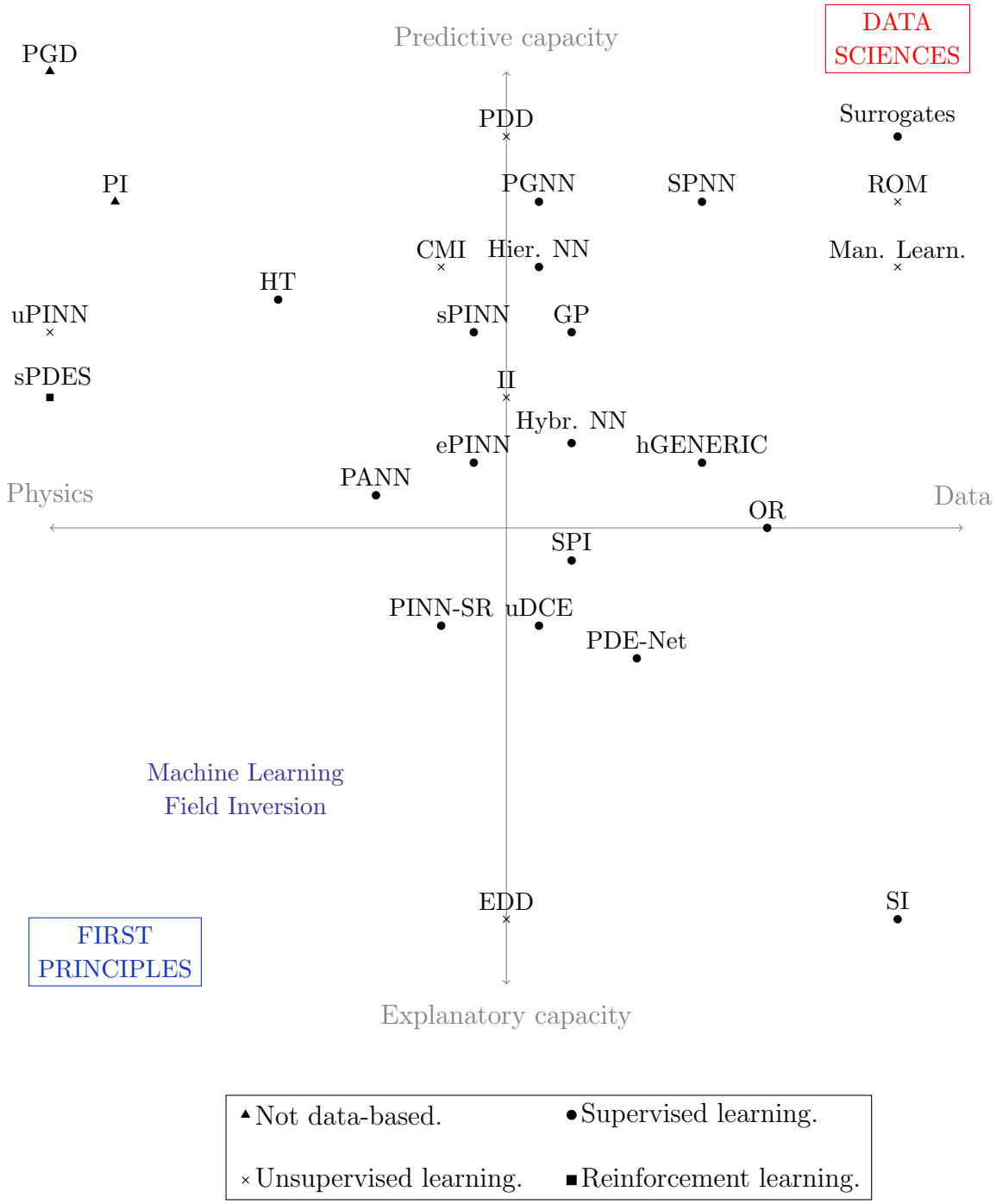


Figure 2.11: Graphical summary of the presented methods. Post ROM: *a posteriori* ROM; Man. Learn: Manifold learning; Hier. N. N.: Hierarchical neural networks; Hybr. N. N.: Hybrid neural networks; P. A. N.N.: Physics-Aware neural networks and related; SPNN: Structure Preserving neural networks; LR: Loss regularisation; PGNN: Physically-Guided neural Networks; uPINN: Unsupervised physics-informed neural networks; sPINN: Supervised physics-informed neural networks; PI: Physics inclusion using constraints; SI.: Sparse Identification; PDE E.: PDE explanation; ePINN: explanatory physics-informed neural network; PINN-SR: Physics-Informed neural network with sparse regression; PDE-Net: Networks for unravelling PDE operators; OR: Operator reconstruction; hGENERIC: hybrid GENERIC; PDD: Predictive DD; EDD: Explanatory DD; II: Instrumental interpolation; EML.: Explanatory Machine Learning; CMI: Constitutive Manifold Identification; SPI: Spectral Parametric Identification; uDCE: unsupervised discovery of constitutive equations.

2.4 A discussion on weaknesses and strengths

From all the above, it is clear that PIDS methods have been intensively exploited in the last years. As seen, many approaches are possible, with a vast amount of strategies for incorporating Physics, different levels of knowledge, a wide variety of algorithms and ML tools, different computational requirements, and different levels of data needs, both in terms of data quality and quantity.

Next, we present a brief summary of these aspects, including the most relevant works among the presented in the literature review:

- **Predictive capacity vs explanatory capacity:** It is clear that the best models in terms of their predictive capacity are *a posteriori* surrogate models, as they are built for such purpose. Particularly, ANNs, due to the universal approximation theorem, are able to predict any physical relation linking data. However, these methods have a total lack of explanatory capacity, except if the ML parameters, $\boldsymbol{\lambda}$ are linked with the problem physics, in one way or another (e.g. for GP used for learning PDEs). This is also true for the methods that exploit the structure of a given physical relation or a given PDE, such as unsupervised PINNs, SPNNs or all the methods assuming a specific PDE, or their expression or stochastic process associated.

  Conversely, the methods assuming no structure at all for the equations have total explanatory capacity, but cannot discriminate between universal laws and constitutive equations, as they learn the physical structure as a whole. Their lack of predictive capacity is explained when we want to predict the state or evolution of the system for another situation that was not represented by the data. For instance, this is the case of sparse identification: if we change some equation parameters (for example the value of σ or β of the Lorentz attractor of Fig. 2.7), the method would not be able to reproduce new trajectories, except if the learning process is restarted for new data.

  In between these two situations, there is a broad range of possibilities, ranging from parametric explanatory capacity (supervised PINNs, hybrid neural networks, FSA and ASA techniques...), to non-parametric discovery of constitutive equations (manifold learning for discovering or reducing constitutive equations, sparse promoting regularisation at the state space...). In particular, a method balancing very well explanatory and predictive capacity is PDE-Net, as they are able to unveil the differential operators and field nonlinearities, and also to predict the field solution for any boundary condition. However, this operator description may be non unique, and also, PDE-Nets cannot afford a parametrisation of the involved operators. The work by Chen et al. (2021) is maybe the one that balances best predictive and explanatory capacity. Finally, and as a special case, DD techniques must focus their performance on prediction (Kirchdoerfer and Ortiz, 2016b) or explanation (Leygue et al., 2018), but not on both at the same time.

- **Stability:** All the methods including regularisation are the best in terms of stability considerations. As explained, regularisation is done by including penalty parameters at the loss function. This could be done for pure mathematical

purposes, looking for sparsity as a way to reduce the model parameters (Tikhonov regularisation, use of sparse norms), or using the physical knowledge that we have about the problem (PINNs, SPNNs). Another kind of regularisation is the one induced by data feedback. In that sense, DDDAS, which are continuously corrected and adapted to the data, are also very robust

On the other side, full models, such as deep ANNs, when used as surrogates of evolution equations, present very often stability problems. Indeed, this is the reason for having many more theoretical results for ML methods adapted to parabolic PDEs, for instance. ANNs are universal approximators, but this character entails the need of large models, compromising long term stability.

ROMs are a middle ground between sparse promotion and full models, at the cost of losing some accuracy in the predictions, so they are good candidates when stability is desired.

- **Computational requirements:** For supervised approaches, when discussing the computational requirements, it is important to distinguish between the learning (or offline) step and the evaluation (or online) step. The learning step essentially depends on two aspects: structural approach and number of parameters. With respect to the structural approach, since the development of back-propagation techniques, compositional function approximation (the one followed by ANNs) has surpassed additive function approximation (for instance the one followed by sparse identification) in terms of computational cost (Han et al., 2018). Therefore, for models with a high number of parameters, the best is to select an ANN framework. Conversely, sparse identification[7] when applied to high dimensional PDEs becomes a hard combinatorial problem that is strongly affected by the curse of dimensionality. Halfway between the two options is the separability assumption, a strategy in which PGD is the paradigmatic example. As it is well-known (Chinesta et al., 2013c), under some hypotheses, PGD is merely an *a priori* POD constructor so it reflects this intermediate point between additive and compositional approaches. Regarding the number of parameters, there is no much to explain, large models (for instance, DL models), are expensive, whereas sparse operators[8], autoencoders or hierarchical models are less expensive. The evaluation step is usually not limiting for any ML model, although it also depends on the number of parameters and function evaluations. A way to better assess the computational cost of the evaluation step is to appraise if the method is able to incorporate extra parameters as input variables (e.g. PGD, RBM, Hybrid twins or other DDDAS approaches...).

  In regards to unsupervised approaches, the computational cost is directly related to the learning algorithm. Our aim is not to include here a detailed discussion, but just as an illustration, regression approaches such as PCA or kPCA are cheaper when compared with methods that explore the data connectivity, such as TDA or k-neighbours search, characteristic of DD approaches.

[7] And of course conventional numerical methods, such as FEM, FVM, finite difference method (FDM) or spectral techniques, but here we are restricted to ML solutions.

[8] By sparse operators we refer to the ones that are defined as so, not to those obtained via sparse regularisation.

- **Data needs:** The data needs of PIDS methods are directly related to two elements. First, to the learning space, which is sometimes equal to the number of parameters and sometimes equal to the latent space dimension. In this sense, heavy models, such as DL models, are very data demanding, whereas *a priori* ROMs are, with respect to data needs, like conventional solvers, so they have no data needs at all. Second, to the number of exogenous constraints (or regularisation terms). In that sense, the methods completely including the problem physics, such as unsupervised PINNs, the Deep Galerkin method, *a priori* ROM methods, or the methods that try to represent the structure of a given PDE whose aim is to overcome the curse of dimensionality, have no data needs at all. Conversely, surrogate models, where no physics is included beyond the one contained in the data, are very data demanding. Of course, there is a wide range of possibilities between these two special cases. The parametric physical explanatory capacity (supervised PINNs, hybrid ANNs, GP for parametric PDEs) is a not very demanding intermediate solution. Methods such as SPNNs, including only some very fundamental equations (degeneracy conditions), are a little more data demanding, and finally DDDAS are quite data demanding but not as demanding as complete surrogate models.
- **Data quality:** Another important aspect about the PIDS methods is their performance when using low-quality data. By low-quality data we mean, essentially, incomplete data, data with a low coverage and noisy data.

  As expected, DD approaches are the most sensitive to noise, particularly when instrumental interpolation is used. Conversely, regression (particularly kernel regression) has a greater filtering capacity. Moreover, Tikhonov regularisation may be interpreted as a whitening filter of the data using a variance-covariance matrix (Bishop, 1995), or equivalently, as the use of Mahalanobis distance (De Maesschalck et al., 2000). Therefore, all the methods including this kind of regularisation are even less sensitive to data noise. Physics-based regularisation acts also as a filter, except that it has a systematic bias, in the sense that we always obtain results closer to the problem physics. In that sense, the bias induced by physical regularisation is the result of *a priori* human bias relying on his knowledge, and therefore, might be desirable or not, depending on the context.

  Regarding data coverage, DD approaches are again the most sensitive to this issue. Indeed, DD methods have no extrapolation capacity at all. The extrapolation capacity of the rest of the methods strongly depends on their capacity to generalise, that is, to avoid over-fitting. In that sense, interpolation-based models and DL models are the most vulnerable to over-fitting, although for DL there are a lot of techniques including *ad-hoc* regularisation, such as dropout (Srivastava et al., 2014), for avoiding this issue.

  Finally, data completeness is chiefly an instrumental problem, as all the methods relative to PIDS assume a complete data-set. Indeed, data completeness issues are addressed by changing the nature of the input or output variables, more than by using imputation methods, that do not incorporate the problem physics.
- **Observability:** A last remark about PIDS methods is data observability. All PIDS methods incorporate a data-set, except the ones where the whole physics

is prescribed. In that sense, we have distinguished previously between observable and non-measurable variables that are the result of a given field theory or framework. The great majority of the presented methods are based on measurable fields (SPNNs, PINNs, sparse identification, hybrid networks, GP for parametric PDEs, ROMs, ...) or assume that there exists a method for sampling internal non-measurable variables at the state space (DD methods, the work by Tartakovsky et al. (2018), manifold learning approaches to constitutive manifold constructions ...). Only in recent years, a few methods combining sparse identification and PDEs (Flaschel et al., 2021) have overcome this difficulty.

Table 2.1 presents a qualitative summary of the previous discussion. The symbols have to be interpreted qualitatively, and by comparison with other methods or when considering a particular strength or weakness of the method, as it is not easy to really compare all the methods with a common benchmark problem and it is of course out of the scope of this thesis.

		Performance			Var. acc.		Comp.	Data	
		P.C.	E.C.	Stab.	M.	N. M.	cost	Quantity	Quality
Post ROM	Linear	+	-	+	Y	N	+	+	+
	Nonlinear	+++	-	+	Y	N	++	++	+
	Man. Learn.		+	+	Y	N	++	+++	+
Surrogates	ANN	++	-	-	Y	N	+	++	+
	DL	+++	-	-	Y	N	++	+++	+
	GP	+++	+	+	Y	N	++	+++	+
Prior ROM	PGD	+	-	+++	Y	N	-	-	-
SPML	Hier. N. N.	++	+	+	Y	N	+	++	+
	Hybr. N. N.	++	++	+	Y	N	+	++	+
	P. A. N. N.	++	+	+	Y	Y	+	+	+
	SPNN	++	+	++	Y	N	+	++	+
Specific PDEs		++	-	+	Y	N	-	-	-
LR	PGNN	++	+	++	Y	N	+	++	+
	uPINN	++	-	+	Y	N	+	-	-
	sPINN	++	+	+	Y	N	+	++	+
PI		+++	-	++	Y	N	+	-	-
SI		-	+++	+++	Y	N	++	++	++
PDE. E	ePINN	++	+	+	Y	N	++	+	+
	PINN-SR	+++	++	+	Y	N	+++	+	+
	PDE-Net	++	++	-	Y	Y	++	++	++
DDDAS	OR	+	-	++	Y	N	+	++	+
	hGENERIC	+	+	++	Y	N	+	+	+
DD	PDD	+++	-	+	Y	X	+++	+++	++
	EDD	-	+++	+	X	Y	+++	+++	++
	II	+	+	+	Y	X	++	+	++
EML	CMI	+++	++	+	Y	Y	+	+	+
	SPI	+++	+	+	Y	Y	++	++	+
	uDCE	+++	++	+	Y	Y	++	++	+

Table 2.1: Summary of the method comparison. Post ROM: *a posteriori* ROM; Man. Learn: Manifold learning; Hier. N. N.: Hierarchical neural networks; Hybr. N. N.: Hybrid neural networks; P. A. N.N.: Physics-Aware neural networks and related; SPNN: Structure Preserving neural networks; LR: Loss regularisation; PGNN: Physically-Guided neural Networks; uPINN: Unsupervised physics-informed neural networks; sPINN: Supervised physics-informed neural networks; PI: Physics inclusion using constraints; SI.: Sparse Identification; PDE E.: PDE explanation; ePINN: explanatory physics-informed neural network; PINN-SR: Physics-Informed neural network with sparse regression; PDE-Net: Networks for unravelling PDE operators; OR: Operator reconstruction; hGENERIC: hybrid GENERIC; PDD: Predictive DD; EDD: Explanatory DD; II: Instrumental interpolation; EML.: Explanatory Machine Learning; CMI: Constitutive Manifold Identification; SPI: Spectral Parametric Identification; uDCE: unsupervised discovery of constitutive equations. P. C.: Predictive capacity; E. C.: Explanatory capacity; Stab.: Stability; Var. acc.: Variable access; M.: Measurable variable: N. M.: Non-measurable variable; Comp. cost: Computational cost.
Performance: Poor (-); Good (+); High (++); Excellent (+++).
Variable access: Yes (Y); No (N); Does not apply (X).
Comp. Cost: No extra cost (-); Low (+); Expensive (++); Very expensive (+++).
Data quantity (data needs): None (-); Moderate (+); High (++); Very high (+++).
Data quality: None (-); Not required (+); Required (++).

3. Reliability-Based Data-Driven solver

Contents

3.1 Introduction

In Chapter 2, we have exposed how Data Sciences, and in particular machine learning (ML) and artificial intelligence (AI) are impacting science. It is clear that Data Analytics techniques can be very useful in simulation-based engineering and science (SBES) to extrapolate the phenomenological submodel, but now constrained with the mathematical expression of first principles. This approach, of increasing importance, is known as data-driven simulation-based engineering and sciencess (DDSBESs). As commented, removing the physical constraints recovers the Data Science and ML framework, while total model specification recovers classical SBESs.

One major research trend in that direction is data-driven (DD), where calculations could be carried out directly from experimental data-sets of real behaviours, taking into account pertinent constraints, such as equilibrium and thermodynamics. This way, the standard phenomenological state modelling is bypassed, based on intelligent and long-term proved, but otherwise simplified, hypotheses. These hypotheses have different accuracy and even validity depending on the particular context of application. Actually, all linear and non-linear phenomenological state models are formulated in terms of parametric mathematical equations, where the variables of interest are forced to remain within a given pre-established manifold. This manifold is derived from observation and experience (empiricism), eventually by means of a trial-error fitting procedure.

One idea in this direction, although limited to a very particular context in which data were acquired and parameterised, was introduced by Sussman and Bathe (2009) for hyperelastic isotropic materials using splines interpolation in what is now known as the what you prescribe is what you get (WYPIWYG) philosophy. This model was extended to anisotropic materials (Latorre and Montáns, 2013, 2014). However, despite the apparent model-free character of this approach, it is no more than a reparameterisation of the standard phenomenological approach in a higher-dimensional parametric space. F. Chinesta and coworkers defined a strategy for DD computational mechanics (CM) (Ibanez et al., 2018), combining Manifold Learning techniques and a (possibly optimised) directional search strategy inspired by the LaTin method (Ladevèze, 1989). M. Ortiz and his group presented a material model-free method based on the minimisation of the distance between the searched solution and a set of experimental data, using a proper energy norm, while remaining in the equilibrium manifold, or equivalently, a well-posed penalty approach (Kirchdoerfer and Ortiz, 2016b). This latter method may be seen as a particular case of the former, with a fixed and constant search direction, given by a projection operator, provided that the energy norm is selected. Other DD hybrid approaches use Gaussian Processes (GP) in a given data-set for dynamical feedback of the parametrical model (Mohanty et al., 2007), or fusion prognosis (Cheng and Pecht, 2009; Xing et al., 2011), to combine DD and physical models, but these methodologies remain encapsulated in the underlying specific physical phenomenological model. Finally, Rajan (2005) and Breneman et al. (2013) have presented some ML tools as support-vector machine (SVM) and basic (linear) ML techniques such as principal component analysis (PCA) that are combined with fundamental principles for a goal-oriented material design.

None of these works takes into consideration the inherent inaccuracy of the data. Conversely, empirical data are considered as error-free for WYPIWYG, directional search and penalty approaches. Only for the penalty approaches, some mathematical convergence results are derived if uncertainty approaches zero, which in practice is never the case. In this paper, a new family of methods, called reliability-based data-driven (RBDD) solvers, based on a metric accounting for uncertainty, is defined, and some mathematical results are derived. It is highlighted how DD solver methodology naturally allows incorporating reliability at the statement of the modelling. With this, DD simulations become sensitive to measurements precision and incorporate uncertainty considerations.

Throughout this chapter, we will discuss the main problems of the classical and the new simulation-based techniques when dealing with noisy data, highlighting the limitations of each methodology. An easy but illustrative one-dimensional problem is used to compare results and to show improvements using this methodology. We also present a second more realistic example with actual experimental data of concrete, emphasising the implications of not having an explicit set of well-defined hypotheses and the corresponding material model. This presents the problem of the importance of the context in which data were obtained and of the particular applications to solve, as discussed in the Discussion and Conclusions section.

3.2 Reliability-Based Data-Driven solvers

3.2.1 Contextualisation

Following Kirchdoerfer and Ortiz (2016b) and Ibanez et al. (2018), DD solvers may be seen as iterative solvers searching for the intersection of a (data-based) empirical manifold $\mathcal{D}$ and a physical manifold $\mathcal{P}$. The first one is in many practical applications experimentally based and has, therefore, a discrete nature. We will denote it as $\mathcal{D} = \{D_i\}_{i=1,\cdots,N}$. The second is usually established in terms of the particular conservation laws of the problem in hands, but otherwise derived from first principles universally accepted as the basis of Physics. Let us consider a certain physical problem defined by a set of partial differential equations (PDEs) that can be split into two main groups, universal physical laws and constitutive equations, as explained in Chapter 2:

$$\mathcal{F}(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{f}) = \boldsymbol{0}, \tag{3.1a}$$

$$\mathcal{H}(\boldsymbol{u}, \boldsymbol{v}) = \boldsymbol{0}. \tag{3.1b}$$

Additionally, Eqs. (3.1a) and (3.1b) must be completed with appropriate initial and/or boundary conditions to make the problem well posed. These extra equations will be denoted as:

$$\mathcal{G}(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{g}) = \boldsymbol{0}. \tag{3.1c}$$

Eqs. (3.1) are the general representation of any physical problem in Continuum Physics. For instance, let us consider the elastic solid problem under static conditions. In that case, we have:

- The unknown fields are $\boldsymbol{u} = \boldsymbol{U}$, the displacement field, and $\boldsymbol{v} = \boldsymbol{\sigma}$, the stress tensor.
- The known fields are $\boldsymbol{f} = \rho\boldsymbol{b}$, the unit volume external forces and $\boldsymbol{g} = (\bar{\boldsymbol{t}}, \bar{\boldsymbol{U}})$ the prescribed traction forces and prescribed displacements at the boundaries.
- The universal law considered is the equilibrium equation at the domain Ω, $\mathcal{F}(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{f}) = \mathrm{div}(\boldsymbol{\sigma}) + \rho\boldsymbol{b}$.
- Splitting the boundary of the domain $\partial\Omega = \Gamma_D \cup \Gamma_N$ in the regions where Dirichlet or Neumann boundary conditions are applied, $\mathcal{G}(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{g}) = (\boldsymbol{U}|_{\Gamma_D} - \bar{\boldsymbol{U}}, \boldsymbol{\sigma} \cdot \boldsymbol{n}|_{\Gamma_N} - \bar{\boldsymbol{t}})$, being $\boldsymbol{n}$ the outward normal vector.

- The constitutive equation is given by $\mathcal{H}(\boldsymbol{u}, \boldsymbol{v}) = \boldsymbol{\sigma} - \boldsymbol{C} : \text{symgrad}(\boldsymbol{U})$, where $\boldsymbol{C}$ is the 4^{th} order stiffness tensor that depends on the considered material.

The discrete version of Eqs. (3.1) is:

$$\boldsymbol{F}(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{f}) = \boldsymbol{0}, \tag{3.2a}$$

$$\boldsymbol{H}(\boldsymbol{u}, \boldsymbol{v}) = \boldsymbol{0}, \tag{3.2b}$$

$$\boldsymbol{G}(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{g}) = \boldsymbol{0}, \tag{3.2c}$$

where now $\boldsymbol{u}$, $\boldsymbol{v}$, $\boldsymbol{f}$ and $\boldsymbol{g}$ are vector (nodal or elemental) representations of the fields and $\boldsymbol{F}, \boldsymbol{G}$ and $\boldsymbol{H}$ are now functions instead of functionals.

Actually, if we assume the relation given by $\boldsymbol{H}$ as known, the physical manifold is defined as:

$$\mathcal{P} = \{(\boldsymbol{u}, \boldsymbol{v}) \in \mathbb{R}^M | \boldsymbol{F}(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{f}) = \boldsymbol{0}, \boldsymbol{G}(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{g}) = \boldsymbol{0}, \boldsymbol{H}(\boldsymbol{u}, \boldsymbol{v}) = \boldsymbol{0}\}. \tag{3.3}$$

However, if we do not have this level of knowledge, the physical manifold is defined as:

$$\mathcal{P} = \{(\boldsymbol{u}, \boldsymbol{v}) \in \mathbb{R}^M | \boldsymbol{F}(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{f}) = \boldsymbol{0}, \boldsymbol{G}(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{g}) = \boldsymbol{0}\}. \tag{3.4}$$

In other words, the physical manifold is the set of states that verify universal conservation laws (for instance, equilibrium and compatibility for the solid continuum mechanics problem). It is possible to add extra knowledge to the system by specifying totally or partially the nature of $\boldsymbol{H}$. We denote by $\boldsymbol{R}$ the combination of $\boldsymbol{F}$, $\boldsymbol{G}$ and the known information about the state model $\boldsymbol{H}$. Therefore, the physical manifold is:

$$\mathcal{P} = \{(\boldsymbol{u}, \boldsymbol{v}) \in \mathbb{R}^M | \boldsymbol{R}(\boldsymbol{u}, \boldsymbol{v}; \boldsymbol{f}, \boldsymbol{g}) = \boldsymbol{0}\}. \tag{3.5}$$

The approach followed in this chapter is to replace the function $\boldsymbol{H}$ by sampled data on the space $(\boldsymbol{u}, \boldsymbol{v})$, something that was proposed by Kirchdoerfer and Ortiz (2016b) for the solid mechanics problem, but with the inclusion of the data uncertainty in the computations. Therefore, our empirical data-set is $\mathcal{D} = \{(\boldsymbol{u}^j, \boldsymbol{v}^j) | j = 1, \ldots, N\}$. This approach is known as DD. Denoting by $P = (\boldsymbol{u}, \boldsymbol{v})$ a DD solver consists basically in computing:

$$P^* = \underset{P \in \mathcal{P}}{\text{argmin}} d(P, \mathcal{D}). \tag{3.6}$$

The key factor here is how to define the distance d. A deterministic distance translates into standard DD solvers, whereas a stochastic distance gives us RBDD solvers.

To fix ideas and simplify the exposition, let us consider the CM problem under static conditions and with no external body forces, whose conservation law is the equilibrium equation:

$$\boldsymbol{\nabla} \cdot \boldsymbol{\sigma} = \boldsymbol{0} \tag{3.7}$$

with $\boldsymbol{\sigma}$ the stress tensor. Eq. (3.7) is usually approximated and solved in a discrete form using numerical methods like finite elements (FE). In that case, after a convenient discretisation we can state:

$$\boldsymbol{B}\boldsymbol{v} = \boldsymbol{0} \tag{3.8}$$

where $\boldsymbol{v}$ is a finite-dimensional vector containing the full stress tensor field information related to a given discretisation (for finite element method (FEM), this vector contains the components of the stress tensor for all the integration points) and $\boldsymbol{B}$ is a matrix encoding the geometry and connectivity of the domain. Once the problem boundary conditions are applied, Eq. (3.8) has to be replaced by

$$\boldsymbol{B}\boldsymbol{v} = \boldsymbol{c} \tag{3.9}$$

for accounting for the forces acting on the system. Eq. (3.9) represents the fundamental Physics of the problem (a discretised version of the equilibrium equation), so this is the universal constraint that will be incorporated to the Data Science approach, according to the physically-informed data science (PIDS) philosophy.

If $\boldsymbol{u} = \boldsymbol{\varepsilon}$ are values representing the different strain states, the empirical manifold is defined via a set $\mathcal{D} = \{(\boldsymbol{u}^j; \boldsymbol{v}^j)\}_{j=1,\cdots,N}$ of data points, resulting from experimental measurements (and therefore not uncertainty free) as it will be illustrated subsequently. The set $\mathcal{D}$ may be seen as a representation of the underlying material behaviour in the following asymptotic sense: (i) $\mathcal{D}$ approximates a topological manifold and (ii) uncertainty of each point approximates to zero. Some basic mathematical results related to these considerations have been already established (Kirchdoerfer and Ortiz, 2016b).

Section 3.2.2 lays down data-driven simulation-based engineering problems in a general framework using optimisation with constraints under a given metric. Next, Section 3.2.3 introduces the RBDD solvers.

3.2.2 Problem formulation

We present here the general framework for DDSBES. With this aim, we postulate that a model-free engineering problem may be defined in terms of state variables $(\boldsymbol{u}, \boldsymbol{v})$ that are related through a latent and unknown relationship $\boldsymbol{H}(\boldsymbol{u}, \boldsymbol{v}) = 0$. Instead of the perfect knowledge of $\boldsymbol{H}$, the information about the nature of this state equation is commonly presented via a measurement data-set $\mathcal{D}$, in a discrete manner, such as $(\boldsymbol{u}^i, \boldsymbol{v}^i)$, where $\boldsymbol{u}^i$ and $\boldsymbol{v}^i$ are vectors whose dimension n is the size of the state vector. Returning to the elastic problem, $\boldsymbol{u}$ is the vector containing all strain components of the whole domain ($\varepsilon_{kl}[\mathsf{J}]$, $k, l = 1, \ldots, d$, with d the spatial dimension of the problem and J referring to a spatial discretisation) and $\boldsymbol{v}$ the vector containing all stress components ($\varepsilon_{kl}[\mathsf{J}]$). It is now necessary to define a distance (a metric) in the state space $\mathcal{S} = \{(\boldsymbol{u}, \boldsymbol{v}) | \boldsymbol{u} \in \mathcal{U}, \boldsymbol{v} \in \mathcal{V}\} = \mathcal{U} \times \mathcal{V}$ that we will assume as Euclidean (if not, the space can be embedded in an Euclidean one thanks to the Whitney immersion theorem, under very general regularity conditions (Whitney, 1944)). That is, for example, for $\boldsymbol{u} \in \mathcal{U}$, to define a symmetric and positive-definite matrix $\boldsymbol{M}_{\mathcal{U}}$ and:

$$\|\boldsymbol{u}\|^2 = \frac{1}{2}\boldsymbol{u}^\intercal \boldsymbol{M}_{\mathcal{U}} \boldsymbol{u}. \tag{3.10}$$

Therefore:

$$d^2_{\mathcal{U}}(\boldsymbol{u}, \boldsymbol{u}') = \|\boldsymbol{u} - \boldsymbol{u}'\|^2_{\mathcal{U}}, \tag{3.11}$$

and we can proceed analogously for $\boldsymbol{v} \in \mathcal{V}$.

As we are considering engineering problems, we have physical constraints. As an example, let us go back to the CM problem. We endow the space of the strain tensors with a norm by defining $\|\boldsymbol{\varepsilon}\| = \frac{1}{2}\boldsymbol{\varepsilon} : \boldsymbol{M}_\varepsilon : \boldsymbol{\varepsilon}$, where $\boldsymbol{M}_\varepsilon$ is a fourth-order tensor, and analogously for the space of stresses, $\|\boldsymbol{\sigma}\| = \frac{1}{2}\boldsymbol{\sigma} : \boldsymbol{M}_\sigma : \boldsymbol{\sigma}$. At each spatial point $\boldsymbol{x}$, we have a different pair $(\boldsymbol{\varepsilon}, \boldsymbol{\sigma})$, which in turn implies a different value for each discrete point of our discretisation, namely, Gauss integration point $(\boldsymbol{\varepsilon}[\mathsf{J}], \boldsymbol{\sigma}[\mathsf{J}])$, $\mathsf{J} \in \mathcal{J}$. Therefore, in general, we have a trial set $\mathcal{D}_\mathsf{J}$ for each $\mathsf{J} \in \mathcal{J}$ that may be thought as the result of experimental tests. Of course, all these data-sets coincide if the material is homogeneous and the experiments are exact. Therefore, we may define the global data-set:

$$\mathcal{D} = \prod_{i=1}^{|\mathcal{J}|} \mathcal{D}_i, \tag{3.12}$$

and the global metric:

$$\boldsymbol{M}_\mathcal{U} = \bigoplus_{i=1}^{|\mathcal{J}|} \boldsymbol{M}_\varepsilon, \tag{3.13a}$$

$$\boldsymbol{M}_\mathcal{V} = \bigoplus_{i=1}^{|\mathcal{J}|} \boldsymbol{M}_\sigma, \tag{3.13b}$$

so we recover the ingredients for the abstract framework presented, that is, the data-set $\mathcal{D}$, and the metrics $\boldsymbol{M}_\mathcal{U}$ and $\boldsymbol{M}_\mathcal{V}$. Once this is done, we define a penalty function:

$$\Pi(\boldsymbol{u}, \boldsymbol{v}|\mathcal{D}) = \min_{(\boldsymbol{u}', \boldsymbol{v}') \in \mathcal{D}} \{d^2_\mathcal{U}(\boldsymbol{u}, \boldsymbol{u}') + d^2_\mathcal{V}(\boldsymbol{v}, \boldsymbol{v}')\}. \tag{3.14}$$

It is obvious that the penalty function vanishes for each point in $\mathcal{D}$, $F|_\mathcal{D} = 0$.

Therefore, a data-driven simulation-based problem (DDSBP) is defined by the constrained optimisation problem:

$$\begin{aligned} &\min_{(\boldsymbol{u}, \boldsymbol{v}) \in \mathcal{S}} && \Pi(\boldsymbol{u}, \boldsymbol{v}|\mathcal{D}) \\ &\text{subject to} && \\ & && \boldsymbol{R}(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{f}, \boldsymbol{g}) = \boldsymbol{0}. \end{aligned} \tag{3.15}$$

Note that, if the physical universal constraints in $\boldsymbol{R}$ affecting $\boldsymbol{u}$ and $\boldsymbol{v}$ are not coupled, the problem is formulated as:

$$\begin{aligned} &\min_{(\boldsymbol{u}, \boldsymbol{v}) \in \mathcal{S}} && \Pi(\boldsymbol{u}, \boldsymbol{v}|\mathcal{D}) \\ &\text{subject to} && \\ & && \boldsymbol{R}_1(\boldsymbol{u}, \boldsymbol{f}, \boldsymbol{g}) = \boldsymbol{0}, \\ & && \boldsymbol{R}_2(\boldsymbol{v}, \boldsymbol{f}, \boldsymbol{g}) = \boldsymbol{0}. \end{aligned} \tag{3.16}$$

Example: Infinitesimal solid mechanics

As an important and illustrative example, let's go back to to the CM problem (Eq. 3.15), which takes the form:

$$\begin{aligned} \min_{(\boldsymbol{\varepsilon}_1,\ldots,\boldsymbol{\varepsilon}_{|\mathcal{J}|},\boldsymbol{\sigma}_1,\ldots,\boldsymbol{\sigma}_{|\mathcal{J}|})} \quad & \Pi(\boldsymbol{\varepsilon}_1,\ldots,\boldsymbol{\varepsilon}_{|\mathcal{J}|},\boldsymbol{\sigma}_1,\ldots,\boldsymbol{\sigma}_{|\mathcal{J}|}|\mathcal{D}) \\ \text{subject to} \quad & \\ & \boldsymbol{B}[\boldsymbol{\sigma}_1,\ldots,\boldsymbol{\sigma}_N]^\intercal = \boldsymbol{c}, \end{aligned} \tag{3.17}$$

where $\boldsymbol{B}$ is a matrix encoding the connectivity and geometry of the problem, depending on the particular discretisation, and $\boldsymbol{c}$ are the nodal forces, that are not necessarily 0 because of the boundary conditions.

It is important to note that the state variables $\boldsymbol{u}$, $\boldsymbol{v}$ may be in a lower dimensional space obtained after dimensionality reduction. For example, for the linear elastic 3D problem, $\boldsymbol{\varepsilon}$ and $\boldsymbol{\sigma}$ live in a space of 6 dimensions, so $(\boldsymbol{\varepsilon},\boldsymbol{\sigma})$ has 12 dimensions and therefore $\mathcal{D}$ has dimension $12\times|\mathcal{J}|$. These dimensions may be reduced if additional simplifications are imposed *a priori* onto the material behaviour. For example, if a homogeneous material is considered, this dimension is actually 12 and if the material is isotropic, this dimension is actually 2 since $\boldsymbol{\sigma} = \lambda \mathrm{Tr}(\boldsymbol{\varepsilon}) + 2\mu\boldsymbol{\varepsilon}$, according to Hooke's law, with λ and μ the Lame's constants, directly related to phenomenological parameters E and ν.

The DD solution as the solution of a constrained minimisation problem

Returning to the general formulation. Let us suppose the pair of variables $(\boldsymbol{u},\boldsymbol{v}) \in \mathcal{P}$. As the problem is formulated in the state space $\boldsymbol{u}-\boldsymbol{v}$, we consider both the constraints affecting to $\boldsymbol{u}$ and $\boldsymbol{v}$ independently. For the sake of simplicity, but without any conceptual limitation, we shall consider linear constraints only, so they can be written as[9]:

$$\boldsymbol{A}\boldsymbol{u} = \boldsymbol{a}, \tag{3.18a}$$
$$\boldsymbol{B}\boldsymbol{v} = \boldsymbol{b}, \tag{3.18b}$$

and the problem represented by Eq. (3.15) writes:

$$\begin{aligned} \min_{(\boldsymbol{u},\boldsymbol{v})\in\mathcal{S}} \quad & \Pi(\boldsymbol{u},\boldsymbol{v}|\mathcal{D}) \\ \text{subject to} \quad & \\ & \boldsymbol{A}\boldsymbol{u} = \boldsymbol{a}, \\ & \boldsymbol{B}\boldsymbol{v} = \boldsymbol{b}. \end{aligned} \tag{3.19}$$

[9]Note that, due to the fact that the method acts over the state-space $\boldsymbol{u}-\boldsymbol{v}$, the constraint $\boldsymbol{R}$ is split in two linear constants, expressed by means of $\boldsymbol{A}$ and $\boldsymbol{B}$

Theorem 3.1: Well-posedness of the linear Data-Driven problem

The problem defined by Eq. (3.19) has a unique solution if $\text{rang}(\boldsymbol{A}) = \text{rang}(\boldsymbol{C}) = r$, where r is the number of constraints.

Proof. Let $(\boldsymbol{u}^*, \boldsymbol{v}^*)$ a pair of state variables verifying $F(\boldsymbol{u}, \boldsymbol{v}) = d^2_{\mathcal{U}}(\boldsymbol{u}, \boldsymbol{u}^*) + d^2_{\mathcal{V}}(\boldsymbol{v}, \boldsymbol{v}^*)$ which exists because of $\mathcal{D}$ finiteness. With these definitions, we can write:

$$\Pi(\boldsymbol{u}, \boldsymbol{v}) = \frac{1}{2}(\boldsymbol{u} - \boldsymbol{u}^*)^\intercal \boldsymbol{M}_{\mathcal{U}}(\boldsymbol{u} - \boldsymbol{u}^*) + \frac{1}{2}(\boldsymbol{v} - \boldsymbol{v}^*)^\intercal \boldsymbol{M}_{\mathcal{V}}(\boldsymbol{v} - \boldsymbol{v}^*). \tag{3.20}$$

We define the augmented lagrangian function $\mathcal{L}(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{\lambda}, \boldsymbol{\mu}) = \Pi(\boldsymbol{u}, \boldsymbol{v}) - \boldsymbol{\lambda}^\intercal(\boldsymbol{A}\boldsymbol{x} - \boldsymbol{a}) - \boldsymbol{\mu}^\intercal(\boldsymbol{B}\boldsymbol{y} - \boldsymbol{b})$. Then:

$$\begin{aligned}
\frac{\partial \mathcal{L}}{\partial \boldsymbol{u}} &= \mathrm{M}_{\mathcal{U}}(\boldsymbol{u} - \boldsymbol{u}^*) - \boldsymbol{A}^\intercal \boldsymbol{\lambda}, \\
\frac{\partial \mathcal{L}}{\partial \boldsymbol{v}} &= \mathrm{M}_{\mathcal{V}}(\boldsymbol{v} - \boldsymbol{v}^*) - \boldsymbol{B}^\intercal \boldsymbol{\mu}, \\
\frac{\partial \mathcal{L}}{\partial \boldsymbol{\lambda}} &= \boldsymbol{A}\boldsymbol{x} - \boldsymbol{a}, \\
\frac{\partial \mathcal{L}}{\partial \boldsymbol{\mu}} &= \boldsymbol{B}\boldsymbol{y} - \boldsymbol{b}.
\end{aligned} \tag{3.21}$$

Using Lagrange multipliers theorem: $\frac{\partial \mathcal{L}}{\partial \boldsymbol{x}} = \boldsymbol{0}$, $\frac{\partial \mathcal{L}}{\partial \boldsymbol{y}} = \boldsymbol{0}$, $\frac{\partial \mathcal{L}}{\partial \boldsymbol{\lambda}} = \boldsymbol{0}$ y $\frac{\partial \mathcal{L}}{\partial \boldsymbol{\mu}} = \boldsymbol{0}$, and therefore:

$$\begin{aligned}
\boldsymbol{M}_{\mathcal{U}}\boldsymbol{u} - \boldsymbol{A}^\intercal \boldsymbol{\lambda} &= \boldsymbol{M}_{\mathcal{U}}\boldsymbol{u}^*, \\
\boldsymbol{M}_{\mathcal{V}}\boldsymbol{v} - \boldsymbol{B}^\intercal \boldsymbol{\mu} &= \boldsymbol{M}_{\mathcal{V}}\boldsymbol{v}^*, \\
\boldsymbol{A}\boldsymbol{u} &= \mathbf{a}, \\
\boldsymbol{B}\boldsymbol{v} &= \mathbf{b}.
\end{aligned} \tag{3.22}$$

We define $\boldsymbol{K}$, $\boldsymbol{w}$ and $\boldsymbol{c}$ as:

$$\boldsymbol{K} = \begin{pmatrix} \boldsymbol{M}_{\mathcal{U}} & -\boldsymbol{A}^\intercal & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{A} & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{M}_{\mathcal{V}} & -\boldsymbol{B}^\intercal \\ \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{B} & \boldsymbol{0} \end{pmatrix}, \tag{3.23}$$

$$\boldsymbol{w} = \begin{pmatrix} \boldsymbol{u} \\ \boldsymbol{\lambda} \\ \boldsymbol{v} \\ \boldsymbol{\mu} \end{pmatrix}, \tag{3.24}$$

$$\boldsymbol{c} = \begin{pmatrix} \mathrm{M}_{\boldsymbol{\mathcal{U}}}\boldsymbol{u}^* \\ \boldsymbol{a} \\ \mathrm{M}_{\boldsymbol{\mathcal{V}}}\boldsymbol{v}^* \\ \boldsymbol{b} \end{pmatrix}. \tag{3.25}$$

Then, Eq. (3.22) writes as $\boldsymbol{K}\boldsymbol{w} = \boldsymbol{c}$ and it would have a single solution if and only if $\det(\boldsymbol{K}) \neq 0$. Using block decomposition of the determinant:

$$\det(\boldsymbol{K}) = \det(\boldsymbol{M}_{\mathcal{U}}) \det(\boldsymbol{M}_{\mathcal{V}}) \det(\boldsymbol{A}\boldsymbol{M}_{\mathcal{U}}\boldsymbol{A}^\intercal) \det(\boldsymbol{B}\boldsymbol{M}_{\mathcal{V}}\boldsymbol{B}^\intercal). \tag{3.26}$$

As $\boldsymbol{M}_{\mathcal{U}}$ and $\boldsymbol{M}_{\mathcal{V}}$ are positive definite matrices, $\det(\boldsymbol{K}) \neq 0 \Leftrightarrow \det(\boldsymbol{A}\boldsymbol{M}_{\mathcal{U}}\boldsymbol{A}^{\intercal})\det(\boldsymbol{B}\boldsymbol{M}_{\mathcal{V}}\boldsymbol{B}^{\intercal}) \neq 0 \Leftrightarrow \boldsymbol{A}\boldsymbol{M}_{\mathcal{U}}\boldsymbol{A}^{\intercal}$ and $\boldsymbol{B}\boldsymbol{M}_{\mathcal{V}}\boldsymbol{B}^{\intercal}$ are regular. Finally, if $\mathbf{D}$ is a positive definite matrix, $\operatorname{rang}(\boldsymbol{B}\boldsymbol{D}\boldsymbol{B}^{\intercal}) = \operatorname{rang}(\boldsymbol{B})$, then the regularity condition is equivalent to $\operatorname{rang}(\boldsymbol{A}) = \operatorname{rang}(\boldsymbol{B}) = r$.

□

The reason why the solution may not be unique relies on the (possible) existence of many points $(\boldsymbol{u}^*, \boldsymbol{v}^*)$ on the set minimising the penalty function.

When solving the nonlinear problem (3.15), two steps are required:

- Local search of a minimum of the penalty function F using the nearest neighbour algorithm. This search looks for the most representative datum in the empirical discrete manifold $\mathcal{D}$.
- Global resolution of the linear system $\boldsymbol{K}\boldsymbol{w} = \boldsymbol{c}$. This equation states that the searched points should remain on the physical manifold.

An easy algorithm for DD problem solving is detailed in the Algorithm 1 and is illustrated in Fig. 2.9a:

Algorithm 1 Data-Driven solver

Input: Empirical data-set, characterised in terms of values $(\boldsymbol{u}, \boldsymbol{v}) \in \mathcal{D}$, Gramm matrices $\boldsymbol{M}_{\mathcal{U}}$ and $\boldsymbol{M}_{\mathcal{V}}$, Physical data $\boldsymbol{A}$, $\boldsymbol{a}$, $\boldsymbol{B}$ and $\boldsymbol{b}$.
Initialisation: $(\boldsymbol{u}^*, \boldsymbol{v}^*) \in \mathcal{D}$.
Compute $\boldsymbol{K}$ using Eq. (3.23).
Compute $\boldsymbol{c}$ using Eq. (3.25).
Solve linear equation $\boldsymbol{K}\boldsymbol{w} = \boldsymbol{c}$. ▷ Projection step.
Extract components $\boldsymbol{u}$ and $\boldsymbol{v}$ from $\boldsymbol{w}$.
Compute $(\boldsymbol{u}^c, \boldsymbol{v}^c) \in \mathcal{D}$, nearest sample point to $(\boldsymbol{v}, \boldsymbol{v})$ using Eq. (3.14). ▷ Local search.
while $(\boldsymbol{v}^c, \boldsymbol{v}^c) \neq (\boldsymbol{v}^*, \boldsymbol{v}^*)$ **do**
    $(\boldsymbol{v}^*, \boldsymbol{v}^*) \leftarrow (\boldsymbol{v}^c, \boldsymbol{v}^c)$
    Compute $\boldsymbol{c}$ using Eq. (3.25).
    Solve linear equation $\boldsymbol{K}\boldsymbol{w} = \boldsymbol{c}$. ▷ Projection step.
    Extract components $\boldsymbol{u}$ and $\boldsymbol{v}$ from $\boldsymbol{w}$.
    Compute $(\boldsymbol{u}^c, \boldsymbol{v}^c) \in \mathcal{D}$, nearest sample point to $(\boldsymbol{u}, \boldsymbol{v})$ using Eq. (3.14). ▷ Local search.
end while
Output: Solution point $(\boldsymbol{u}, \boldsymbol{v})$. Closest data point $(\boldsymbol{u}^*, \boldsymbol{v}^*)$.

3.2.3 Reliability-based data-driven solver

Now, each of the pairs $\boldsymbol{p}^j = (\boldsymbol{u}^j, \boldsymbol{v}^j)$, $j = 1, \cdots, N$ is considered to have random nature. To distinguish random and deterministic variables, we will denote the former with capital letters, $\boldsymbol{P}^j = (\boldsymbol{U}^j, \boldsymbol{V}^j)$. Now, we may define the stochastic problem analogous to the deterministic one (3.15) as:

$$\begin{aligned} &\min_{(\boldsymbol{u},\boldsymbol{v})\in\mathcal{S}} && \mathbb{E}\left[\Pi(\boldsymbol{u}, \boldsymbol{v}|\mathcal{D})\right] \\ &\text{subject to} && \\ & && \boldsymbol{R}(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{f}, \boldsymbol{g}) = \mathbf{0}. \end{aligned} \tag{3.27}$$

Note that, with this formulation, the solution candidate $\boldsymbol{p} = (\boldsymbol{u}, \boldsymbol{v})$ is not random, while $D^2_{\mathcal{D}} = F(\boldsymbol{u}, \boldsymbol{v}|\mathcal{D}) = \Pi(\boldsymbol{p}|\mathcal{D})$ is a random variable due to the random nature of

$\mathcal{D}$. We define the analogous certain squared distance, $d^2_{\mathcal{D}}$ obtained by substituting in the penalty function given by Eq. (3.14), $\boldsymbol{u}'$ and $\boldsymbol{v}'$ by by $\mathbb{E}[\boldsymbol{U}']$ and $\mathbb{E}[\boldsymbol{V}']$:

$$\bar{\Pi}(\boldsymbol{u}, \boldsymbol{v}|\mathcal{D}) = \min_{(\boldsymbol{U}', \boldsymbol{V}') \in \mathcal{D}} \{d^2_{\mathcal{U}}(\boldsymbol{u}, \mathbb{E}[\boldsymbol{U}']) + d^2_{\mathcal{V}}(\boldsymbol{v}, \mathbb{E}[\boldsymbol{V}'])\}. \tag{3.28}$$

The problem in Eq. (3.28) is called a Reliability-based data-driven simulation-based problem (RBDDSBP) and a method for solving it is called a RBDD solver.

3.3 Mathematical derivations

We give here some mathematical properties about the RBDDSBP expressed by the problem (3.27).

Proposition 3.1: Second-order properties of minimal distance

Let us denote $\boldsymbol{P}^* = (\boldsymbol{U}^*, \boldsymbol{V}^*)$ the random vector associated with the minimisation of $d^2_{\mathcal{D}}$, i.e, verifying that $\Pi(\boldsymbol{p}|\mathcal{D}) = \Pi(\boldsymbol{u}, \boldsymbol{v}|\mathcal{D}) = d^2_{\mathcal{U}}(\boldsymbol{u}, \mathbb{E}[\boldsymbol{U}^*]) + d^2_{\mathcal{V}}(\boldsymbol{v}, \mathbb{E}[\boldsymbol{V}^*])$ is minimal. If $\boldsymbol{\Sigma}$ is the variance-covariance matrix of $\boldsymbol{P}^*$ and $\boldsymbol{\Omega}$ is the fourth-order moment tensor of $\boldsymbol{p} - \boldsymbol{P}^*$, that is, the tensor defined by $\Omega_{ijkl}(\boldsymbol{p} - \boldsymbol{P}^*) = \mathbb{E}[(p_i - P^*_i)(p_j - P^*_j)(p_k - P^*_k)(p_l - P^*_l)]$, then:

$$\mathbb{E}[D^2_{\mathcal{D}}] = \frac{1}{2}\mathrm{Tr}(\boldsymbol{M}\boldsymbol{\Sigma}) + d^2_{\mathcal{D}}, \tag{3.29}$$

and:

$$\mathrm{Var}(D^2_{\mathcal{D}}) = \frac{1}{4}\boldsymbol{M} : \boldsymbol{\Omega} : \boldsymbol{M} - \left(\frac{1}{2}\mathrm{Tr}(\boldsymbol{M}\boldsymbol{\Sigma}) + d^2_{\mathcal{D}}\right)^2, \tag{3.30}$$

with $\boldsymbol{M} = \boldsymbol{M}_{\mathcal{U}} \oplus \boldsymbol{M}_{\mathcal{V}}$.

Proof. We have:

$$\mathbb{E}[\Pi(\boldsymbol{u}|\mathcal{D})] = \mathbb{E}\left[\frac{1}{2}(\boldsymbol{u} - \boldsymbol{U}^*)^\intercal \boldsymbol{M}_{\mathcal{U}}(\boldsymbol{u} - \boldsymbol{U}^*) + \frac{1}{2}\mathbb{E}[(\boldsymbol{v} - \boldsymbol{V}^*)^\intercal \boldsymbol{M}_{\mathcal{V}}(\boldsymbol{v} - \boldsymbol{V}^*)\right],$$

$$\mathbb{E}[\Pi(\boldsymbol{u}|\mathcal{D})] = \mathbb{E}\left[\frac{1}{2}[\boldsymbol{u} - \boldsymbol{U}^*, \boldsymbol{v} - \boldsymbol{V}^*]^\intercal \left(\boldsymbol{M}_{\mathcal{U}} \oplus \boldsymbol{M}_{\mathcal{V}}\right)[\boldsymbol{u} - \boldsymbol{U}^*, \boldsymbol{v} - \boldsymbol{V}^*]\right],$$

$$\mathbb{E}[\Pi(\boldsymbol{u}|\mathcal{D})] = \mathbb{E}\left[\frac{1}{2}(\boldsymbol{p} - \boldsymbol{P}^*)^\intercal \boldsymbol{M}(\boldsymbol{p} - \boldsymbol{P}^*)\right],$$

$$\mathbb{E}[\Pi(\boldsymbol{u}|\mathcal{D})] = \mathbb{E}\left[(\boldsymbol{p} - \boldsymbol{P}^*)^\intercal \left(\frac{1}{2}\boldsymbol{M}\right)(\boldsymbol{p} - \boldsymbol{P}^*)\right].$$

It is, therefore, possible to define a random quadratic form:

$$Q_{\frac{1}{2}\boldsymbol{M}}(\boldsymbol{p} - \boldsymbol{P}^*) = F(\boldsymbol{p}|\mathcal{D}). \tag{3.31}$$

Then, we have $D^2_{\mathcal{D}} = \Pi(\boldsymbol{p}|\mathcal{D}) = Q_{\frac{1}{2}\boldsymbol{M}}(\boldsymbol{p} - \boldsymbol{P}^*)$, $d^2_{\mathcal{D}} = Q_{\frac{1}{2}\boldsymbol{M}}(\boldsymbol{p} - \mathbb{E}[\boldsymbol{P}^*])$. In Appendix A we show that for a stochastic quadratic form $Q_{\boldsymbol{A}}(\boldsymbol{Z})$ with expected value $\mathbb{E}(\boldsymbol{Z}) = \boldsymbol{\mu}$, variance-covariance matrix $\mathrm{Cov}(\boldsymbol{Z}) = \boldsymbol{\Sigma}$ and fourth-order moment tensor $\mathrm{M}^4(\boldsymbol{Z}) = \boldsymbol{\Upsilon}$, it is possible to write:

$$\mathbb{E}[Q_{\boldsymbol{A}}(\boldsymbol{Z})] = \mathrm{Tr}(\boldsymbol{A}\boldsymbol{\Sigma}) + \boldsymbol{\mu}^\intercal \boldsymbol{A}\boldsymbol{\mu}, \tag{3.32}$$

and:

$$\mathrm{Var}\left(Q_{\boldsymbol{A}}(\boldsymbol{Z})\right) = \boldsymbol{A} : \boldsymbol{\Upsilon} : \boldsymbol{A} - \left(\mathrm{Tr}(\boldsymbol{A}\boldsymbol{\Sigma}) + \boldsymbol{\mu}^{\intercal}\boldsymbol{A}\boldsymbol{\mu}\right)^2. \tag{3.33}$$

Then, using $\boldsymbol{A} = \frac{1}{2}\boldsymbol{M}$ and $\boldsymbol{Z} = \boldsymbol{p} - \boldsymbol{P}^*$, so that $\mathbb{E}[\boldsymbol{Z}] = \boldsymbol{p} - \mathbb{E}[\boldsymbol{P}^*]$, $\mathrm{Var}(\boldsymbol{Z}) = \boldsymbol{\Sigma}$ and $\boldsymbol{\Upsilon} = \boldsymbol{\Omega}$, we obtain:

$$\mathbb{E}[D^2_{\mathcal{D}}] = \frac{1}{2}\mathrm{Tr}(\boldsymbol{M}\boldsymbol{\Sigma}) + d^2_{\mathcal{D}}, \tag{3.34}$$

and:

$$\mathrm{Var}(D^2_{\mathcal{D}}) = \frac{1}{4}\boldsymbol{M} : \boldsymbol{\Omega} : \boldsymbol{M} - \left(\frac{1}{2}\mathrm{Tr}(\boldsymbol{M}\boldsymbol{\Sigma}) + d^2_{\mathcal{D}}\right)^2. \tag{3.35}$$

□

Under normality conditions, we can go a little bit further:

Proposition 3.2: Second-order properties of minimal distance under normality

Assuming that $\boldsymbol{P}^*$ is a multivariate normally distributed random vector, $\boldsymbol{P}^* \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, then:

$$\mathbb{E}[D^2_{\mathcal{D}}] = \frac{1}{2}\mathrm{Tr}(\boldsymbol{M}\boldsymbol{\Sigma}) + d^2_{\mathcal{D}}, \tag{3.36}$$

and:

$$\mathrm{Var}(D^2_{\mathcal{D}}) = \frac{1}{2}\mathrm{Tr}(\boldsymbol{M}\boldsymbol{\Sigma}\boldsymbol{M}\boldsymbol{\Sigma}) + (\boldsymbol{p} - \boldsymbol{\mu})^{\intercal}\boldsymbol{M}\boldsymbol{\Sigma}\boldsymbol{M}(\boldsymbol{p} - \boldsymbol{\mu}). \tag{3.37}$$

Proof. It is again a consequence of the definition of $D^2_{\mathcal{D}} = Q_{\frac{1}{2}\boldsymbol{M}}(\boldsymbol{p} - \boldsymbol{P}^*) = \Pi(\boldsymbol{p}|\mathcal{D})$ and the result for quadratic forms shown in Appendix A, $Q_{\boldsymbol{A}}(\boldsymbol{Z})$, when $\boldsymbol{Z}$ is a multivariate normally distributed random vector $\boldsymbol{Z} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$:

$$\mathrm{Var}(Q_{\boldsymbol{A}}(\boldsymbol{Z})) = 2\mathrm{Tr}(\boldsymbol{A}\boldsymbol{\Sigma}\boldsymbol{A}\boldsymbol{\Sigma}) + 4\boldsymbol{\mu}^{\intercal}\boldsymbol{A}\boldsymbol{\Sigma}\boldsymbol{A}\boldsymbol{\mu}. \tag{3.38}$$

□

It is important to highlight that, when using a RBDD solver for the solution of a problem, the uncertainty of the data plays an important role, as may be seen from the contribution of the uncertainty in the minimisation of $\mathbb{E}[D^2_{\mathcal{D}}]$, via the variance-covariance matrix of data, $\boldsymbol{\Sigma}$. We have to select a suitable norm (equivalently, a matrix $\boldsymbol{M}$) in order to solve the stochastic problem, but the procedure is very similar to the one associated with DD solvers, as we have:

$$\mathbb{E}\left[\Pi(\boldsymbol{u}, \boldsymbol{v}|\mathcal{D})\right] = \bar{\Pi}(\boldsymbol{u}, \boldsymbol{v}|\mathcal{D}) + \frac{1}{2}\mathrm{Tr}\left((\boldsymbol{M}_{\mathcal{U}} \oplus \boldsymbol{M}_{\mathcal{V}})\boldsymbol{\Sigma}\right). \tag{3.39}$$

Indeed, let us consider a RBDDSBP with linear constraints:

$$\begin{aligned} \min_{(\boldsymbol{u},\boldsymbol{v})\in\mathcal{S}} \quad & \mathbb{E}\left[\Pi(\boldsymbol{u}, \boldsymbol{v}|\mathcal{D})\right] \\ \text{subject to} \quad & \\ & \boldsymbol{A}\boldsymbol{u} = \boldsymbol{a}, \\ & \boldsymbol{B}\boldsymbol{v} = \boldsymbol{b}. \end{aligned} \tag{3.40}$$

Therefore, the algorithm adapted for RBDDSBP solving is Algorithm 2.

Algorithm 2 Reliability-Based Data-Driven solver

Input: Empirical data-set, characterised in terms of the expected values $(\bar{\boldsymbol{u}}, \bar{\boldsymbol{v}}) \in \mathcal{D}$ **and** a variance-covariance matrix $\boldsymbol{\Sigma}$, Gramm matrices $\boldsymbol{M}_{\mathcal{U}}$ and $\boldsymbol{M}_{\mathcal{V}}$, Physical data $\boldsymbol{A}$, $\boldsymbol{a}$, $\boldsymbol{B}$ and $\boldsymbol{b}$.
Initialisation: $(\bar{\boldsymbol{u}}^*, \bar{\boldsymbol{v}}^*) \in \mathcal{D}$.
Compute $\boldsymbol{K}$ using Eq. (3.23).
Compute $\boldsymbol{c}$ using Eq. (3.25).
Solve linear equation $\boldsymbol{K}\boldsymbol{w} = \boldsymbol{c}$. ▷ Projection step.
Extract components $\boldsymbol{u}$ and $\boldsymbol{v}$ from $\boldsymbol{w}$.
Compute $(\bar{\boldsymbol{u}}^c, \bar{\boldsymbol{v}}^c) \in \mathcal{D}$, nearest sample point to $(\boldsymbol{v}, \boldsymbol{v})$ using Eq. (3.39). ▷ Local search.
while $(\bar{\boldsymbol{v}}^c, \bar{\boldsymbol{v}}^c) \neq (\bar{\boldsymbol{v}}^*, \bar{\boldsymbol{v}}^*)$ **do**
    $(\bar{\boldsymbol{v}}^*, \bar{\boldsymbol{v}}^*) \leftarrow (\bar{\boldsymbol{v}}^c, \bar{\boldsymbol{v}}^c)$
    Compute $\boldsymbol{c}$ using Eq. (3.25).
    Solve linear equation $\boldsymbol{K}\boldsymbol{w} = \boldsymbol{c}$. ▷ Projection step.
    Extract components $\boldsymbol{u}$ and $\boldsymbol{v}$ from $\boldsymbol{w}$.
    Compute $(\bar{\boldsymbol{u}}^c, \bar{\boldsymbol{v}}^c) \in \mathcal{D}$, nearest sample point to $(\boldsymbol{u}, \boldsymbol{v})$ using Eq. (3.39). ▷ Local search.
end while
Output: Solution point $(\boldsymbol{u}, \boldsymbol{v})$. Closest data point $(\bar{\boldsymbol{u}}^*, \bar{\boldsymbol{v}}^*)$.

3.3.1 Using Mahalanobis distance

A very recommended distance, which has very good properties for dealing with RBDD solvers, is the Mahalanobis distance (De Maesschalck et al., 2000). If $\boldsymbol{p} \in \mathcal{S}$ and $\boldsymbol{Q}$ is a random vector that takes values in $\mathcal{P}$, whose expected value is $\mathbb{E}[\boldsymbol{Q}] = \boldsymbol{\mu}$, and whose variance-covariance matrix is $\mathrm{Cov}(\boldsymbol{Q}) = \boldsymbol{\Sigma}$, the distance between $\boldsymbol{p}$ and $\boldsymbol{Q}$ is defined as:

$$d(\boldsymbol{p}, \boldsymbol{Q}) = \sqrt{(\boldsymbol{p} - \boldsymbol{\mu})^{\intercal} \boldsymbol{\Sigma}^{-1} (\boldsymbol{p} - \boldsymbol{\mu})}. \tag{3.41}$$

This is equivalent to choose as metric matrix $\boldsymbol{M} = 2\boldsymbol{\Sigma}^{-1}$. If we define $\mathcal{N} = \dim(\mathcal{S})$, the expected value of the optimum of the penalty function is then, using Proposition 3.1:

$$\mathbb{E}[D_{\mathcal{D}}^2] = \mathcal{N} + d_{\mathcal{D}}^2, \tag{3.42}$$

and, the variance:

$$\mathrm{Var}(D_{\mathcal{D}}^2) = \boldsymbol{\Sigma}^{-1} : \boldsymbol{\Omega} : \boldsymbol{\Sigma}^{-1} - (\mathcal{N} + d_{\mathcal{D}}^2)^2. \tag{3.43}$$

Also, under normality conditions, using Proposition 3.2 the variance writes:

$$\mathrm{Var}(D_{\mathcal{D}}^2) = 2\mathcal{N} + 4d_{\mathcal{D}}^2. \tag{3.44}$$

Also, under normality conditions, we can state the following:

Proposition 3.3: Squared distance distribution under normality conditions

Let $D_{\mathcal{D}}^2$ the (random) squared optimal distance to $\mathcal{D}$ using Mahalanobis distance and $d_{\mathcal{D}}^2$ the optimal distance of the certain equivalent problem. Assume that $\boldsymbol{P}^*$ is a multivariate normally distributed random vector, $\boldsymbol{P}^* \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, then $D_{\mathcal{D}}^2$

follows a non-central chi-squared distribution with $\mathcal{N}$ degrees of freedom and non-centrality parameter $\lambda = (\boldsymbol{p}-\boldsymbol{\mu})^\intercal\boldsymbol{\Sigma}^{-1}(\boldsymbol{p}-\boldsymbol{\mu})$:

$$D_{\mathcal{D}}^2 \sim \chi^2\left(\mathcal{N}, (\boldsymbol{p}-\boldsymbol{\mu})^\intercal\boldsymbol{\Sigma}^{-1}(\boldsymbol{p}-\boldsymbol{\mu})\right). \tag{3.45}$$

Proof. Given $\boldsymbol{p}$, we have $\boldsymbol{\Delta p} = \boldsymbol{p} - \boldsymbol{P}^* \sim \mathcal{N}(\boldsymbol{p}-\boldsymbol{\mu}, \boldsymbol{\Sigma})$, therefore, $\boldsymbol{\Sigma}^{-1/2}\boldsymbol{\Delta p} \sim \mathcal{N}(\boldsymbol{\Sigma}^{-1/2}(\boldsymbol{p}-\boldsymbol{\mu}), \boldsymbol{I})$. By using the non-central chi-squared distribution definition we get:

$$D_{\mathcal{D}}^2 = \boldsymbol{\Delta p}^\intercal\boldsymbol{\Sigma}^{-1}\boldsymbol{\Delta p} = (\boldsymbol{\Sigma}^{-1/2}\boldsymbol{\Delta p})^\intercal(\boldsymbol{\Sigma}^{-1/2}\boldsymbol{\Delta p}) \sim \chi^2(\mathcal{N}, \lambda), \tag{3.46}$$

where:

$$\lambda = (\boldsymbol{\Sigma}^{-1/2}(\boldsymbol{p}-\boldsymbol{\mu}))^\intercal(\boldsymbol{\Sigma}^{-1/2}(\boldsymbol{p}-\boldsymbol{\mu})) = (\boldsymbol{p}-\boldsymbol{\mu})^\intercal\boldsymbol{\Sigma}^{-1}(\boldsymbol{p}-\boldsymbol{\mu}). \tag{3.47}$$

□

Having some knowledge on the expected value, variance and distributional properties of the optimal distance to the initial data-set, gives us tools for some considerations on the uncertainty. For example, in this sense, low mean values are related to good convergence while low variance implies neighbourhood to certain convergence.

It is important to highlight that, in practical common applications, the expected value and the variance-covariance matrix are not known and must be estimated. This can be easily done using parametric estimation from a data sample of finite size. Thus, the expected value and variance-covariance matrix may be estimated using standard methods, such as the method of moments (MoM) (Hall et al., 2005) or maximum likelihood estimation (MLE) (Eliason, 1993), among others (Beck and Arnold, 1977). Recall that the distributional properties of $D_{\mathcal{D}}^2$ when substituting population parameters by sample estimators could be derived but are out of the scope of this work.

3.4 1D numerical experiments

Now we evaluate the performance of different DD solvers, including the reliability-based one proposed herein. As expected, the main problem of the linearisation approach appears when dealing with irregular (non-smooth) empirical manifolds. This is typical in Physics when working with models that have discontinuities, like in many mechanical problems such as plasticity, damage, fracture and contact problems. A very basic one-dimensional trivial problem exemplifies well these pathologies.

Let us consider a simple uniaxial loaded rod, as schematised in Fig. 3.1, with $F = 100$ kN, $A = a^2 = 200$ cm^2 and $L = 10$ m. This problem may be easily solved through traditional model-based techniques. The solution is based on the combination of three equations. The equilibrium equation states that $\sigma A = F$; the kinematic equation is expressed as $\varepsilon = \frac{u}{L}$. For this problem to be mathematically closed, we need to establish a mathematical relation, i.e. a phenomenological model, relating the internal (state) variable stress, σ, and the measurable variable strain, ε, which is known as the constitutive relation of the material $\sigma = f(\epsilon)$. For instance, in the linear elastic case, $\sigma = E\varepsilon$.

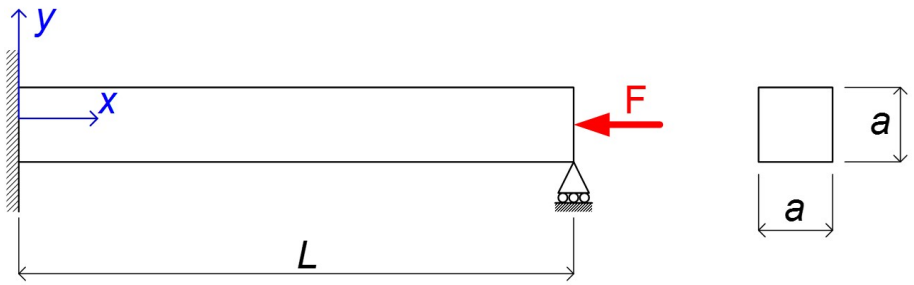


Figure 3.1: Rod under uniaxial load. The variables involved are the rod length L, the applied force F and the cross-section $a \times a$.

Here the approach is different. Let us consider that the constitutive relation is unknown and the material behaviour could be linear, smoothly nonlinear or non-smoothly nonlinear. In any case, to describe the material behaviour we need considerable amount of experimental pair values (ε, σ), $\mathcal{D} = \{(\varepsilon^j; \sigma^j)\}_{j=1,\cdots,N}$. To test DD solvers based on linearisation, let us compare the computed results when considering a non-smoothly nonlinear behaviour and using the well-known iterative tangent Newton-Raphson method.

3.4.1 Solvers comparison

We test the solvers using a constitutive nonlinear equation $\sigma = f(\varepsilon)$ similar to the one of steel. For low strains, the true empirical manifold is almost linear. As the strain increases, we reproduce the steel-like behaviour by considering a hardening plastic behaviour that turns progressively into a clear nonlinear behaviour. It is important to note that this function graph $\sigma = f(\epsilon)$, represents the actual material behaviour obtained in the experiments, whose expression is not known *a priori*. When using linearisation approaches, we need to define, at least locally, a smooth manifold in order to work with tangent spaces. There are many methods for building the underlying manifold from data, ranging from pure interpolation to pure regression, including all Manifold Learning techniques (kernel principal component analysis (kPCA) (Schölkopf et al., 1998), Self Organizing Map (SOM) (Kohonen, 1990), locally linear embedding (LLE) (Roweis and Saul, 2000), Isomap (Tenenbaum et al., 2000), Laplacian Eigenmap (Belkin and Niyogi, 2003), t-distributed stochastic Neighbour embedding (t-SNE) (Maaten and Hinton, 2008) among others (LJP and Van Den, 2007)). As a rule of thumb, the more accurate and structured empirical data-set, the better interpolation-based techniques perform. On the other hand, regression techniques are preferred when dealing with noisy and unstructured data when low dimensional and regular underlying manifolds are desirable.

Besides, sometimes $\mathcal{D}$ can be generated with control in one of the variables (laboratory controlled tests) and sometimes not (for example sensors in dynamic data-driven application system (DDDAS)). The latter case is the most general and challenging. We tested the convergence for these two cases using the typical Newton-Raphson solver.

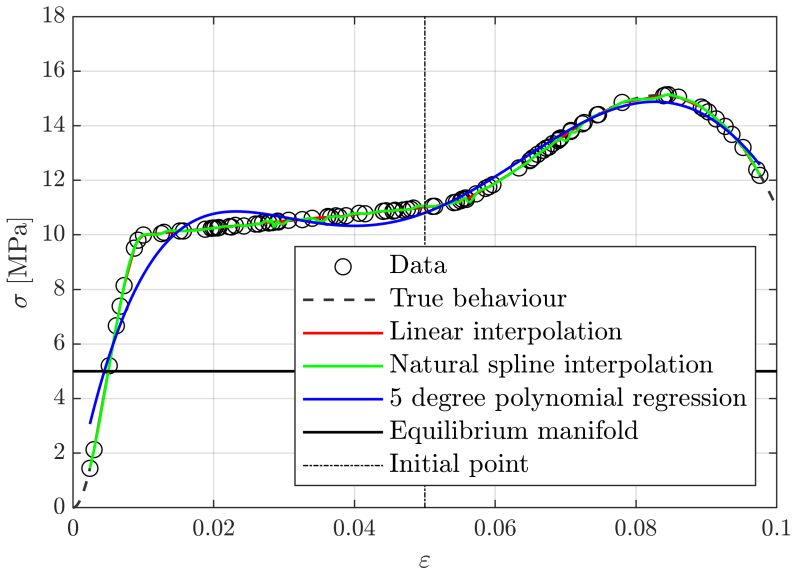
Four analyses were considered, varying the number of data sample points, N, and the error measure related to uncertainty, s. Data generation is as follows: for each $\varepsilon^j \sim \mathcal{U}(0; \varepsilon_{\max})$, $j = 1, \cdots, N$ and, as before, $\sigma^j \sim \mathcal{N}(\mu^j, s)$, with $\mu^j = f(\varepsilon^j)$ and $s = \alpha\sigma_{max}$, where $\varepsilon_{\max}$ and $\sigma_{\max}$ are the maximum strain and $\sigma_{\max}$ maximum stress considered.

We evaluate the different solvers under different conditions:

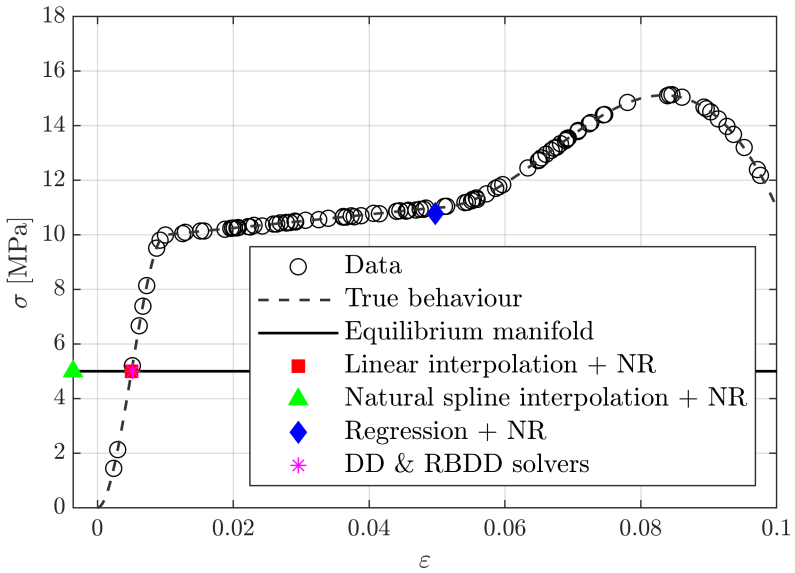
- **Data-set size:** Varying the value of N.
- **Noise level:** Varying the value of α.
- **Empirical manifold regularity:** By considering the equilibrium manifold intersecting the empirical true manifold at two different regimes: the elastic regime, obtained for $\sigma < 10$ MPa and the plastic regime, $\sigma \geq 10$ MPa.

Figs. 3.2, 3.3, 3.4, 3.5, 3.6 show the results of these numerical experiments. Each figure shows the considered empirical set, the equilibrium manifold and the constitutive manifold built with some fitting techniques (linear interpolation, natural spline interpolation and 5 degree polynomial regression). The vertical dashed line shows initial points considered for the Newton-Raphson solver. Then, we show the empirical set, the equilibrium manifold and the final point for each solver. Both RBDD and DD solvers converge to the same point. We can extract the following conclusions:

1. In Fig. 3.2, solvers are based on an empirical set of $N = 100$ pairs $(\varepsilon; \sigma)$ and a low homogeneous uncertainty is considered ($\alpha = 0.001$). Convergence is sometimes not achieved by the Newton-Raphson solver based on regression fit because of the untrue convexity of the built manifold, which is inherent to multiparametric regression. In the case of natural spline regression, the almost linear behaviour in the hardening part of the curve causes bad convergence.
2. In Fig. 3.3, we use an empirical set of $N = 100$ pairs $(\varepsilon; \sigma)$ but higher noise is considered ($\alpha = 0.05$). Even if polynomial regression is not sensitive to noise, convergence is again not achieved because of the untrue convexity of the manifold constructed. Besides, due to noise, natural splines suffer spurious oscillations provoking bad convergence. This can be avoided using linear interpolation, but in this case, non-smoothness of the broken line is incompatible with a tangent-based solver, which in turn results in non-convergence.
3. For reduced sample sizes ($N = 20$), as shown in Fig. 3.4, the regression based solver is again not convergent. Due to the lack of data, the data-driven solver has, in that case, less accuracy than linearisation approaches based on interpolation techniques.
4. In Fig. 3.5, we see the effect of increasing the noise level keeping the data-set small. Even if the regression solver has bad convergence, linear and spline interpolators reach a good solution, improving the performance of DD solver, but this fact is related to the nature and regularity of the empirical manifold.
5. Convergence problems increase dramatically when considering reduced sample sizes ($N = 20$), higher noise ($\alpha = 0.05$) and we approach the plastic regime ($F = 200$ kN), as seen in Fig. 3.5, where only DD solvers converge to an accurate enough solution.

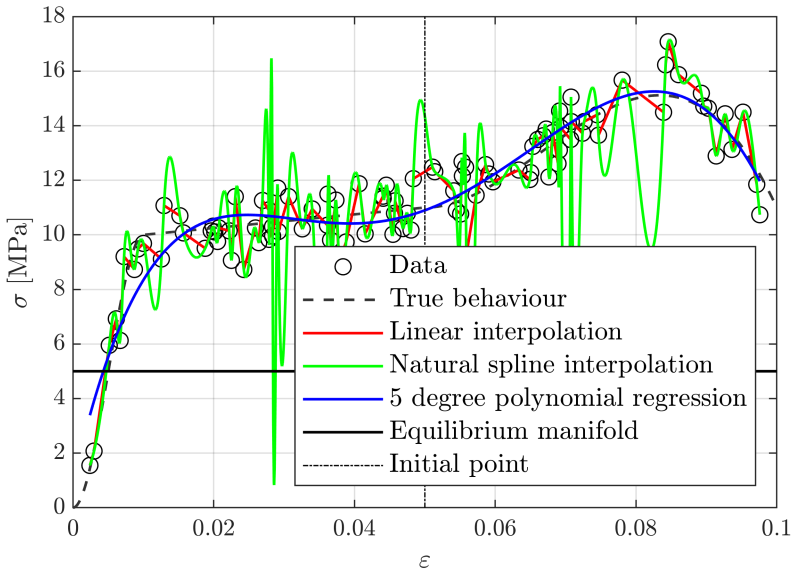


(a) Built manifold for linearisation techniques.

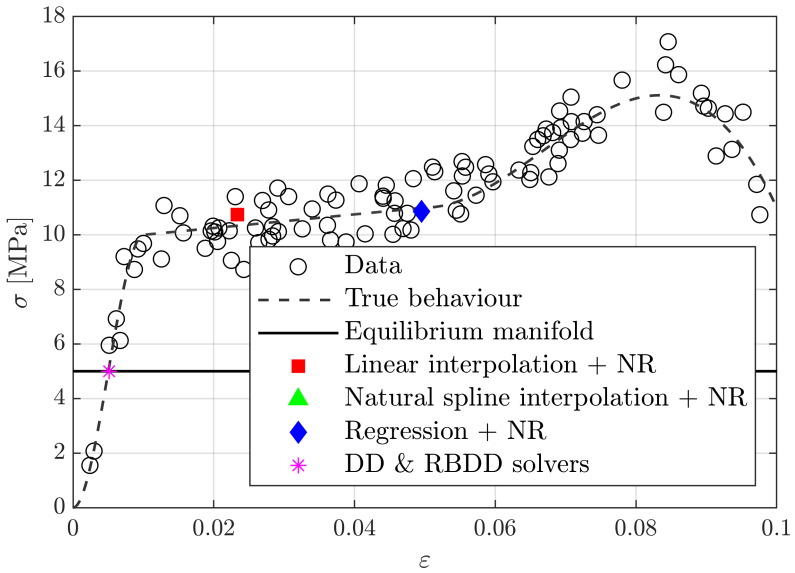


(b) Solution point for different DD solvers.

Figure 3.2: Performance of the different solvers for a large data-set and low noise. Elastic regime. We set $N = 100$, $\alpha = 0.001$ and $F = 100$ kN.

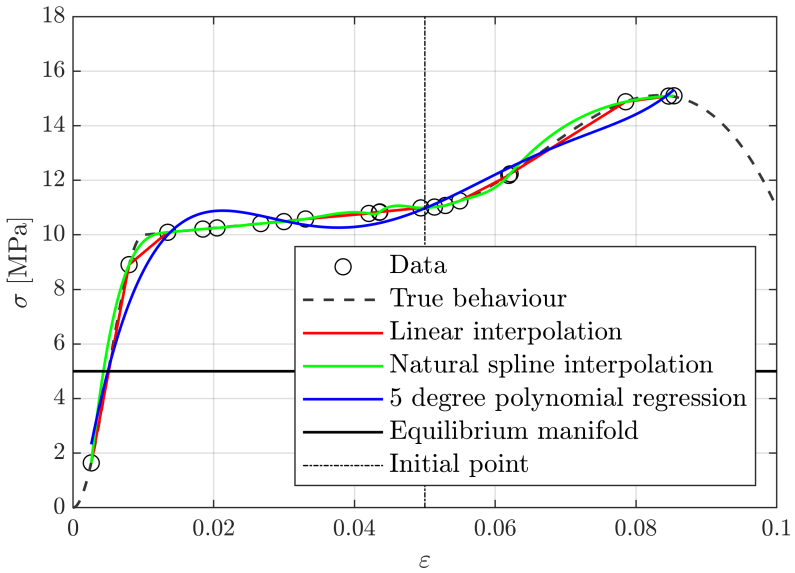


(a) Built manifold for linearisation techniques.

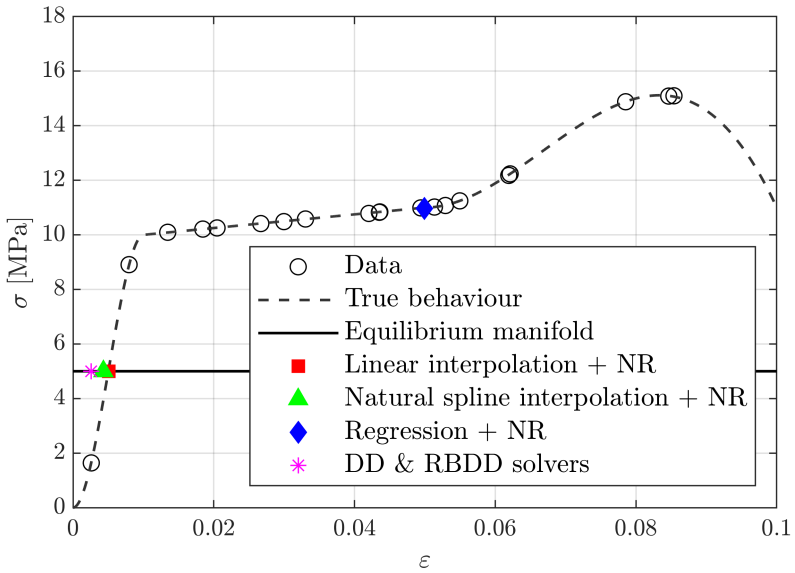


(b) Solution point for different solvers.

Figure 3.3: Performance of the different solvers for a large data-set and high noise. Elastic regime. We set $N = 100$, $\alpha = 0.05$ and $F = 100$ kN.

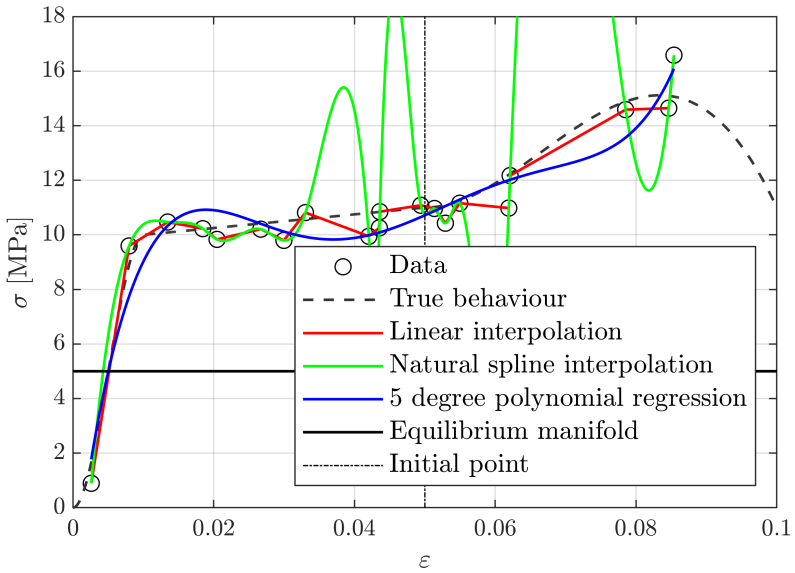


(a) Built manifold for linearisation techniques.

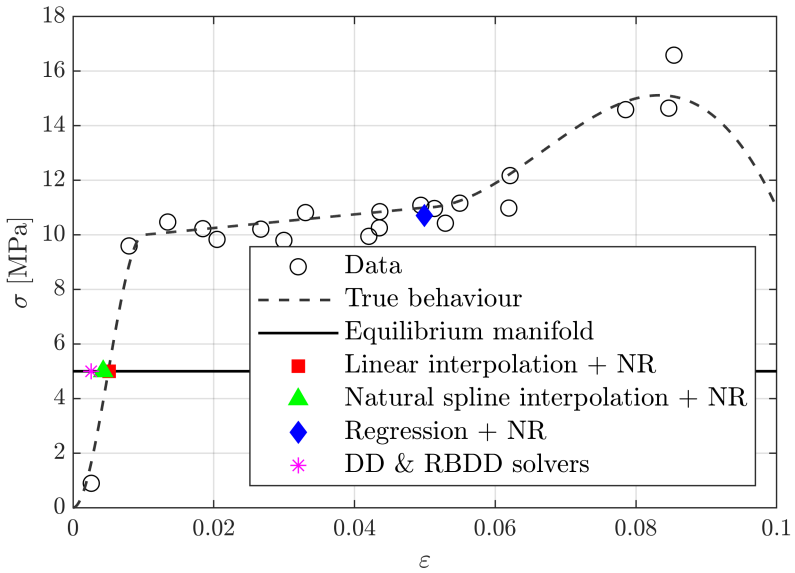


(b) Solution point for different solvers.

Figure 3.4: Performance of different solvers for a small data-set and low noise. Elastic regime We set $N = 20$, $\alpha = 0.001$ and $F = 100$ kN.

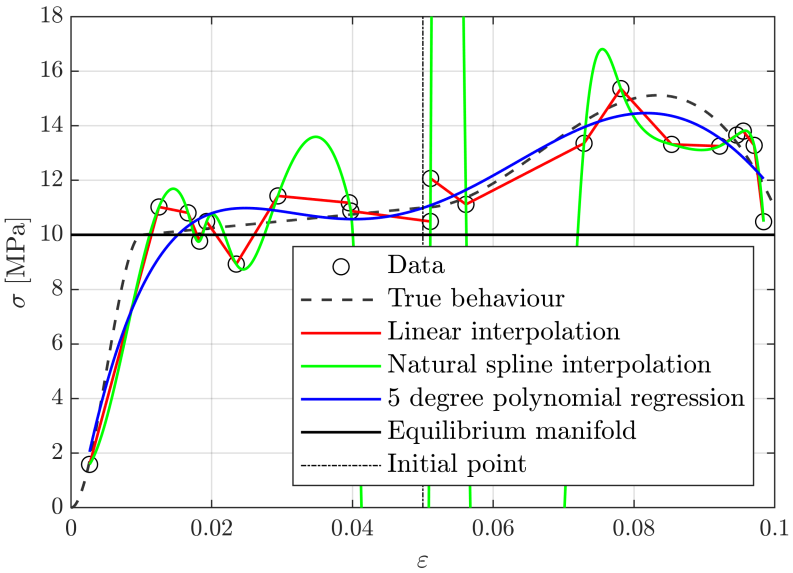


(a) Built manifold for linearisation techniques.

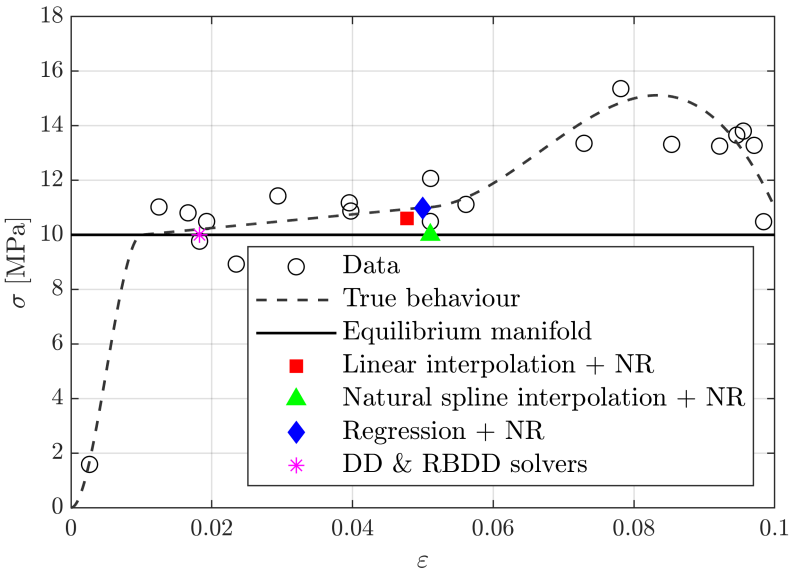


(b) Solution point for different solvers.

Figure 3.5: Performance of different solvers for a small data-set and high noise. We set $N = 20$, $\alpha = 0.05$ and $F = 100$ kN.



(a) Built manifold for linearisation techniques, $F = 200$ kN.



(b) Solution point for different solvers, $F = 200$ kN.

Figure 3.6: Performance of different solvers for a small data-set and high noise. Plastic regime. We set $N = 20$, $\alpha = 0.05$ and $F = 200$ kN).

For homogeneous noise (that is, independent of ε), the DD solver and the RBDD solver give the same result, as pointed out before. Table 3.1 shows the squared distance results for DD solvers for different values of α, F and N. For the DD solver and following Kirchdoerfer and Ortiz (2016b), we have taken as norm:

$$\|(\varepsilon,\sigma)\|^2 = \frac{1}{2}C\varepsilon^2 + \frac{1}{2C}\sigma^2, \tag{3.48}$$

with $C = 300\,\text{MPa}$, whereas for the RBDD solver we adopted the Mahalanobis norm:

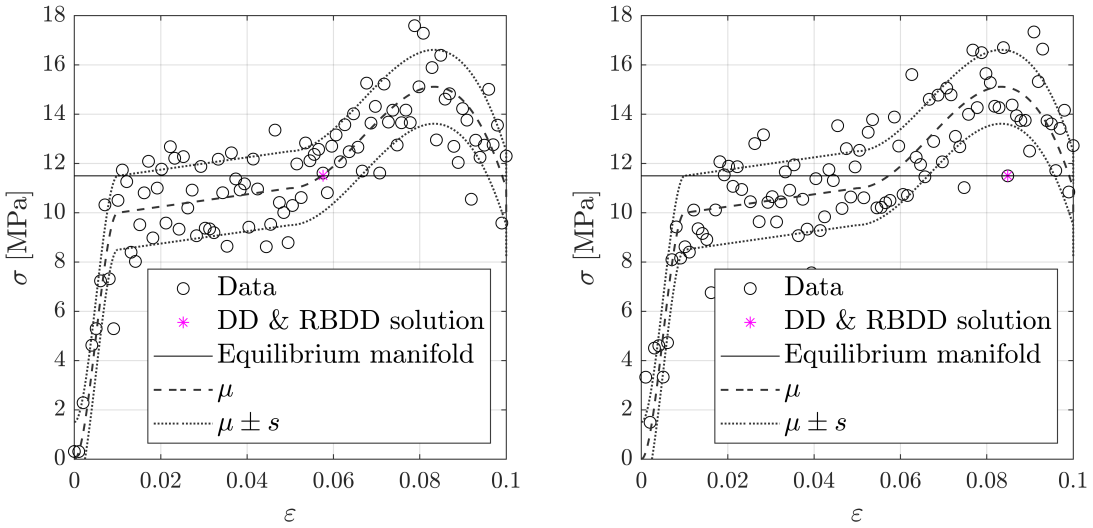
$$\|(\varepsilon,\sigma)\|^2 = \frac{1}{s_\varepsilon^2}\varepsilon^2 + \frac{1}{s_\sigma^2}\sigma^2, \tag{3.49}$$

with $s_\varepsilon = \alpha\varepsilon_{\max}$ and $s_\sigma = \alpha\sigma_{\max}$, so the RBDD solver is sensitive to the uncertainty/noise level. Consequently, it is more informative in the following sense: for $N = 100$, the distance to the empirical set increases when passing from $\alpha = 0.001$ to $\alpha = 0.05$. Each realisation provides a different distance depending only on the empirical set sample. The RBDD solver does not have this problem because it is uncertainty-dependent and can detect when uncertainty is of the order of the optimal distance. Only when $\alpha \to 0$ and the empirical set is almost a subset of the constitutive manifold, this distance can be used as a good uncertainty-free indicator. Otherwise, the locus of the underlying manifold is unknown and there is no way to interpret the DD optimal distance in a coherent manner.

N	100		20		20	
F [kN]	100		100		200	
α	0.001	0.05	0.001	0.05	0.001	0.05
DD solver [Pa]	275	6040	75251	112431	6692	1602
RBDD solver	183.43	1.61	50167.62	29.98	29.59	0.096

Table 3.1: Squared distance results for DD solvers: The squared distance is dimensionless for the RBDD solver and considers the noise level, making it more context-dependent and with an easier interpretation.

To analyse the statistical properties of the squared distance, we consider $\alpha = 0.1$, $N = 100$ and $F = 230$ kN to generate two possible empirical sets with the same statistical properties. Let us suppose that the expected value of the empirical set is known. This should approximate the true underlying manifold but actually it may be estimated from experimental samples. Fig. 3.7 shows the considered data points, expected values μ and an error band, defined by $\mu \pm s$, where s is the standard deviation for two different data samples obtained from the same probabilistic distribution. Table 3.2 shows some statistical properties of the RBDD solver for both cases, assuming normality. Here we consider ε as an uncertainty free variable, and therefore the chi-squared distributed of $D_{\mathcal{E}}^2 = \left(\frac{\sigma-\mu}{s}\right)^2$ has $n = 1$ degree of freedom. The squared distance computed from the data is almost zero in both cases, but a deeper insight onto the empirical set statistics ($\mu = \mu(\varepsilon)$, $s = s(\varepsilon)$) highlights the distance to the true manifold. We can see pathological convergence at Fig. 3.7b for illustration purposes, even if 96 simulations were carried out to obtain such a bad result.



(a) Solution near the empirical manifold. **(b)** Solution far from the empirical manifold.

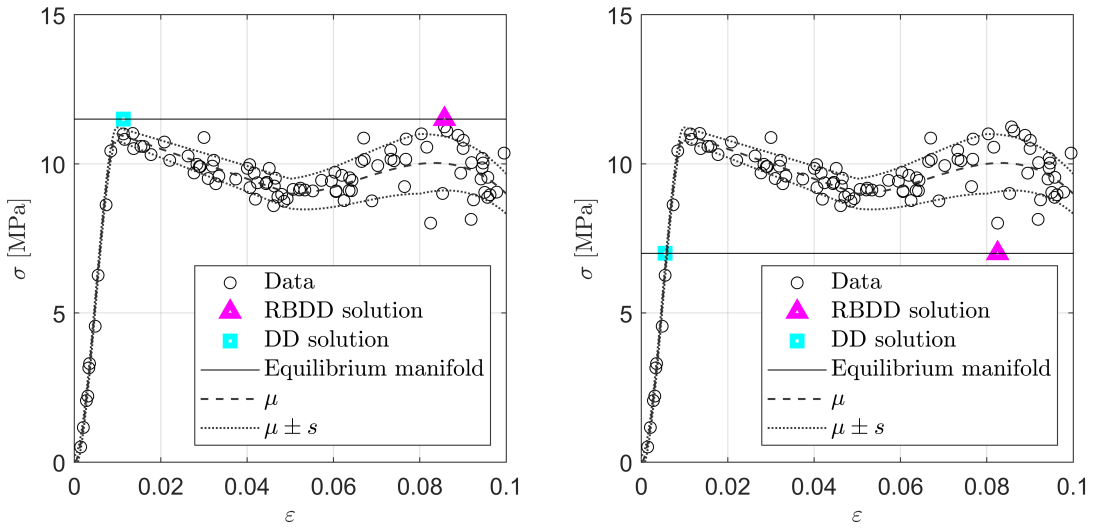
Figure 3.7: Performance of DD and RBDD solvers. We have considered $N = 100$ and $\alpha = 0.1$. Two different samples are shown.

Position w.r.t. manifold	Close (Fig. 3.7a)	Far (Fig. 3.7b)
Optimal squared distance	0.0056	0.0000
Expected value	1.001	6.725
Variance	2.003	24.899
95%-Confident bound	3.84	16.30

Table 3.2: Statistical characteristics of the two solution points. Optimal squared distance, expected value and variance do not depend on the underlying distribution, whereas the confident bound was computed assuming normality.

3.4.2 Uncertainty considerations

The RBDD solver is not only more suitable and informative, but for non-homogeneous uncertainty, it results in a better convergence in the following sense. Fig. 3.8 shows solution points for the DD and RBDD solvers for another *virtual* material, with a different $\sigma = f(\varepsilon)$ constitutive relationship. Note that the uncertainty associated with the actual material behaviour is inhomogeneous: in the elastic zone, where the material is very well characterised, the uncertainty is low, while it increases for higher strains. The RBDD solver is sensitive to this variation, while the DD one is not. For a more complete information, Fig. 3.8 is complemented by the statistical properties summarised in Table 3.3. In Fig. 3.8a, we can see that the DD solver converges to a very unlikely point while the RBDD solver converges to a more likely one. This is due to the smaller ratio between the euclidean distance from the solution point to the empirical data-set point and the bandwidth, for the RBDD solver. However, even though the squared distance is small, the expected value indicates that the RBDD solver has converged to a point not very close to the constitutive manifold. In Fig. 3.8b, a different convergence point is observed. Now, the RBDD solver has an undesirable behaviour because of the lack of data in the linear region, with high certainty. This is detected by means of the expected value and variance, as well as the 95% upper bound. As a first conclusion, the RBDD solvers may be therefore used for sampling strategy. In any case, this is a very unreasonable case, because often, more sample points are associated with less uncertainty.



(a) $F = 230$ kN. **(b)** $F = 140$ kN.

Figure 3.8: Performance of DD and RBDD solvers for heterogeneous uncertainty: The RBDD solver may have pathological convergence, as the DD one, but its associated statistics show this pathology more straightforwardly.

Note that knowledge of the upper confidence bound of the squared distance, $D^2_{\mathcal{D}}$, could be interesting for defining a quantitative criterion for convergence.

	Case	$F = 230$ kN	$F = 140$ kN
DD	Squared distance [Pa]	2472	5414
RBDD	Squared distance	0.08	1.12
	Expected value	3.70	10.98
	Variance	12.82	41.91
	95%-Confident bound	10.82	23.07

Table 3.3: Statistical properties of both solvers for each of the presented cases. Note that the RBDD solver is more informative.

3.5 Application to scale data reduction

An interesting application of the RBDD solver in the domain of CM appears when dealing with several scales. One of the main strategies used when coupling two scales (multiscale approach) is selecting a representative volume element (RVE) and establishing a sound transition procedure between the microscale properties and the macroscale response (Zohdi and Wriggers, 2008). This strategy has allowed setting up implicitly material constitutive relationships that were not known explicitly at the macroscale (Brieu and Devries, 1999; Miehe et al., 1999). Recent works foreground the crucial point of scales decoupling in the averaging process and the need of uncertainty quantification when building the restriction operator (Xu and Chen, 2009). As an answer, many works have incorporated microscale randomness in the multiscale procedure, either using Montecarlo method (MCM) sampling (Kouznetsova et al., 2001; Ma et al., 2014, 2015) or Perturbation Method (Sakata et al., 2010). However, these considerations are still used for model validation and uncertainty has not been incorporated routinely in macroscale computations, except through expensive MCM sampling. A different alternative has been proposed by using stochastic partial differential equations (SPDEs) (Ganapathysubramanian and Zabaras, 2007).

We can apply the RBDD solver for uncertainty propagation from the microscale to the macroscale allowing to incorporate it in macroscale computations. Let us assume that we have at the microscopic scale a (discrete) coupled field $(\boldsymbol{u}^i, \boldsymbol{u}^i)$, $i = 1, \cdots K$. Therefore, classical RVE techniques allow us to define a macroscopic reference value $(\boldsymbol{U}, \boldsymbol{V})$ where $\boldsymbol{X} = \overline{\boldsymbol{x}^i}$ and $\boldsymbol{V} = \overline{\boldsymbol{v}^i}$. For elastic problems, this could be strain-stress pairs $(\boldsymbol{\varepsilon}^i, \boldsymbol{\sigma}^i)$. It is possible to compute the variance-covariance matrix $\boldsymbol{\Sigma}$ of the sample $\boldsymbol{p}^i = (\boldsymbol{u}^i, \boldsymbol{v}^i)$. Geometrically, this means to define a $n + m$-dimensional ellipsoid in the state space $(\boldsymbol{U}, \boldsymbol{V})$ associated with each single macroscopic point, where n and m are the space dimension of state variables $\boldsymbol{u}$ and $\boldsymbol{v}$. Fig. 3.9 shows an ellipsoid in a two-dimensional plane for two possible microstructural fields. Note that accounting only for average values, as done in classical RVE techniques, gives the same result in both of them. We have presented here an approach from the point of view of dimensionality reduction (we use first and second order statistics instead of the whole microscopic field). Besides, this could be exploited in multiscale computational mechanics if a resourceful method allowed second order statistical characterisation of the microscale without the whole microscale fields computation.

An application of this methodology is shown in Fig. 3.10 for the rod problem presented in the previous section. For each macroscopic point (ε, σ), we have the mean strain $\mu(\varepsilon)$, mean stress, $\mu(\sigma)$, strain variance $s^2(\varepsilon)$, stress variance $s^2(\sigma)$ and

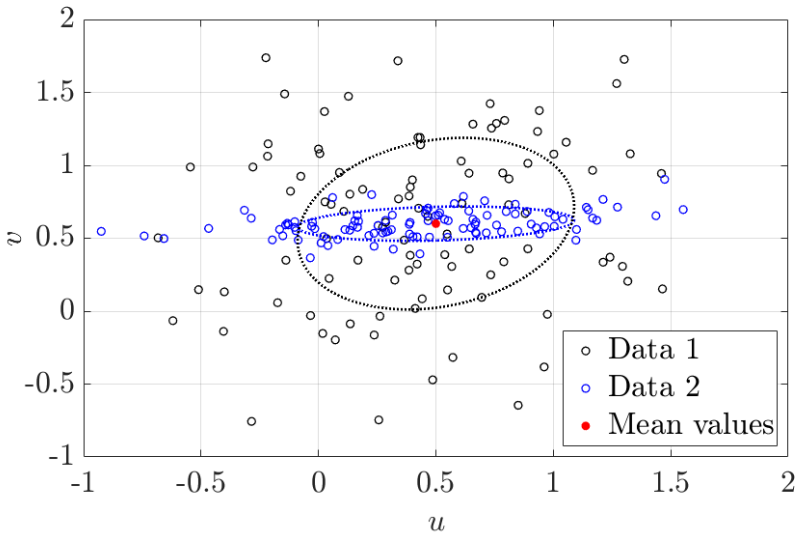


Figure 3.9: Two possible data-sets and associated 2nd order statistics.. The two data-sets have the same mean value but a different uncertainty ellipsoid.

correlation coefficient $\rho(\varepsilon, \sigma)$ computed from data in the microscale. For the sake of simplicity, these data have been randomly generated using a parametric law but should be interpreted as the result of measurements in the lower scale or obtained through more complex multiscale procedures and techniques. In Figure 3.10, the stresses and strains are normalised using $\varepsilon_0 = 0.1$ and $\sigma_0 = 10$ MPa, respectively.

As it can be seen, RBDD methods allow uncertainty propagation, through a second order moment characterisation of the state variables. In other words, geometry of the state space is distorted by means of uncertainty: the solution point is agreed to be the nearest point to a given uncertainty ellipsoid, built from input data or specific computations.

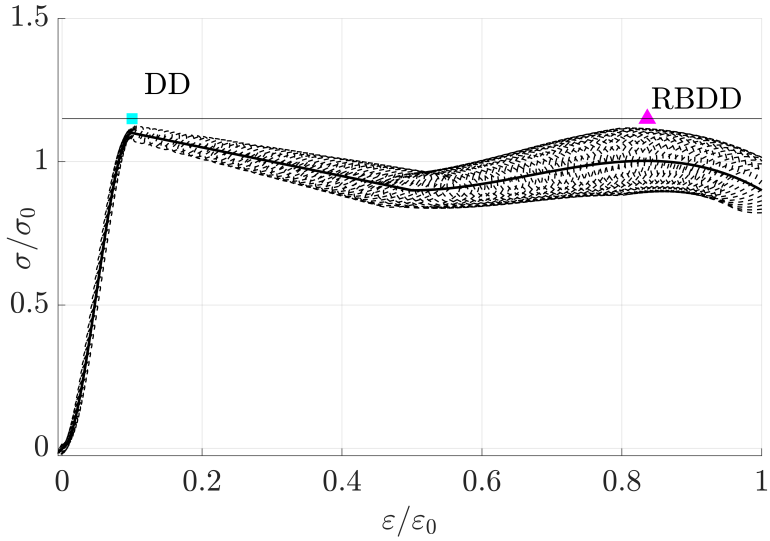


Figure 3.10: Uncertainty propagation at the macroscale. Illustration of a coupling scale strategy using DD and RBDD solvers, with the latter accounting for uncertainty propagation from the microscale to the macroscale.

3.6 A 3D case study

In this section, the potential of the proposed methodology is highlighted in a real application. With this aim, the RBDD solver was applied to a 3D model using actual data of concrete behaviour, thus introducing a more complex level of numerical implementation (now using FEM). Besides, additional hypotheses are required for the practical use of the RBDD methodology. Both issues are introduced next.

Data regarding mechanical characterisation of concrete were obtained from the experimental setup shown in Fig. 3.11. A square mortar concrete specimen of 100 mm size is subjected to a uniaxial stress state by means of two compression plates, as sketched in Fig. 3.11. Concrete includes Portland cement and a calibrated dosage to get an ultimate strength of 40 MPa expected value. Four experimental tests were carried out at a compression rate of 0.015 mm/s with displacement control. Displacement values and loading were recorded up to rupture of the specimen as seen in Fig. 3.11. These values were treated to build a 3D data-set as commented above.

On the other hand, a concrete specimen subjected to a compressive load - reminiscent to the bottom part of a structural column (see Fig. 3.12) was selected as the 3D problem of interest for the application of the RBDD methodology. Three steps of loading were considered in order to check the performance of the solver at different regions of the mechanical behaviour shown in Fig. 3.13.

The 3D numerical RBDD solver implemented herein partially follows the work by Kirchdoerfer and Ortiz (2016b). Briefly, the algorithm proceeds iteratively based on a finite element methodology to search at each Gauss point of every element the closest solution to the material experimental data-set, i.e.

$$(\sigma_I^{k+1}, \sigma_{II}^{k+1}, \sigma_{III}^{k+1}) - (\varepsilon_I^{k+1}, \varepsilon_{II}^{k+1}, \varepsilon_{III}^{k+1}),$$

to:

$$(\sigma_I^{D-k+1}, \sigma_{II}^{D-k+1}, \sigma_{III}^{D-k+1}) - (\varepsilon_I^{D-k+1}, \varepsilon_{II}^{D-k+1}, \varepsilon_{III}^{D-k+1}).$$

The optimality criterion is based on minimising the Mahalanobis distance in Eq. (3.41) and therefore follows strictly the methodology explained in previous sections. For the sake of simplicity and computational cost, the searching algorithm proceeds in the space of principal directions. Convergence is considered to be achieved once $|W| < \text{TOL}$, being TOL a certain (tolerance) value and W a certain criterion defined in this section as follows:

$$W = \sqrt{\frac{1}{s}\|\sigma^{k+1} - \sigma^k\|^2 + \frac{1}{e}\|\varepsilon^{k+1} - \varepsilon^k\|^2}, \tag{3.50}$$

being s and e representative values of the stress and strain ranges in the test data, respectively. The code was implemented in Matlab software.

The stress component along the compression direction is analysed in Figure 3.14 for different regimes (steps) of the strain-strain curve, at two representative points (top and bottom) located at the surface of the specimen (see Figure 3.13). Figure 3.14 also shows the stress-strain level of points 1 and 2 along the data-set as well as the mean and mean $\pm$ standard deviation curves. It is observed that the stress remains in the

linear range at steps 1 and 2. Conversely, point 1 at step 3 falls into the so-called damaged region of the concrete behaviour. It is convenient to note that the DD numerical methodology naturally deals with nonlinear material behaviour without the need of elaborated model-dependent formulations and associated nonlinear solvers, while the RBDD solver turns out to be uncertainty-robust as well. For completeness, Table 3.4 shows the optimal Mahalanobis distance of the obtained solution at points 1 and 2 for the different analysed steps.

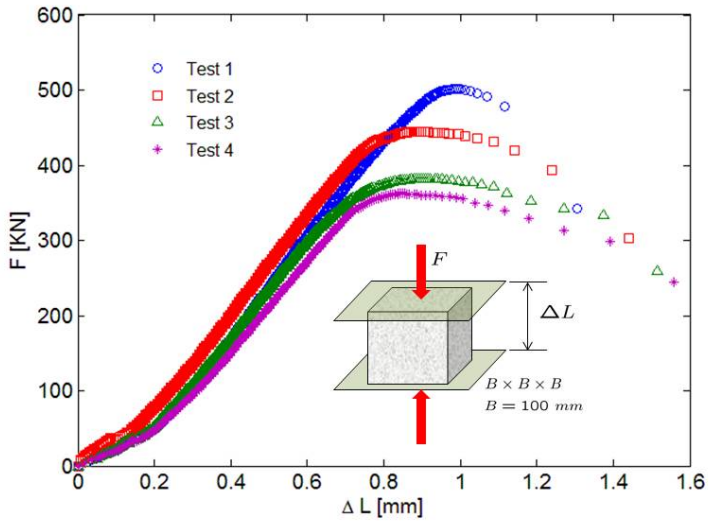


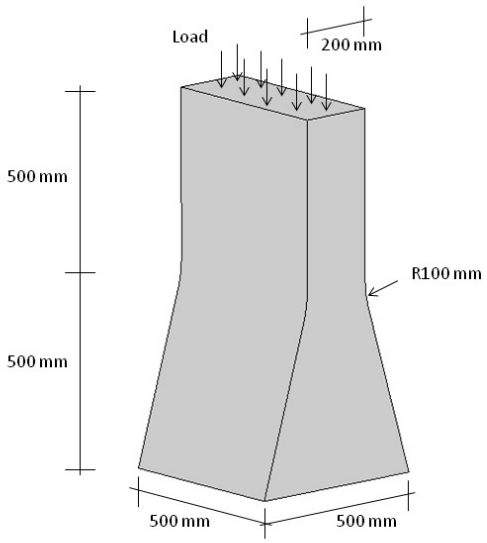
Figure 3.11: Experimental setup and obtained experimental data from four different tests. The different curves come from different uniaxial tests.

Point	1	2
Step 1	$1,44 \cdot 10^{-3}$	$7.93 \cdot 10^{-4}$
Step 2	$2.41 \cdot 10^{-3}$	$3.34 \cdot 10^{-3}$
Step 2	$6.89 \cdot 10^{-3}$	$3.60 \cdot 10^{-3}$

Table 3.4: Results for the two points considered. Mahalanobis optimal distance for RBDD methodology applied to the 3D example of application

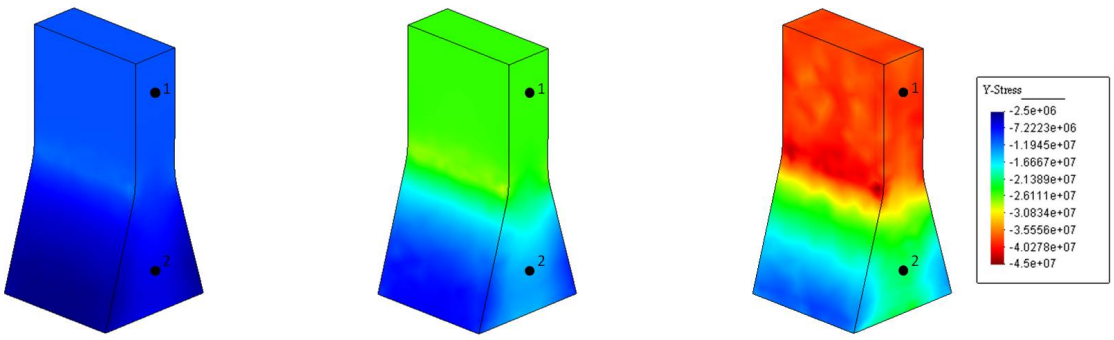
3.7 Discussion and conclusions

In this chapter, a new RBDD solver has been formulated for DDSBES, allowing uncertainty considerations in the input data that are, therefore, not considered as uncertainty-free, but of random nature. The DD simulation problem is defined as a constrained stochastic optimisation problem. Constraints encode all relevant physical information of the system, such as conservation or other physical laws. Calculations are carried out directly from data, avoiding any modelling error via state



Step	Load [Pa]
1	1.0e7
2	2.5e7
3	3.8e7

Figure 3.12: Geometry and dimensions of the concrete test piece used in numerical simulation.



(a) Step 1. **(b)** Step 2. **(c)** Step 3.

Figure 3.13: Results for the stress field. Stress field σ_{yy} obtained using RBDD methodology at the different steps.

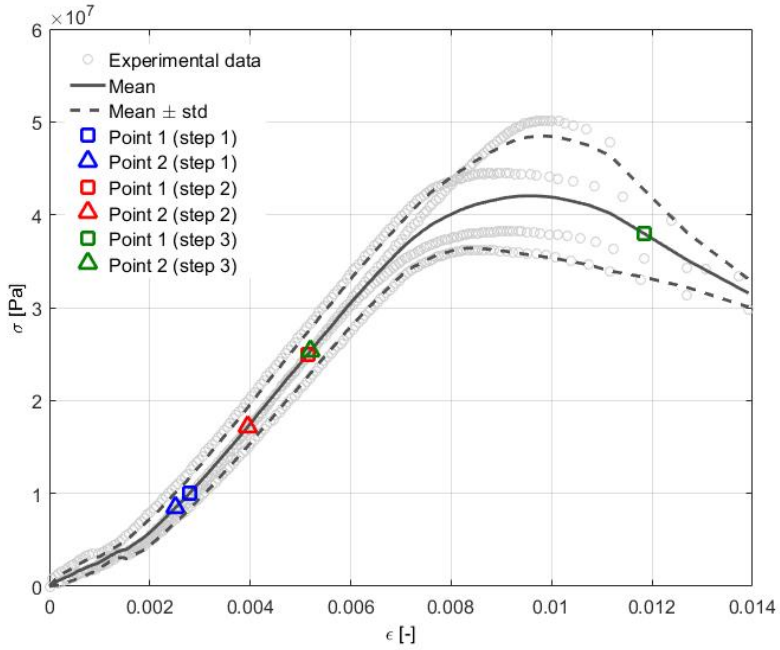


Figure 3.14: Data statistics and results. Experimental data, confidence band at level $\mu \pm s$ and numerical solution at different points and steps.

or constitutive equation assumptions. Optimality is sought in terms of a penalty function showing the distance between a candidate solution point and the input data set.

It has been shown that selecting a proper uncertainty dependent distance, the Mahalanobis distance, results in very good statistical properties as well as easily interpretable optimal distances. Indeed, optimal distance is computed in terms of the data sample and data uncertainty, which allows assessing when the solution point is accurate enough, up to data precision. Moreover, this distance offers the possibility of considering heterogeneous uncertainty, leading to most likely solution points, instead of getting deterministic solution points which may be very sampling-dependent.

Excluding very simple problems with uncertainty-free linear behaviour and small sample sizes, where conventional solvers could have been used, the method here proposed has shown better convergence, higher precision, clearer interpretation, major flexibility and more soundness. Besides, the statistical interpretation, depending on sampling statistics, allows decision-oriented statistical inference.

RBDD solvers appear to be a very suitable tool for facing at least three important problems:

1. **Dynamic data-driven systems**. In these systems, predicting features and learning capabilities are combined. The presented solver could start predictions from scratch, where sample sets are small and the underlying true manifold is unknown. Further knowledge of the analysed system due to the increase of the sample size will feed the RBDD solver, thus allowing faster updates of the solution points and statistical inference. This will, additionally, enhance the possibility to define

different sampling strategies, data coverage and improve solver performance. Moreover, it is sensitive to measurement errors, that are inherent to any experimental test, that depends on equipment and human precision. In this sense, RBDD solvers conform a robust framework that provides coherent results within the experimental context.

2. **Scale dimensionality reduction problems**. From purely theoretical (sound physical reasons) and/or practical (speed-up calculations) points of view, it could be interesting to define the transition from a small scale to a bigger one, defining a hierarchical procedure. The presented RBDD solver is an ideal tool for uncertainty propagation from one scale to another. Moreover, this may be helpful in case of a dimensionality reduction strategy anchored to Big Data frameworks. For instance, if we work with an n-dimensional field and two scales with two mesh sizes N and M, the whole problem will have $N \cdot M \cdot n$ degrees of freedom. Averaging techniques could reduce the problem to a $M \cdot n$ degrees of freedom problem, but all the variability of the lower scale is lost. With the RBDD approach, lower scales variability is conserved, at low computational cost, resulting in $M \cdot \frac{n(n+3)}{2} \sim \theta(Mn^2)$ degrees of freedom. If $n \ll M$, savings are evident.

3. **Model-free engineering based on empirical measurements**. The new presented solver offers the possibility of carrying out simulations directly from data, without explicit model assumptions. However, this raises an important limitation not only of this methodology, but of Data Analytics in general. This corresponds to the need of contextualising the data to be sure that they can be extrapolated to a possibly different context corresponding to the particular application. The possibility (or not) of this extrapolation is analysed explicitly in the standard model-driven approach when making explicit the assumptions that drive to such particular model. For example, in the 3D example presented above (but also extended to other scenarios of common practical use), stress-strain data are available along the direction of the load only. To extrapolate this 1D behaviour to a multiaxial situation some hypotheses have to be made, that are explicitly stated when defining the 3D constitutive model, but are not so explicit (while still necessary) when applying directly the experimental data. The most important are the following:

   (a) The material is considered homogeneous, or at last, with the same level of homogeneity as the experimental sample used.

   (b) A given stress state is associated with a certain strain state regardless of the particular material orientation. This implies that the material is isotropic in average at the microstructural level.

   (c) Only the behaviour in one direction is known, so the stress-strain relation in other directions has to be assumed as equal (again isotropy) while the relation between different directions (e.g. Poisson ratio) has to be assumed and estimated. This is a classical hypothesis made during characterisation of the mechanical behaviour of materials. A value of 0.2 was assumed for concrete in accordance with standard codes of practice.

(d) The measured behaviour (in principal components) is extrapolated to a multiaxial state using (a) and (b). This extrapolation is made using the same sampling interval as in the original data.

This raises again the problem of having enough data to extract all possible situations (point location, direction, level of strain, etc.) in order to have the possibility of accurately extrapolating every condition possible in our particular application. This is rarely the case in reality, so, at least a profound reflection onto the applicability of the data to the particular context and the assumptions it implies is mandatory.

To conclude, RBDD solvers present a meeting point between theoretical sciences, through epistemologic constraints, and experimental sciences, through uncertain real world data. The elegance of the mathematical formulation enables many analyses and theoretical considerations for the whole spectrum of Continuum Physics. The easiness of combining the presented concepts with all trendy Data Science and deep learning (DL) tools opens huge possibilities for facing highly challenging problems.

The need for matching the DD methodology with existing (simplified) experimental setups today available to capture the mechanical behaviour of materials implies the need of making some explicit assumptions or, at least, to think about the context in which the data have been obtained and the one of the application in hand to decide if they can be extrapolated or not, and if the additional data required to fulfil the problems demands are available.

Scientific outcomes
As a result of this work, we published the following scientific articles:

- Ayensa-Jiménez J., Sanz-Herrera, J. A., Doweidar M. H., Doblaré M. "A new reliability-based data-driven approach for noisy experimental data with physical constraints", *Computer Methods in Applied Mechanics and Engineering*, 2018.

4. The missing data problem

Contents

4.1 Introduction

We have seen in Chapter 2 that, in the last years, a new approach to simulation-based engineering and science (SBES) using the power of Data Science methods has been proposed. This approach, of increasing importance and known as data-driven simulation-based engineering and sciences (DDSBES), combines physical constraints and raw data. We can think about DDSBES as an intermediate point between the standard Data Science and machine learning (ML) framework (no physical information) and an *a priori* parametrisation of constitutive laws (classical

SBES). Actually, all linear and nonlinear phenomenological constitutive models can be formulated in terms of parametric mathematical equations, where the variables of interest are forced to remain within a given pre-established manifold. This manifold is derived from observation and experience (empiricism), eventually by means of a trial-error fitting procedure.

One idea in this direction was started by F.Chinesta and coworkers (Ibanez et al., 2018) who defined a strategy for data-driven (DD) computational mechanics (CM), combining manifold learning techniques and a (possibly optimised) directional search strategy inspired in the LaTin method (Ladevèze, 1989). As exposed in Chapter 2, Kirchdoerfer and Ortiz (2016b) presented a model-free method based on the minimisation of the distance between the searched solution and a set of experimental data, using a proper energy norm. The solution was also forced to remain in the equilibrium manifold, by means of a well-posed penalty approach. In Chapter 3, we extended this work to define the so-called reliability-based data-driven (RBDD) solver (Ayensa-Jiménez et al., 2018).

However, the referred works are often restricted to the frame of perfect information, that is, the data-set is complete. In other words, all state variables are assumed to be known for a given measurement. This is not always the case in practical situations, being this one of the main concerns in Data Science (Allison, 2002, 2003; Graham, 2009; Nakai and Ke, 2011; Rubin, 1976; Schafer and Graham, 2002).

There are many methods that have been developed to address this problem, both model-free and model-based. Among the model-free, the most fundamental are listwise deletion (LD) and pairwise deletion (PD) (Graham, 2009), which consist in discarding incomplete data. These methods, however, decrease the statistical power (Peng et al., 2006) and introduce biases (Schafer, 1997) if the so-called missing process is missing not at random (MNAR) (Allison, 2001; Briggs et al., 2003; Rubin, 1976), which is obviously the case when dealing with data obtained from experiments or measurements. Another common approach is single imputation (SI), based on a filling strategy for the missing data that uses values obtained from complete data, for example, mean or mode imputation (Allison, 2002; Schafer and Graham, 2002). However, this technique reduces the variability and weakens the covariance between variables. Another approach is to create dummy variables accounting for the missing data variables (Dummy Variable Adjustment) (Allison, 2002; Schafer and Graham, 2002). This results in biased estimators and is not theoretically based. Finally, it is also possible to replace missing values with a predicted score from a regression equation (Allison, 2002; Schafer and Graham, 2002). This weakens the variance and overestimates model fit and correlation estimates. Moreover, these methods do not take into account the local structure and geometry of the data, which is critical when the data have some underlying physics. In order to solve this problem, interpolation (linear interpolation, nearest interpolation or spline interpolation, (Junninen et al., 2004)) is a common technique. In this approach, only the physics inherent to the data is learned in the imputation process.

Other model-based methods have been developed to deal with the missing data problem, such as Multiple Imputation (Little and Rubin, 2002) using the regression method (Rubin, 1996), the Predictive Mean Matching Method (Schenker and Taylor, 1996) or the Markov chain Montecarlo method (MCMCM) (Schafer, 1997, 1999),

Full Information Maximum-Likelihood (FIML) estimation (Enders, 2001; Enders and Bandalos, 2001; Hartley and Hocking, 1971) and Expectation-maximisation (Dempster et al., 1977). The problem of all these model-based methods is that they assume, in one way or another, a statistical model for the data (e.g. normality). This is usually the case in social and economical sciences (Enders, 2010) but is not the general case for physical problems, where variables follow some fundamental laws that do not follow normality or other distributional assumptions.

The amount of missing data is not, however, the sole criterion to assess the quality of the available data, especially if they correspond to a problem that relies on some physical laws (Tabachnick et al., 2001). Accordingly, our aim in this work is to establish a framework in which both the local structure of the data and the supplementary physics, not explicitly included in the data structure, are used to improve the imputation procedure. In this context, the imputation method can be compatible with any DDSBES method. The presented technique is based on the mean concept and, therefore, could be interpreted as a generalisation of the Mean Imputation Method. On the one hand, the local structure of the empirical data-set is preserved since the data are forced to belong to specific manifolds, which depend on the problem nature. On the other hand, the underlying physics of the problem is imposed via supplementary constraints on the data. The imputation procedure is then performed by using an unsupervised learning algorithm that finds the point that minimises an, in general context-dependent, weighted quadratic error, while preserving the local and global physics of the problem.

4.2 Mathematical formulation

4.2.1 General framework

Our aim is to present a methodology that fits within the context of DD problems. In particular, problems in which the governing equations of the system are fundamental laws of Physics and the modelling strategy is replaced by data, that, in general, may be incomplete. This means that the computations will be carried out directly from data without an *a priori* parametrisation step of the state equations (e.g. constitutive model in Continuum Mechanics).

Let us go back to our general (discretised) framework:

$$\boldsymbol{F}(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{f}) = \boldsymbol{0}, \tag{4.1a}$$

$$\boldsymbol{H}(\boldsymbol{u}, \boldsymbol{v}) = \boldsymbol{0}, \tag{4.1b}$$

$$\boldsymbol{G}(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{g}) = \boldsymbol{0}. \tag{4.1c}$$

In fact, the physical system can be defined as a manifold $\mathcal{P}$, the state space, that includes the admissible states that fulfil a set of equations defining the particular physical problem in hands (Sussman and Wisdom, 2015). Eqs. (4.1) may be symbolically represented using one single equation, $\boldsymbol{R}(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{f}, \boldsymbol{g}) = \boldsymbol{0}$ where $\boldsymbol{u}, \boldsymbol{v}$ are the unknown fields and $\boldsymbol{f}, \boldsymbol{g}$ are problem data. Usually, the state space is treated as an embedded manifold $\mathcal{P} \subset \mathbb{R}^{n+k}$ in a higher Euclidean dimensional space defined in

terms of the set of state governing equations $\boldsymbol{R}(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{f}, \boldsymbol{g}) = \boldsymbol{0}, (\boldsymbol{u}, \boldsymbol{v}) \in \mathbb{R}^{n+k}$, that is:

$$\mathcal{P} = \{\boldsymbol{p} = (\boldsymbol{u}, \boldsymbol{v}) \in \mathbb{R}^{n+k} | \boldsymbol{R}(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{f}, \boldsymbol{g}) = \boldsymbol{0}\}, \tag{4.2}$$

where n is the number of the essential variables, associated with the measurable fields, and k is the number of internal variables. As stated in Chapter 2, the observables of the system are magnitudes that can be measured directly, and are related to the state variables by means of geometric or physical relations, $\boldsymbol{z} = \boldsymbol{O}(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{f}, \boldsymbol{g})$. In most cases, for instance, two direct observables are the fields $\boldsymbol{z} = \boldsymbol{f}$ and $\boldsymbol{z} = \boldsymbol{g}$, that is, the source term and boundary conditions. For example, in Continuum Mechanics, forces, displacements and strains are observable while stresses are not. The relationship between forces and stresses is defined by means of the equilibrium equations, while the relation between displacements and strains is derived from kinematic conditions. A set of measurements of the system is a set of observables obtained in particular conditions. Particularising again for Continuum Mechanics, the state variables are the stress and the strain tensors, with $9 + 9 = 18$ components in 3D problems for each spatial point. These components are not independent, but have to fulfil the universal and/or empirical laws (e.g. symmetry), given by $\boldsymbol{R}$, and are related to the observables $\boldsymbol{f}$ and $\boldsymbol{g}$ by kinematic and equilibrium (linear and angular momentum conservation) constraints.

The fundamental problem in Physics is stated as: from a set of measurements in the physical system given by the values of several observables $\boldsymbol{z} = \boldsymbol{O}(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{f}, \boldsymbol{g})$, derive the value of another target variable $\boldsymbol{z} = \boldsymbol{Q}(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{f}, \boldsymbol{g})$. In particular, this latter variable may be another observable and/or the rest of state variables $\boldsymbol{u}$ or $\boldsymbol{v}$. Unfortunately, this inference strongly depends on the quality of measurements as well as on the particular complexity of the system.

In a DD framework, the state of the system as well as any desired target variable has to be derived from a set of measurements. However, in practical situations, we do not have complete information about the state variables, but only particular measurements in particular states. That is, we have an incomplete set of data. As we have seen, measurements are specific realisations of the observables, and are therefore related to the state-space variables. However, the information given by these measurements is frequently less informative than the state variables themselves. In other words, measurements are known variables living in a lower dimensional space, that is, the number of independent components of $\boldsymbol{z}$ is lower than the number of independent components of $\boldsymbol{p} \in \mathcal{P}$. For example, the displacements along a given direction do not provide the complete information on the strain tensor in the whole domain, not even at the specific point considered, while the forces over a surface do not characterise the whole stress tensor at the point considered. As these measurements are related to the state variables, they may be formulated in terms of manifolds. For the Continuum Mechanics problem, for instance, the knowledge of the stress associated with a given plane orientation tells us the relationship between components of the stress tensor at that point, so we have a set of measurements with a reduced dimension of the state space.

For instance, if the aim is to use a DD solver for solving the problem, e.g. the RBDD solver presented in Chapter 2, it is essential to have complete information associated with the sample point of the empirical manifold $\boldsymbol{p} = (\boldsymbol{u}, \boldsymbol{v}) \in \mathcal{D}$. We usually have

incomplete information, $\boldsymbol{z}^* = \boldsymbol{O}(\boldsymbol{u}, \boldsymbol{v})$, being then the goal to properly complete these incomplete measurements to perform DD computations, that is, to reconstruct $\boldsymbol{p}$ from the measurement $\boldsymbol{z}$. Using the formalism described here, it is possible to define a measurement as a manifold, described as:

$$\mathcal{M} = \{(\boldsymbol{u}, \boldsymbol{v}) \in \mathbb{R}^{n+k} | \boldsymbol{z} - \boldsymbol{O}(\boldsymbol{u}, \boldsymbol{v}) = \boldsymbol{0}\}. \tag{4.3}$$

It is important to note that, due to the noise inherent to any measurement, it may be the case that $\mathcal{M} \cap \mathcal{P} = \emptyset$. This fact justifies the following definition:

Definition 4.1: Measurement

A measurement is a manifold $\mathcal{M} \subset \mathbb{R}^{n+k}$. A complete measurement is a manifold of dimension 0, that is, a point $\boldsymbol{p} \in \mathbb{R}^{n+k}$.

The reconstruction of a complete measurement from an arbitrary measurement is therefore a strategy for obtaining points from arbitrary manifolds. When the function $\boldsymbol{O}$ is a cartesian projection, the reconstruction may be seen merely as a filling data strategy.

Any measurement reconstruction should take into account the following assumptions:

- It should consider all measurements, formulated in terms of manifolds embedded in the ambient space (incomplete measurements) rather than points (complete measurements). We call this condition the **generalisation** assumption.
- It should respect the geometry of the system: the new derived data obtained from the incomplete data should be consistent with the geometric structure defined by the data manifolds. We call this condition the **consistency** assumption.
- It should be compatible with the actual state of the system. Among all observations in our data-set, those closer to our physical constraints or to the actual value of some observables should be overweighted. This is done by means of an **appropriate weighting strategy**.

The methodology for incomplete data processing presented herein will be defined in accordance with this framework for DDSBES.

First, it is fundamental to define an averaging technique that takes into account complete and incomplete measurements according to the generalisation assumption and that respects the data structure, according to the consistency assumption. Next, an appropriate weighting strategy for this kind of problems will be defined, i.e, how to compute a set of weighting values in the averaging process in order to make good predictions for the incomplete measurements. Consequently, the data completion step should take into account how far are the measurements from the known observables and/or the physical manifolds.

4.2.2 Averaging procedure: Generalisation and consistency

In this section, the reconstruction strategy is presented, using a formal mathematical framework. Let us consider a set of complete measurements in the measurement

space, $\boldsymbol{p} \in \mathbb{R}^n$, being n the dimension of the ambient space[10], and a set of incomplete measurements that, in general, will be embedded manifolds $\mathcal{M} \subset \mathbb{R}^n$, with $\mathcal{M} = \{\boldsymbol{p} \in \mathbb{R}^n \,|\, \boldsymbol{z} - \boldsymbol{O}(\boldsymbol{p}; \boldsymbol{f}, \boldsymbol{g}) = 0\}$, where we consider $\boldsymbol{O} : \mathbb{R}^n \to \mathbb{R}^d, d < n$ as a function of the state variables $\boldsymbol{p}$[11], the map defining the manifold $\mathcal{M}$, with $\dim(\mathcal{M}) = n - d = m$. A very particular example of these maps is the one of orthogonal projections on a linear manifold. In Continuum Mechanics, normal stresses associated with a given plane, strains associated with a given direction or the mean pressure at a given point are examples of these incomplete measurements, where only linear relations between the components of the whole stress or strain tensors are known.

Next, some mathematical generalisations of the mean concept will be derived. Let us assume that we have a set of N weighted points $\{(w_j, \boldsymbol{p}_j)| \, , 0 \leq w_j, \, \boldsymbol{p}_j \in \mathbb{R}^n, j = 1, \cdots, N\}$. A possible interpretation of this mathematical structure is a set of data points (complete measurements) with different reliability. Weights can then be associated with the measurement accuracy or reliability (explored later in Section 4.2.4) or other criteria as clustering or outlier filtering. We define the mean square error (MSE) function associated with a given point $\boldsymbol{x} \in \mathcal{P}$ as:

$$\mathrm{MSE}(\boldsymbol{x}) = \sum_{j=1}^{N} w_j d^2(\boldsymbol{x}, \boldsymbol{p}_j) = \sum_{j=1}^{N} w_j \|\boldsymbol{p}_j - \boldsymbol{x}\|^2. \tag{4.4}$$

Suppose that the weights are normalised such that $\sum_{j=1}^{N} w_j = 1$. A classical result from probability theory, presented in Appendix A, states that the function MSE is minimised when $\boldsymbol{x} = \bar{\boldsymbol{p}}$, and the value of this minimum is $\mathrm{Tr}(\boldsymbol{S})$ where $\bar{\boldsymbol{p}} = \sum_{j=1}^{N} w_j \boldsymbol{p}_j$ is the weighted mean value of the data and $\boldsymbol{S} = \sum_{j=1}^{N} w_j (\boldsymbol{p}_j - \bar{\boldsymbol{p}})(\boldsymbol{p}_j - \bar{\boldsymbol{p}})^\intercal$ its variance-covariance matrix.

Inspired by the precedent result, our aim is to generalise the concept of mean and variance for arbitrary manifolds and arbitrary constraints.

4.2.2.1 Manifolds instead of points: generalised mean and variance

Let us suppose now that we have N measurements, that is, N manifolds $\mathcal{M}_1, \cdots, \mathcal{M}_N$. Inspired by the previous theorem, given a set of N weights $\{0 \leq w_j, \, j = 1, \cdots, N\}$, we can generalise the concept of mean by defining the value $\boldsymbol{p}^*$ that minimises the weighted MSE function given by Eq. (4.4) except for the fact that now we consider manifolds instead of points (and therefore the distance from a point to a manifold, not between points). We can therefore try to minimise:

$$\mathrm{MSE}(\boldsymbol{x}) = \sum_{j=1}^{N} w_j d^2(\boldsymbol{x}, \mathcal{M}_j). \tag{4.5}$$

Hence, it is natural to state:

[10] For generality purposes, we will denote as n instead of $n + k$ the dimension of the ambient space, even if this is later applied for problems involving internal and external variables.

[11] As in the previous footnote, for problems involving internal and external variables $\boldsymbol{p} = (\boldsymbol{u}, \boldsymbol{v})$.

Definition 4.2: Generalised mean and variance

The generalised mean $\boldsymbol{p}^* \in \mathbb{R}^n$ is the value minimising the MSE defined in Eq. (4.5) and the value of the minimum V^* is called the generalised variance:

$$\boldsymbol{p}^* = \arg\min_{\boldsymbol{x}\in\mathbb{R}^n} \mathrm{MSE}(\boldsymbol{x}), \tag{4.6}$$

and

$$V^* = \min_{\boldsymbol{x}\in\mathbb{R}^n} \mathrm{MSE}(\boldsymbol{x}) = \mathrm{MSE}(\boldsymbol{p}^*). \tag{4.7}$$

The problem given by Eq. (4.5) is in general nonlinear. However, one may derive a closed form for the expression of the generalised mean and variance if the measurements are all linear manifolds.

Computational solution.

Let us first derive a computational solution to the problem (4.5) for linear manifolds. Let us consider the linear manifolds defined in terms of their director vector spaces $\mathcal{M}_j = \boldsymbol{m}_j + M_j$, where $\boldsymbol{m}_j \in \mathbb{R}^n$ and M_j is the generator vector space associated with $\mathcal{M}_j$, that can be defined with an orthonormal basis $M_j = \langle \boldsymbol{u}_{j,1}, \boldsymbol{u}_{j,2}, \ldots, \boldsymbol{u}_{j,d_j} \rangle$. Here d_j is the dimension of $\mathcal{M}_j$. Let $\boldsymbol{A}_j$ be the matrix with column vectors $\boldsymbol{u}_{j,i}$, $A_{ji} = u_{j,i}$, $j = 1, \cdots, N$, $i = 1, \cdots, d_j$, then we have the following result.

Proposition 4.1: Computational characterisation of the generalised mean for linear manifolds

The solution of the problem given by Eq. (4.5) is obtained by solving the linear system:

$$\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}, \tag{4.8}$$

where:

$$\boldsymbol{A} = \left(\sum_{j=1}^{N} w_j\right) \boldsymbol{I} - \sum_{j=1}^{N} w_j \boldsymbol{A}_j \boldsymbol{A}_j^\intercal \tag{4.9}$$

and

$$\boldsymbol{b} = \left(\sum_{j=1}^{N} w_j \boldsymbol{m}_j\right) - \sum_{j=1}^{N} w_j \boldsymbol{A}_j \boldsymbol{A}_j^\intercal \boldsymbol{m}_j. \tag{4.10}$$

Proof. The distance from the point $\boldsymbol{x}$ to a linear manifold $\mathcal{M}_j = \boldsymbol{m}_j + M_j$, is given by

$$d_j^2(\boldsymbol{x}, \mathcal{M}_j) = \|\boldsymbol{x} - (\pi_{M_j}(\boldsymbol{x} - \boldsymbol{m}_j) + \boldsymbol{m}_j)\|^2, \tag{4.11}$$

where π_{M_j} is the (vectorial) orthogonal projection over the vector subspace M_j. Using the matrix expression in coordinates of the orthogonal projection:

$$d_j^2(\boldsymbol{x}, \mathcal{M}_j) = \|\boldsymbol{x} - (\boldsymbol{A}_j \boldsymbol{A}_j^\intercal (\boldsymbol{x} - \boldsymbol{m}_j) + \boldsymbol{m}_j)\|^2 \tag{4.12}$$

with $\boldsymbol{A}_j$ the matrix associated to M_j.

If $D^2(\boldsymbol{x}) = \sum_{j=1}^{N} w_j d_j^2(\boldsymbol{x}, \mathcal{M}_j)$, the function to minimise yields:

$$D^2(\boldsymbol{x}) = \sum_{j=1}^{N} w_j \|\boldsymbol{x} - (\boldsymbol{A}_j \boldsymbol{A}_j^\intercal(\boldsymbol{x} - \boldsymbol{m}_j) + \boldsymbol{m}_j)\|^2. \tag{4.13}$$

We can compute the gradient of D^2 as:

$$\begin{aligned}
\frac{\partial(D^2)}{\partial \boldsymbol{x}} &= 2\sum_{j=1}^{N} w_j \left(\boldsymbol{I} - \boldsymbol{A}_j \boldsymbol{A}_j^\intercal\right)\left(\boldsymbol{x} - \left(\boldsymbol{A}_j \boldsymbol{A}_j^\intercal(\boldsymbol{x} - \boldsymbol{m}_j) + \boldsymbol{m}_j\right)\right), \\
\frac{\partial(D^2)}{\partial \boldsymbol{x}} &= 2\sum_{j=1}^{N} \left(\left[w_j \boldsymbol{I} - w_j \boldsymbol{A}_j \boldsymbol{A}_j^\intercal\right]\boldsymbol{x} - \left[w_j \boldsymbol{I} - w_j \boldsymbol{A}_j \boldsymbol{A}_j^\intercal\right]\boldsymbol{m}_j\right).
\end{aligned} \tag{4.14}$$

Solving for $\frac{\partial(D^2)}{\partial \boldsymbol{x}} = \boldsymbol{0}$, we obtain:

$$\left[\left(\sum_{j=1}^{N} w_j\right)\boldsymbol{I} - \sum_{j=1}^{N} w_j \boldsymbol{A}_j \boldsymbol{A}_j^\intercal\right]\boldsymbol{x} = \left[\sum_{j=1}^{N} w_j \boldsymbol{m}_j - \sum_{j=1}^{N} w_j \boldsymbol{A}_j \boldsymbol{A}_j^\intercal \boldsymbol{m}_j\right]. \tag{4.15}$$

□

We observe that, if $w_j = \frac{1}{N}$ and the linear manifolds have 0 dimension, that is, they are points, $\mathcal{M}_j = \{\boldsymbol{m}_j\}$ and $\boldsymbol{A}_j = \boldsymbol{0}$, then:

$$\boldsymbol{A} = \boldsymbol{I} - \sum_{j=1}^{N} w_j \boldsymbol{0}\boldsymbol{0}^\intercal = \boldsymbol{I}, \tag{4.16}$$

$$\boldsymbol{b} = \frac{1}{N}\sum_{j=1}^{N} \boldsymbol{m}_j = \tilde{\boldsymbol{m}}, \tag{4.17}$$

obtaining, therefore, the mean of the points $\boldsymbol{m}_j$, being this the reason for the denomination of generalised mean.

4.2.2.2 Manifolds instead of the whole space: consistent mean and variance

Let us suppose now that we have a manifold $\mathcal{M}$ (that is, an incomplete measurement) and N points of $\mathbb{R}^n$, $j = 1, \cdots, N$, described in terms of N coordinate vectors $\boldsymbol{p}_1, \ldots, \boldsymbol{p}_N$.

Given a set of N weights $\{0 \leq w_j,\ j = 1, \cdots, N\}$, we are now interested in the value that minimises the weighted MSE function given by Eq. (4.4), but assuming that it belongs to $\mathcal{M}$. The value $\boldsymbol{p}^* \in \mathcal{M}$ that minimises the weighted MSE function given in Eq. (4.4) is the solution of the constrained minimisation problem:

$$\min_{\boldsymbol{x} \in \mathcal{M}} \ \mathrm{MSE}(\boldsymbol{x}). \tag{4.18}$$

This justifies the following definition.

Definition 4.3: Consistent mean and variance

The consistent mean $\boldsymbol{p}^*_{\mathcal{M}}$ with respect to $\mathcal{M}$ is the value minimising the MSE, that is, the solution of Eq. (4.18) and the value of the minimum $V^*_{\mathcal{M}}$ is called the consistent variance:

$$\boldsymbol{p}^*_{\mathcal{M}} = \arg\min_{\boldsymbol{x}\in\mathcal{M}} \mathrm{MSE}(\boldsymbol{x}), \tag{4.19}$$

and

$$V^*_{\mathcal{M}} = \min_{\boldsymbol{x}\in\mathcal{M}} \mathrm{MSE}(\boldsymbol{x}) = \mathrm{MSE}(\boldsymbol{p}^*). \tag{4.20}$$

Note that now the manifold $\mathcal{M}$ acts as a constraint of the problem. As before, the problem is in general nonlinear, although for linear manifolds, we can derive a closed form for the value of the mean.

Computational solution

Now, to derive a computational solution to the problem (4.18) when $\mathcal{M}$ is a linear manifold, we solve an equivalent unconstrained minimisation problem.

In Appendix A we show that, if $\pi_{\mathcal{M}}$ is the orthogonal projection over $\mathcal{M}$, the solution to the problem (4.18) is given by:

$$\boldsymbol{x}^* = \pi_{\mathcal{M}}(\bar{\boldsymbol{x}}), \tag{4.21}$$

and, that if $s^2_{\mathcal{M}} = \mathrm{MSE}(\boldsymbol{x}^*)$,

$$s^2_{\mathcal{M}} = \sum_{j=1}^{N} w_j(\boldsymbol{x}_j - \pi_{\mathcal{M}}(\boldsymbol{x}_j))^2 + \sum_{j=1}^{N} w_j(\pi_{\mathcal{M}}(\boldsymbol{x}_j) - \boldsymbol{x}^*)^2. \tag{4.22}$$

Note that we have obtained an orthogonal decomposition of the quadratic spread of vectors $\boldsymbol{x}_j$. The term $\sum_{j=1}^{N} w_j(\pi_{\mathcal{M}}(\boldsymbol{x}_j) - \boldsymbol{x}^*)^2$ is denoted as $s^2_{\mathcal{M}}$ since it represents the spread of the points $\boldsymbol{x}_j$ projected on the manifold $\mathcal{M}$. On the other hand, as $(\boldsymbol{x}_j - \pi_{\mathcal{M}}(\boldsymbol{x}_j))^2 = d^2(\boldsymbol{x}_j, \mathcal{M})$, the term $\sum_{j=1}^{N} w_j(\boldsymbol{x}_j - \pi_{\mathcal{M}}(\boldsymbol{x}_j))^2$ may be expressed as $\sum_{j=1}^{N} w_j d^2(\boldsymbol{x}_j, \mathcal{M})$ and computes how far are the points from the manifold $\mathcal{M}$.

Therefore, the next result is straightforward.

Proposition 4.2: Computational characterisation of the consistent mean for linear manifolds

Using the same hypothesis as in the previous result and if the linear manifold $\mathcal{M}$ is described using an orthonormal basis $\{\boldsymbol{p}; \boldsymbol{u}_1, \ldots, \boldsymbol{u}_d\}$, we have:

$$\boldsymbol{x}^* = \boldsymbol{A}\boldsymbol{A}^\intercal \bar{\boldsymbol{x}}, \tag{4.23}$$

$$s^2_{\mathcal{M}} = \mathrm{Tr}(\boldsymbol{A}\boldsymbol{A}^\intercal \boldsymbol{S}), \tag{4.24}$$

with $\boldsymbol{A}$ the matrix with column vectors $\boldsymbol{u}_i$, $\bar{\boldsymbol{x}}$ the mean of points $\boldsymbol{x}_j$, that is, $\bar{\boldsymbol{x}} = \sum_{j=1}^{N} w_j \boldsymbol{x}_j$ and $\boldsymbol{S}$ the variance - covariance matrix of the points $\boldsymbol{x}_j$, that is $\boldsymbol{S} = \sum_{j=1}^{N} w_j(\boldsymbol{x}_j - \bar{\boldsymbol{x}})(\boldsymbol{x}_j - \bar{\boldsymbol{x}})^\intercal$.

Proof. Let us define $\boldsymbol{T} = \sum_{j=1}^{N} w_j(\pi_{\mathcal{M}}(\boldsymbol{x}_j) - \pi_{\mathcal{M}}(\bar{\boldsymbol{x}})(\pi_{\mathcal{M}}(\boldsymbol{x}_j) - \pi_{\mathcal{M}}(\bar{\boldsymbol{x}}))^{\intercal}$. Therefore:

$$\begin{aligned}
\boldsymbol{T} &= \sum_{j=1}^{N} w_j(\boldsymbol{A}\boldsymbol{A}^{\intercal}\boldsymbol{x}_j - \boldsymbol{A}\boldsymbol{A}^{\intercal}\bar{\boldsymbol{x}})(\boldsymbol{A}\boldsymbol{A}^{\intercal}\boldsymbol{x}_j - \boldsymbol{A}\boldsymbol{A}^{\intercal}\bar{\boldsymbol{x}})^{\intercal}, \\
\boldsymbol{T} &= \boldsymbol{A}\boldsymbol{A}^{\intercal}\left[\sum_{j=1}^{N} w_j(\boldsymbol{x}_j - \bar{\boldsymbol{x}})(\boldsymbol{x}_j - \bar{\boldsymbol{x}})^{\intercal}\right]\boldsymbol{A}\boldsymbol{A}^{\intercal}, \\
\boldsymbol{T} &= \boldsymbol{A}\boldsymbol{A}^{\intercal}\boldsymbol{S}\boldsymbol{A}\boldsymbol{A}^{\intercal}.
\end{aligned} \tag{4.25}$$

Consequently, $s^2_{\mathcal{M}} = \mathrm{Tr}(\boldsymbol{T}) = \mathrm{Tr}(\boldsymbol{A}\boldsymbol{A}^{\intercal}\boldsymbol{S}\boldsymbol{A}\boldsymbol{A}^{\intercal}) = \mathrm{Tr}(\boldsymbol{A}\boldsymbol{A}^{\intercal}\boldsymbol{S})$, where we have used the fact that Tr is a cyclic operator and $\boldsymbol{A}^{\intercal}\boldsymbol{A} = \boldsymbol{I}$ as $\boldsymbol{u}_i$ are orthonormal vectors. □

We observe that, if $w_j = \frac{1}{N}$ and $\mathcal{M} = \mathbb{R}^n$, that is, $\mathcal{M}$ is the whole space, $\boldsymbol{A} = \boldsymbol{I}$ and:

$$\boldsymbol{x}^* = \bar{\boldsymbol{x}}, \tag{4.26}$$

$$s^2_{\mathcal{M}} = \mathrm{Tr}(\boldsymbol{S}), \tag{4.27}$$

obtaining directly the mean value for the points $\boldsymbol{x}_j$ and the whole uncertainty, being this the reason for the denomination of consistent mean.

4.2.3 Filling data using the consistent generalised mean and variance: an unsupervised learning technique

The idea of using both generalisations at the same time for missing data techniques is natural. Let us assume that we have a set of partially incomplete data $\mathcal{D}$ of size N, that is, a set of manifolds $\{\mathcal{M}_j, j = 1, \cdots, N\}$, with $\mathcal{M}_j \subset \mathbb{R}^n$, with respective weights $\{w_j, j = 1, \cdots, N\}$. Here, $\mathbb{R}^n$ is the embedding space and $\mathcal{M}_j$ represents the (incomplete) measurements related to different states.

One strategy for data completion of missing data is using the tools presented in Sections 4.2.2.1 and 4.2.2.2. When those measurements are defined in terms of linear manifolds, we have derived, also, a closed linear expression (Proposition 4.1 and Proposition 4.2). The idea is to solve the next minimisation problem for each measurement $\mathcal{M}_i$, $i = 1, \cdots, N$:

$$\min_{\boldsymbol{x} \in \mathcal{M}_i} \quad \mathrm{MSE}(\boldsymbol{x}) = \sum_{j=1}^{N} w_j d^2(\boldsymbol{x}, \mathcal{M}_j). \tag{4.28}$$

The solution $\boldsymbol{x}_i$ of this problem will be identified with the completed data associated with the incomplete data $\mathcal{M}_i$. To summarise, and for linear manifolds, a strategy is depicted for finding a minimum candidate, which strongly reduces the time required to obtain the actual solution of the minimisation problem (4.28) directly using minimisation algorithms. This strategy follows two steps:

1. **Global computation step.** Computation of the solution point for the unconstrained minimisation problem using the expression given in Proposition 4.1.

2. **Projection step.** Computation of the solution point for the constrained minimisation problem using the projection of the mean value and uncertainty defined in Proposition 4.2.

The filling data procedure is described by the Algorithm 3, with the geometric interpretation provided in Fig. 4.1.

Algorithm 3 Filling data procedure for linear manifolds

Input: Data-set of incomplete measurements $\{\mathcal{M}_j\}_{j=1,\dots,N}$.
for $j = 1, \dots, N$ **do**
    Compute an orthogonal basis of $\mathcal{M}_j$, and the matrix of column vectors $\boldsymbol{A}_j$.
end for
Compute $\boldsymbol{A}$ using Eq. (4.9) and $\boldsymbol{b}$ using Eq. (4.10).
Solve the equation $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$. ▷ Computation of the generalised mean.
for $j = 1, \dots, N$ **do**
    Projection of $\boldsymbol{x}$ to the tangent space of $\mathcal{M}_j$, $\boldsymbol{m}_j$ using Eq. (4.23). ▷ Computation of the consistent mean.
end for
Output: Completed measurements $\boldsymbol{m}_j, \forall j = 1, \dots, N$.

This two-step method may be used when working with non-linear manifolds as an iterative tangent-based algorithm. At each step, the tangent space to the nonlinear manifold is computed at the current point and the linear problem is solved. Thus, a point belonging to the tangent space is obtained. Using the exponential map (Do Carmo and Flaherty Francis, 1992), it is possible to obtain an associated point belonging to the manifold, which is used as the starting point for the next iteration. This strategy is usual in nonlinear computational mechanics (Simo et al., 1995). The iteration scheme stops when the distance between the subsequent global solutions is lower than a given tolerance. This construction is illustrated by the schematic diagram in Fig. 4.2 and the algorithm is detailed next.

Algorithm 4 Filling data procedure for nonlinear manifolds

Input: Data-set of incomplete measurements $\{\mathcal{M}_j\}_{j=1,\dots,N}$. Prescribed tolerance TOL.
Initialization: Selection of points $\boldsymbol{m}_j \in \mathcal{M}_j$.
for $j = 1, \dots, N$ **do**
    Compute tangent base of $\mathcal{M}_j$ at $\boldsymbol{m}_j$, and the matrix of column vectors $\boldsymbol{A}_j$.
end for
Compute $\boldsymbol{A}$ using Eq. (4.9) and $\boldsymbol{b}$ using Eq. (4.10).
Solve the equation $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$. ▷ Computation of the generalised mean.
for $j = 1, \dots, N$ **do**
    Projection of $\boldsymbol{x}$ to the tangent space of $\mathcal{M}_j$, $\boldsymbol{p}_j$ using Eq. (4.23). ▷ Computation of the consistent mean.
    Compute $\epsilon_j = \|\boldsymbol{m}_j - \exp_{\mathcal{M}_j}(\boldsymbol{p}_j)\|$.
    $\boldsymbol{m}_j \leftarrow \exp_{\mathcal{M}_j}(\boldsymbol{p}_j)$.
end for
while $\epsilon_j \geq \text{TOL}, \forall j = 1, \dots, N.$ **do**
    for $j = 1, \dots, N$ **do**
        Compute tangent base of $\mathcal{M}_j$ at $\boldsymbol{m}_j^{(i)}$, and the matrix $\boldsymbol{A}_j$ of column vectors.
    end for
    Compute $\boldsymbol{A}$ using Eq. (4.9) and $\boldsymbol{b}$ using Eq. (4.10).
    Solve the equation $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$. ▷ Computation of the generalised mean.
    for $j = 1, \dots, N$ **do**
        Projection of $\boldsymbol{x}$ to the tangent space of $\mathcal{M}_j$, $\boldsymbol{p}_j$ using Eq. (4.23). ▷ Computation of the consistent mean.
        Compute $\epsilon_j = \|\boldsymbol{m}_j - \exp_{\mathcal{M}_j}(\boldsymbol{p}_j)\|$.
        $\boldsymbol{m}_j \leftarrow \exp_{\mathcal{M}_j}(\boldsymbol{p}_j)$.
    end for
end while
Output: Completed measurements $\boldsymbol{m}_j, \forall j = 1, \dots, N$.

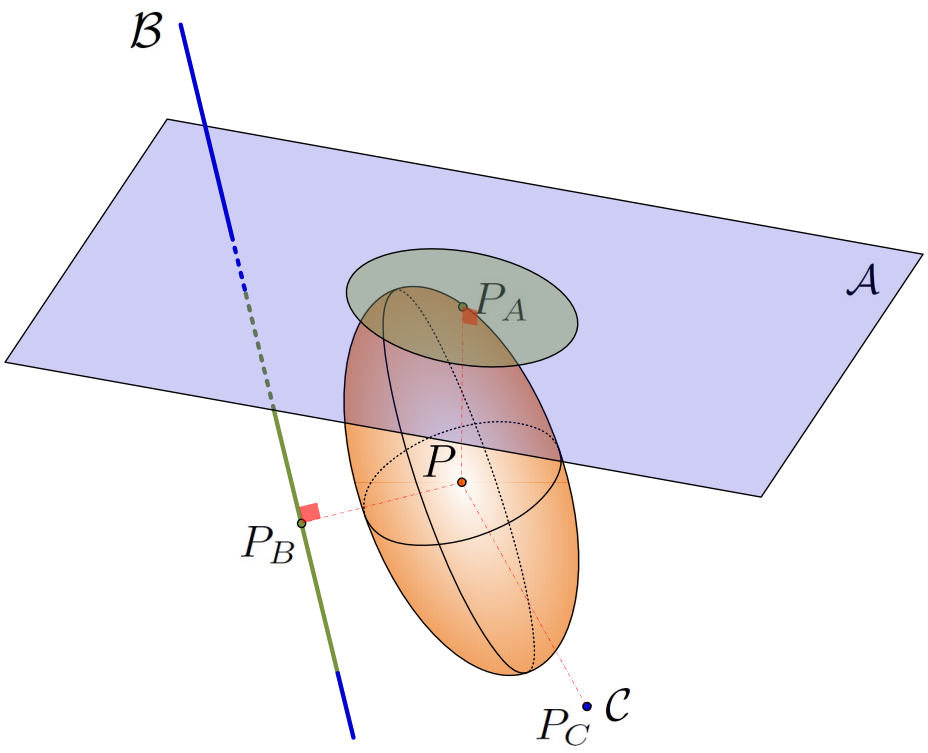


Figure 4.1: Geometric representation of the method proposed in Algorithm 3. $\mathcal{A}$, $\mathcal{B}$ and $\mathcal{C}$ represent three linear manifolds of dimension 2, 1 and 0 respectively, associated with three measurements in a space of dimension 3, the last one complete. Point P is the generalised mean in the sense defined in Eq. (4.5,) that is, the point minimising the sum of (eventually weighted) squared distances to the manifolds. In orange, the uncertainty ellipsoid, related to the generalised variance, represents the spread of (eventually incomplete) measurements. Points P_A, P_B and P_C are the consistent generalised means associated with each of the manifolds in the sense defined in Eq. (4.28), that is, the projection of the generalised mean on each manifold. In green, the associated uncertainty ellipsoid, related to the consistent generalised variance, for each manifold is depicted.

It is important to note that this algorithm may be computationally expensive and is very dependent on the manifold smoothness and convexity, being this type of problems out of the scope of this work.

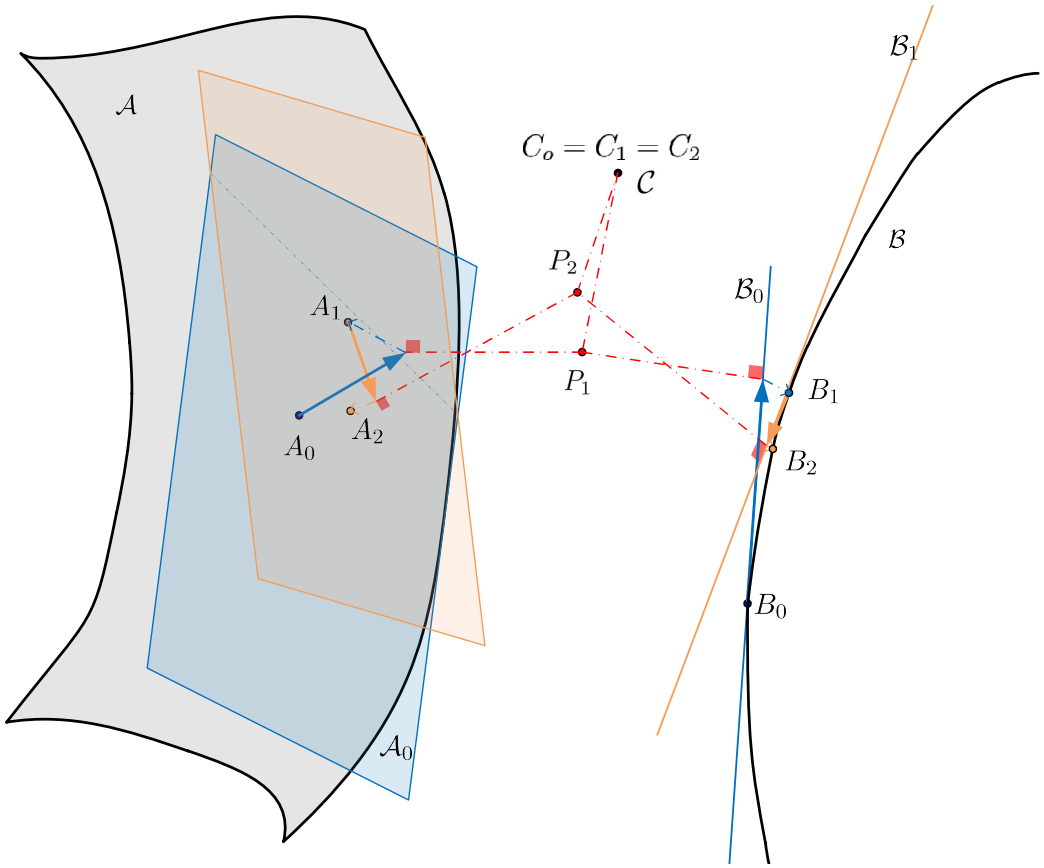


Figure 4.2: Extension of the method to nonlinear problems. Initially, for each manifold ($\mathcal{A}$, $\mathcal{B}$ and $\mathcal{C}$) an initial point is selected (A_0, B_0 and $\mathcal{C}_0 = \mathcal{C}$) and the tangent spaces are computed. Then, the optimal completion, i.e., the generalised consistent mean for each linear manifold (in red), is computed. Using the exponential map, these projections on the linear manifolds are translated to the respective manifolds ($\mathcal{A}$, $\mathcal{B}$ and $\mathcal{C}$) obtaining a new point in each manifold (A_1, B_1 and $C_1 = C_0$). The process is repeated until convergence.

In any case, when the algorithm achieves convergence, N complete measurements are obtained, one for each manifold, $\boldsymbol{x}_i$, $i = 1, \cdots, N$, representing the expected value associated with the manifold $\mathcal{M}_i$. Besides, for linear problems, we obtain for each manifold a value $s^2_{\mathcal{M}_i}$ that characterises the uncertainty related to the manifold $\mathcal{M}_i$ and the matrix $\boldsymbol{S}_{\mathcal{M}_i}$ characterising the uncertainty spread on this manifold. This uncertainty ellipsoid can be seen in Fig. 4.1 for linear manifolds of dimension 0, 1 and 2, respectively.

4.2.4 Introducing physical laws: weighting strategy

Here we introduce the weighting strategy. Let us consider a system defined in terms of a physical manifold $\mathcal{P}$ and a set of measurements, that is manifolds $\mathcal{M}_j, j = 1, \dots, N$. The starting point from which we want to derive the state of the system is another measurement, that is, another manifold $\mathcal{M}^*$ that could be, for instance, boundary conditions for a given problem. In the general framework presented in Eq. (4.2), these observables correspond to the values of $\boldsymbol{f}$ and $\boldsymbol{g}$. The idea is to compute the manifolds $\mathcal{N}_j = \mathcal{M}_j \cap \mathcal{M}^* \cap \mathcal{P} = \mathcal{M}_j \cap \mathcal{P}^*$ where $\mathcal{P}^* = \mathcal{M}^* \cap \mathcal{P}$ is the manifold defined in Eq. (4.2) and to perform the unsupervised learning strategy for this reduced space. This strategy has two direct consequences:

- **Physical consequence**: Since we are using second order statistics of physically admissible manifolds, the result will have a more physical sense.
- **Numerical consequence**: Projections are performed in a smaller space so the computational cost will be lower.

In that case, we are looking for a physically admissible (incomplete) data point measurement that has the lowest uncertainty. However, this strategy can dramatically fail for few data with non-negligible uncertainty. For example, a measurement $\mathcal{M}_j$ may be close to the manifold $\mathcal{P}^*$ but $\mathcal{P}^* \cap \mathcal{M}_j = \emptyset$ so this measurement will not be used for the system learning, even though it is very close to the real state.

An intermediate solution is using an activation function in the learning step, depending on the distance to the manifold $\mathcal{P}^*$, $d = d(\mathcal{P}, \mathcal{M}_j)$. That is, a function $\phi : \mathbb{R}^+ \to [0; 1]$ so that if $z = 0$, $\phi(z) = 1$, and eventually if $z \to +\infty$, $\phi(z) \to 0$. Given $\mathcal{M}_j \subset \mathbb{R}^n$, defining $u_j = u(\mathcal{M}_j) = d(\mathcal{M}_j, \mathcal{P}^*)$, it is possible to define $w_j = \phi(u_j)$ in the learning process. In other words, the nearer the considered data set to $\mathcal{P}^*$, the higher the weight should be in the minimisation of the optimal distance. Some possible activation functions are the step function $\phi(u) = \chi_{[0;a]}(u)$, where χ_A is the characteristic function of the set A and $a \geq 0$, radial basis functions (RBFs), $\phi(u) = \exp\left(-\frac{u^2}{2\varsigma^2}\right)$ with $\varsigma > 0$, homographic functions, $\phi(u) = \left(\frac{a}{a+bu}\right)^k$, with $a, b, k > 0$, or generalised ramp functions $\phi(u) = \left(1 - \left(\frac{u}{a}\right)^k\right) \chi_{[0;a]}(u)$, with $a, k > 0$. Fig. 4.3 illustrates the geometric idea under the presented filling method when combined with the physics of the problem.

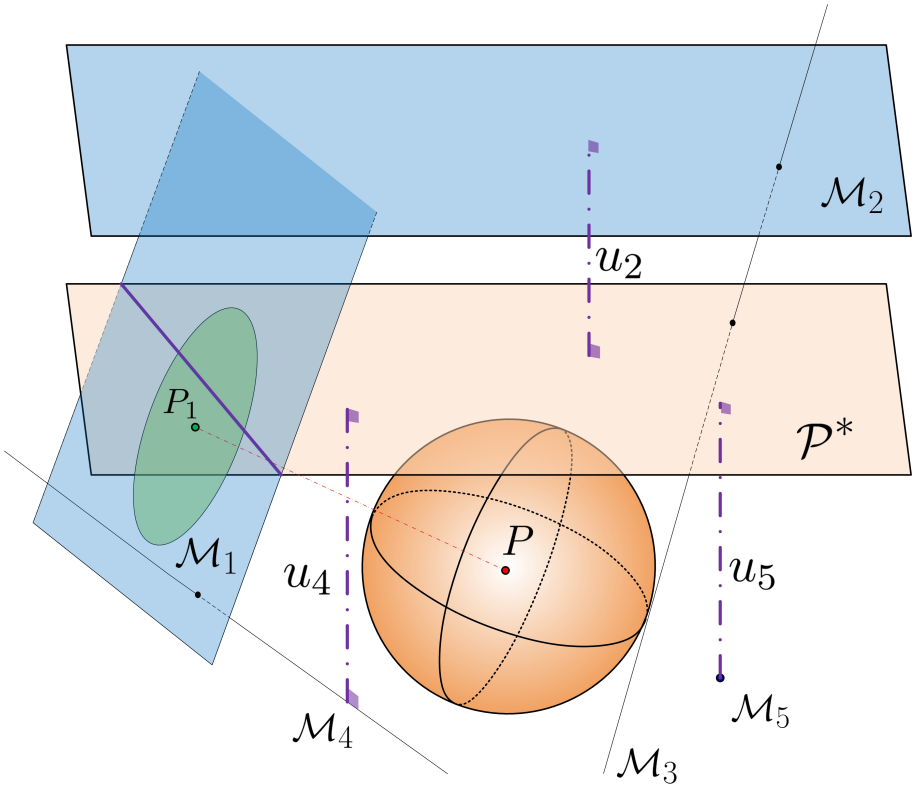


Figure 4.3: Geometric idea of including physics to the learning step. Data completion of the incomplete measure $\mathcal{M}_1$ and its corresponding uncertainty. Blue manifolds $\mathcal{M}_j$, $j = 1, \cdots, 5$. represent measurements (all incomplete except $\mathcal{M}_5$, which is a point). These measurements may correspond to states far from the current state of the system, that should belong to the orange physical manifold $\mathcal{P}^* = \mathcal{M}^* \cap \mathcal{P}$ that corresponds to the intersection of the points satisfying the governing equations of the problem (equilibrium, thermodynamics, Maxwell's equations etc.) and knowledge on the system state, e.g. boundary conditions or measured control variables. Each of the measurement manifolds is at a certain distance u_j from the physical manifold. In particular, in the figure case, $u_1 = 0$ and $u_3 = 0$ because $\mathcal{M}_1 \cap \mathcal{P}^* \neq \emptyset$ and $\mathcal{M}_3 \cap \mathcal{P}^* \neq \emptyset$. Note that this situation is frequent in a three-dimensional state-space, but it is less and less probable when the dimension of the total space becomes very high. From these distances, the weights are calculated by $w_j = \phi(u_j)$, so that if $u = 0$, $\phi(u) = 1$ and if $u \to \infty$, $\phi(u) \to 0$. Thus, if $\boldsymbol{x} \in \mathbb{R}^3$, the distance $D^2(\boldsymbol{x}) = \sum_{j=1}^{5} w_j d_j^2(\boldsymbol{x}) = \sum_{j=1}^{5} \phi(u_j) d^2(\boldsymbol{x}, \mathcal{M}_j)$ is minimised (generalised mean) and the solution point P (in red) is projected onto the measure manifolds (consistent mean) obtaining the measure completion, as illustrated in Fig. 4.1. Likewise, depending on the "spread" of the sets, an ellipsoid of three-dimensional uncertainty (in red), is obtained, related to the generalised variance. This ellipsoid is projected in each of the manifolds, obtaining ellipsoids related to the associated consistent variance (in green). Note that if the red ellipsoid is very slender in the direction orthogonal to the measure manifold that we are completing, this would have no impact on the projected green ellipsoid. The point P_1, is then the generalised consistent mean associated with the incomplete measure $\mathcal{M}_1$, weighted by the neighbourhood to the problem physics.

4.3 Looking for the nearest measure to a given point: solving the data-driven problem

4.3.1 Preliminary mathematical results

Let $\mathcal{M} \subset \mathbb{R}^n$ be an embedded manifold of the Euclidean space of dimension n with associated probability distribution ρ, that accounts for the probability distribution of a given random vector $\boldsymbol{X}$ belonging to $\mathcal{M}$. For instance, in the case of linear manifolds, $\mathcal{M}$ can be described as $\mathcal{M} = \boldsymbol{m} + \langle \boldsymbol{u}_1, \boldsymbol{u}_2, \ldots, \boldsymbol{u}_d \rangle$, where $\langle \boldsymbol{u}_1, \boldsymbol{u}_2, \ldots, \boldsymbol{u}_d \rangle$ is the linear span generated by $\boldsymbol{u}_1, \ldots, \boldsymbol{u}_d$. That is, $\mathcal{R} = \{\boldsymbol{m}; \boldsymbol{u}_1, \boldsymbol{u}_2, \ldots, \boldsymbol{u}_d\}$ is a reference frame of $\mathcal{M}$ and $d = \dim(\mathcal{M})$. ρ may be described using a density function function $\rho : \mathbb{R}^d \to \mathbb{R}^+$ that associates to a vector $\boldsymbol{x} = \boldsymbol{m} + \sum_{i=1}^{d} \alpha_i \boldsymbol{u}_i$ the density $\rho(\alpha_1, \ldots, \alpha_d)$. In general, let $\boldsymbol{X} \in \mathcal{M}$ be a random vector and $\rho : \mathcal{M} \to \mathbb{R}^+$, be the probability distribution describing the position of $\boldsymbol{X}$. Let $\boldsymbol{p} \in \mathbb{R}^n$. We define the square distance random variable $D^2 = d^2(\boldsymbol{p}, \boldsymbol{x})$, with $\boldsymbol{p}$ (deterministic) and $\boldsymbol{X}$ (random) as:

$$D^2 = \|\boldsymbol{p} - \boldsymbol{X}\|^2. \tag{4.29}$$

It is possible to define (under some integrability conditions) the expected value $\mathbb{E}[D^2] = \mathbb{E}\left[\|\boldsymbol{p} - \boldsymbol{X}\|^2\right]$, or, using the probability density function, $\mathbb{E}[D^2] = \int_{\mathcal{M}} ||\boldsymbol{p} - \boldsymbol{x}||^2 \rho(\boldsymbol{x})\, dV(\boldsymbol{x})$. More important than the explicit computation of the expected value of D^2, in terms of a given parametrisation of the manifold, are the following results for linear manifolds, proven in Appendix A, and relating the moments of D^2 to the moments of $\boldsymbol{X}$. If $\boldsymbol{\mu}$ is the expected value of the random vector $\boldsymbol{X}$ ($\mu = \mathbb{E}[\boldsymbol{X}]$), $\boldsymbol{\Sigma}$ its variance - covariance matrix ($\boldsymbol{\Sigma} = \mathbb{E}[\boldsymbol{X}\boldsymbol{X}^\intercal] - \mathbb{E}[\boldsymbol{X}]\mathbb{E}[\boldsymbol{X}]^\intercal$) and $\boldsymbol{\Upsilon}$ the $\pi_{\mathcal{M}}(\boldsymbol{p})$-centered fourth order moment tensor (the tensor $\boldsymbol{\Upsilon}$ with components $\Upsilon_{ijkl} = \mathbb{E}[(X_i - q_i)(X_j - q_j)(X_k - q_k)(X_l - q_l)]$ with $\boldsymbol{q} = \pi_{\mathcal{M}}(\boldsymbol{p})$), then:

$$\mathbb{E}[D^2] = ||\boldsymbol{p} - \pi_{\mathcal{M}}(\boldsymbol{p})||^2 + ||\pi_{\mathcal{M}}(\boldsymbol{p}) - \boldsymbol{\mu}||^2 + \mathrm{Tr}(\boldsymbol{\Sigma}), \tag{4.30}$$

$$\mathrm{Var}(D^2) = \boldsymbol{I} : \boldsymbol{\Upsilon} : \boldsymbol{I} - \left(\mathrm{Tr}(\boldsymbol{\Sigma}) + \|\pi_{\mathcal{M}}(\boldsymbol{p}) - \boldsymbol{\mu}\|^2\right)^2. \tag{4.31}$$

Here, $\boldsymbol{I}$ is the second order identity tensor. Finally, under normality conditions, D^2 follows a noncentral χ^2 distribution with $m = \dim(\mathcal{M})$ degrees of freedom and non-centrality parameter $\lambda = (\pi_{\mathcal{M}}(\boldsymbol{p}) - \boldsymbol{\mu})^\intercal (\boldsymbol{\Sigma})^{-1} (\pi_{\mathcal{M}}(\boldsymbol{p}) - \boldsymbol{\mu})$.

4.3.2 The closest point in a stochastic sense

Let us go back now to the methodology and tools introduced in Section 4.2.2. Once the constrained minimisation problem is solved for each incomplete measurement $\mathcal{M}_i$ and all filled data points $\boldsymbol{m}_i$ are derived following the filling data procedure, it is possible to compute how far a state $\boldsymbol{p} \in \mathbb{R}^n$ of the system is from a given data point $\boldsymbol{m}_i$. Moreover, we can determine, for each $\boldsymbol{p} \in \mathbb{R}^n$, which is the closest measurement, and from this define a tessellation of the state space in terms of the measurements.

One could consider the deterministic distance $d_i = d(\boldsymbol{p}, \boldsymbol{m}_i)$, but this distance would not have into consideration the accuracy of the filling step and the effect of physical weights on uncertainty. It is more natural to consider a stochastic distance. Indeed,

considering again the random variable D_i^2 defined in Section 4.3.1 associated with the manifold $\mathcal{M}_i$, we may define, denoting $s^2_{\mathcal{M}_i}$ by s_i^2 and $\pi_{\mathcal{M}_i}$ by π_i:

$$d_i^2 = \mathbb{E}[D_i^2] = d^2(\boldsymbol{p}, \pi_i(\boldsymbol{p})) + d^2(\pi_i(\boldsymbol{p}), \boldsymbol{m}_i) + s_i^2. \tag{4.32}$$

The manifold $\mathcal{M}_i$ verifying that d_i^2 is minimal is the closest manifold, in the statistical sense, to the point $\boldsymbol{p}$. Besides, each term in d_i^2 has its own interpretation:

- $T_{i,1} = d^2(\boldsymbol{p}, \pi_i(\boldsymbol{p}))$ is the statistical error due to finite measurements of the sample. It is related to the lack of knowledge on the system, since the information is obtained by means of a given finite data-set. The more measurements are added, the lower the error usually is.
- $T_{i,2} = d^2(\pi_i(\boldsymbol{p}), \boldsymbol{m}_i)$ is inherent to the manifold and depends on the manifold selection. It is unavoidable to some extent.
- $T_{i,3} = s_i^2$ is the term associated with the uncertainty and is characteristic of the self-learning process: the worse the manifold learning, the higher this term. Locating properly the completed point in a manifold, even though it actually belongs to that manifold, is less accurate when this term increases.

Moreover, if measurement uncertainty is taken into consideration, it is possible to state $T_{i,3} = s_i^2 + s_i'^2$, where $s_i'^2$ is the quadratic uncertainty of the i-th measurement and, therefore, orthogonal to the uncertainty associated with the filling procedure. Nevertheless, this uncertainty is not considered in the applications presented in this work.

The geometric idea behind these considerations is illustrated in Figure 4.4.

Once the learning step is finished, the DD problem may be solved as usual either using the deterministic distance (Kirchdoerfer and Ortiz, 2016b) or the stochastic one (Ayensa-Jiménez et al., 2018), as described in Chapter 3, provided the uncertainty of the completion is considered, as explained above. Fig. 4.5 illustrates the geometric idea.

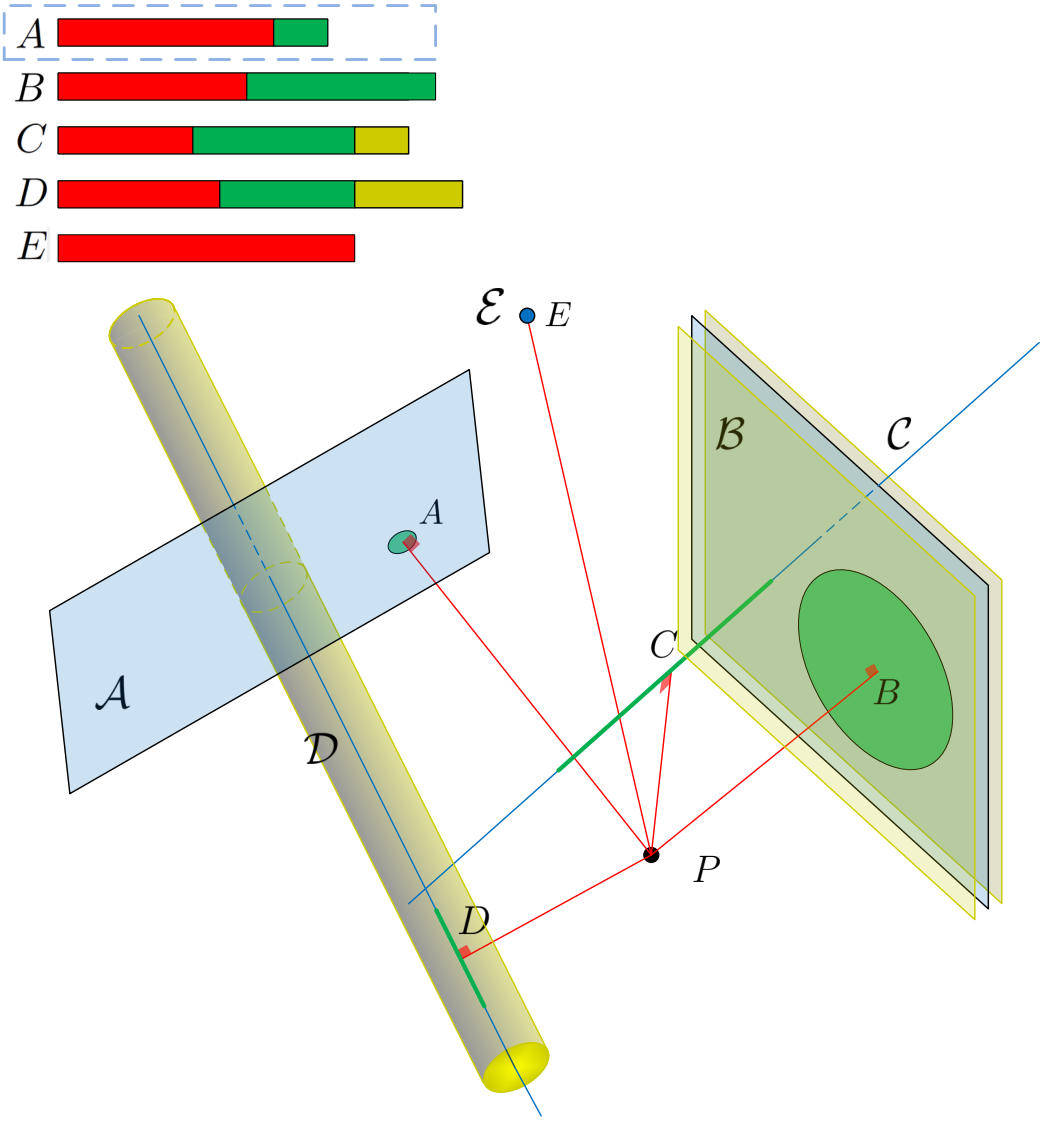


Figure 4.4: Stochastic distances to different measurements. Manifolds $\mathcal{A}$, $\mathcal{B}$, $\mathcal{C}$, $\mathcal{D}$ and $\mathcal{E}$, associated with incomplete measurements, have been completed using the procedure described above. Completed measurements are represented by the points A, B, C, D and E as well as their associated uncertainty ellipsoids (including terms $T_{j,2}$ and $T_{j,3}$). Measurement uncertainty is illustrated in yellow and is taken into account in distance computations. Even if the completed measure associated with the manifold $\mathcal{C}$ is the closest in a deterministic sense, the one associated to the manifold $\mathcal{A}$ is the closest in a stochastic sense and the one associated with the manifold $\mathcal{D}$ the farthest

.

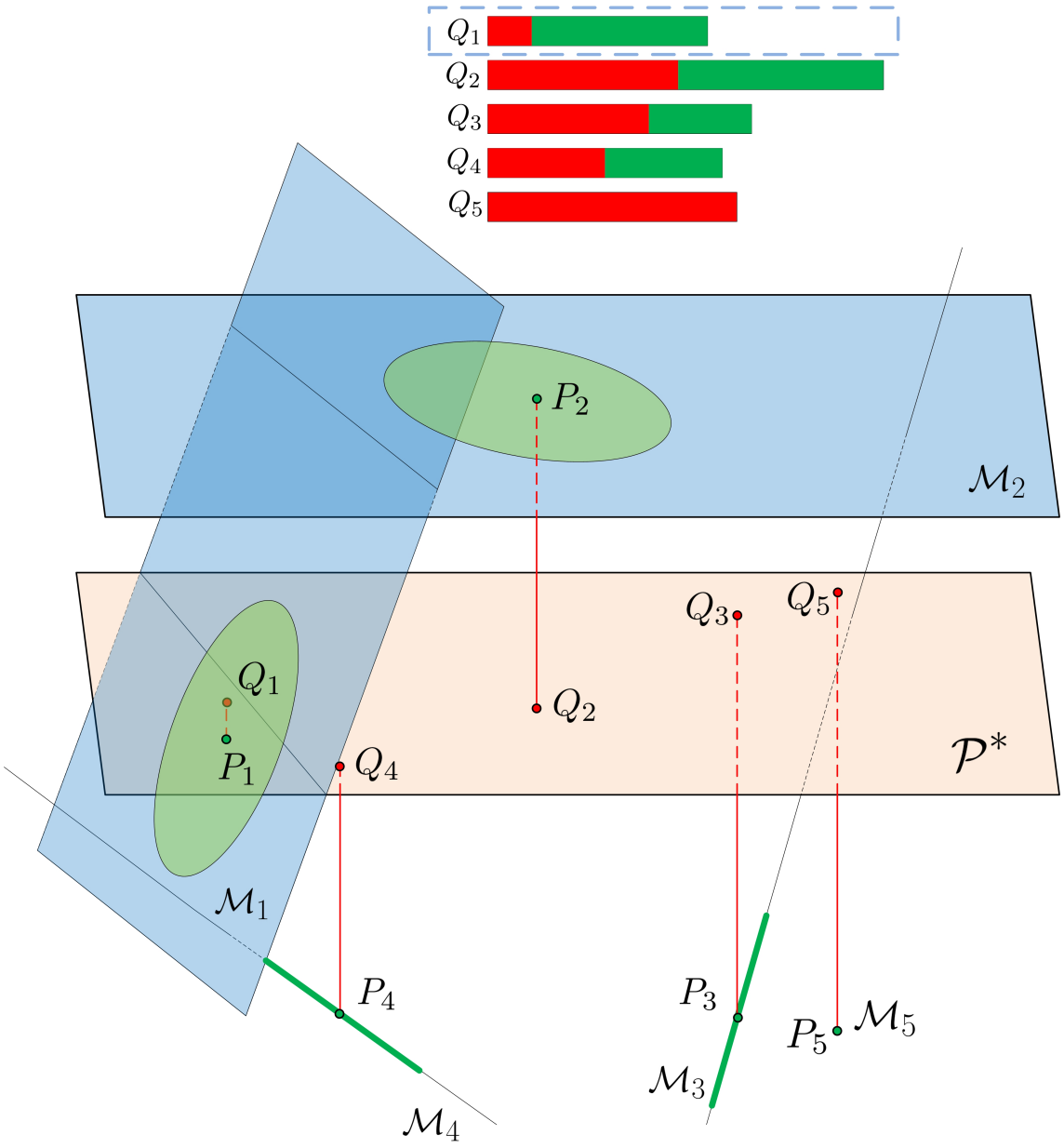


Figure 4.5: Geometric idea of the problem solving stage. We start from complete measures given by points P_i, $i = 1, \ldots, 5$, and associated quadratic uncertainties (consistent generalised variances) s_i^2, $i = 1, \ldots, 5$. Suppose that the previous process has been repeated for all the measurement manifolds and that we have associated complete expected points with their associated uncertainty. Now, the algorithm looks for the complete point closest to the physical manifold, including both the deterministic and the stochastic parts of the squared distance (as shown in Figure 4.4). Then, the point associated with the complete measurement closest to the physical manifold (in this case it would be the measurement associated with the manifold $\mathcal{M}_1$) is selected and its projection over the physical manifold $\mathcal{M}$ is the solution to the problem. In this case, Q_1 is the solution point.

4.4 Applications

Next, we analyse three applications of the presented method. The first corresponds to a standard regression problem. Several model-free missing data techniques will be compared with the one proposed in this work. The second is a physically based example illustrating how the method can be seen as a physically-based mean generalisation, including constraints based on the problem discretisation. Finally, the third one illustrates how the described methodology is particularly suitable for real DD problems based on a physical frame, multiscale tissue biomechanics. For this last problem, we only describe the aspects related to the filling data procedure; the whole work may be found in Mora-Macías et al. (2020).

4.4.1 Standard data-science problem

Let us consider different concrete material specimens. Each of them is characterised in terms of the mass fraction of their constituents: cement, slag, fly ash, water, superplasticizer, coarse aggregate and fine aggregate (in kg/m^3). For each sample, the compression strength on the 28$^{\text{th}}$ day is tested. Assuming a linear relationship between the compression strength, that is the response variable Y, and the water content, that is the explanatory variable X, we set-up a linear regression model, $Y = aX + b$. The goal is to obtain an estimate of the strength for $X = 100$. This can be easily obtained using the standard least squares technique.

Once the full data analysis is performed, we define the following data loss process from the complete data-set, depending on a threshold parameter $0 \leq p \leq 1$

- For data having a water content lower than the $1 - p/2$ quantile and higher than the $p/2$, the water content is removed. This represents a loss of the $100p\%$ of the data due to, for example, experimental difficulties for characterising high and low water contents.
- For data having a cement content higher than the $1 - p$ quantile, the strength is removed. This represents a loss of the $100p\%$ of the data due to, for example, loss of the data for a given batch of experimental trials.

Note that the described loss process is MNAR so that we are in a context where the filling data method should be fine enough to not include bias and then, error in the predicted value.

As the presented method is non-parametric, it is compared to other non-parametric standard methods: Listwise Deletion, and four interpolation techniques (linear interpolation, nearest point interpolation, piece-wise cubic spline interpolation and shape-preserving piecewise cubic interpolation). The error of the method is defined as:

$$\epsilon = \frac{|Y - Y_c|}{Y_c}, \tag{4.33}$$

where Y is the prediction of the incomplete data, following the filling data procedure described before and performing linear regression as if it was the complete data-set, and Y_c is the target value.

In this case, incomplete measurements are the canonical manifolds defined as follows. If $\boldsymbol{X}$ is the matrix of data where each row represents a specimen and each column a variable (cement, slag, fly ash, water, superplasticizer, coarse aggregate and fine aggregate content). A missing data value is described by some specimen i where the j field value is lost. We may then define a missing value matrix $\boldsymbol{M}$, where $M_{ij} = 1$ if the data at $i - j$ slot is missed, and $M_{ij} = 0$ otherwise. Suppose that we have N specimens, $N - K$ of which have the $n = 7$ values fully reported while the rest (K) have incomplete data vectors. For each of the $I = 1, \ldots, K$ incomplete data, the missing value matrix is completed such that $i = I$ and, for each row, some j values are removed, so $M_{ij} = 1$. Let us suppose that we have, for the first incomplete vector $I = 1$, $i = 24$ and $j = 1, 4, 6$. This incomplete data point is then associated with the manifold that may be described using a parametric equation:

$$\mathcal{M}_1 = \{(\lambda, X_{24,2}, X_{24,3}, \mu, X_{24,5}, \nu, X_{24,7}) | (\lambda, \mu, \nu) \in \mathbb{R}^3\}. \tag{4.34}$$

Note that our method could define missing values in a much more sophisticated framework (oblique linear manifolds or even nonlinear manifolds, where we know a relationship between some variables but not the variable itself) using the general expression:

$$\mathcal{M}_I = \{(x_1, x_2, x_3, x_4, x_5, x_6, x_7) | \Phi^I(x_1, x_2, x_3, x_4, x_5, x_6, x_7) = 0\}. \tag{4.35}$$

Since we are interested in the estimate of Y for $X = 100$ kg/m^3, we introduce here the physical or target manifold:

$$\begin{aligned} \mathcal{M} &= \{(\mu_1, \mu_2, \mu_3, 100, \mu_4, \mu_5, \mu_6) | (\mu_1, \mu_2, \mu_3, \mu_4, \mu_5, \mu_6) \in \mathbb{R}^6\} \\ &= \{(x_1, x_2, x_3, x_4, x_5, x_6, x_7) \in \mathbb{R}^7 | x_4 = 100\}. \end{aligned} \tag{4.36}$$

The presented methodology is then used with the manifolds $\mathcal{M}^I$, $I = 1, \cdots, K$ and $\mathcal{M}$ defined by the Eqs. (4.34) and (4.36). The RBF function $\phi(u) = \exp\left(-\frac{u^2}{2\varsigma^2}\right)$, with $\varsigma = 20$ kg/m^3, is selected as weighting function.

The number of incomplete data, K, is dependent on the parameter p. For the present example, we deal with $N = 103$ specimens and our method is compared to the other ones for different values of p. The results are shown in Fig. 4.6 in terms of the fraction of missing data with respect to the complete data-set $F = \frac{K}{N}$.

It is clear from Figure 4.6 that the presented methodology yields better results than the rest of standard filling methods. This is due to the fact that the bias induced by the missing process is here corrected since the filling procedure takes into account how far the data points used are from the target manifold. This local counterpart of the presented methodology makes the method more robust with respect to the missing data fraction in comparison with other interpolation methods. Note that the error of the presented method (UL) does not increase with the amount of lost data. This is due to the fact that the performance of the method depends on how far the missed data points are, and not on their number, i.e. data quality and not data

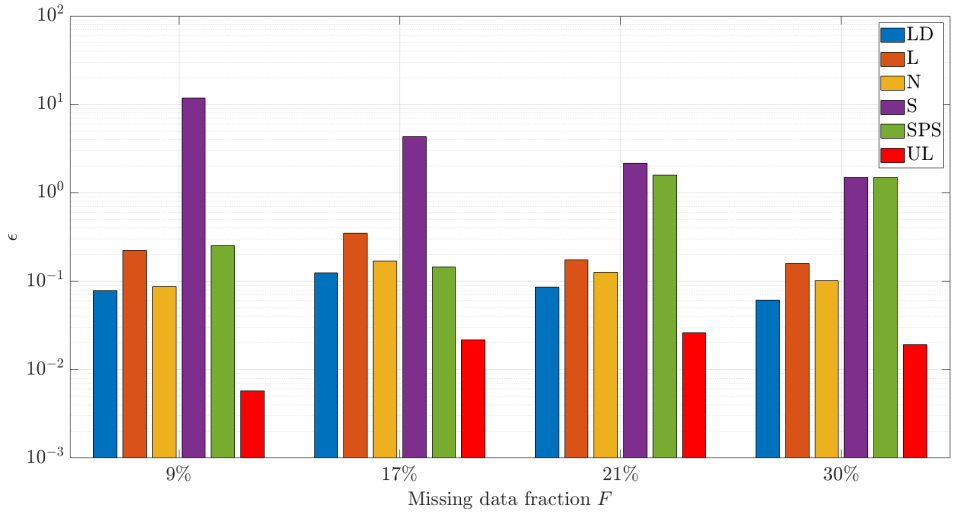


Figure 4.6: Comparison of different filling data procedures for a standard regression problem. Error of the different model-free filling data procedures for the estimation of Y for $X = 100$ and different missing data fractions F (LD: Listwise deletion, L: Linear interpolation, N: Nearest neighbour interpolation, S: Piecewise cubic spline interpolation, SPS: Shape preserving cubic spline interpolation, UL: Unsupervised learning).

quantity. On the other hand, standard interpolation techniques are more dependent on the distance of the missing data from the true solution than on the volume of missing data itself.

Moreover, low order interpolation techniques (nearest or linear interpolation) are sometimes unable to reproduce the underlying data structure, whereas the performance of high order interpolation techniques (cubic splines) is strongly dependent on data sampling (Kahaner et al., 1989). Shape preserving interpolation, for example, was conceived as a compromise solution to these problems, but is strongly dependent on the missing data process as has been demonstrated (Fritsch and Carlson, 1980). Listwise deletion is the most robust method with respect to missing data fraction, but has statistical power and bias problems for MNAR data as reported in the literature (Briggs et al., 2003; Nakai and Ke, 2011). The proposed method shows a more robust behaviour with respect to the missing data fraction.

4.4.2 Model-based data-driven problem

The performance of the method is now illustrated in a classical problem of strength of materials. Let us consider a two-end clamped beam of length L under bending by a linearly distributed load $q = q(x)$, $0 \leq x \leq L$. In the Euler-Bernoulli framework, and supposing the beam composed of a linearly elastic material with Young modulus $E = E(x)$, $0 \leq x \leq L$, and a section with moment of inertia $I = I(x)$, the vertical beam displacement $u = u(x)$, $0 \leq x \leq L$, may be computed solving the linear differential equation:

$$\frac{d^2}{dx^2}\left(EI\frac{d^2u}{dx^2}\right) = q, \tag{4.37}$$

with boundary conditions:

$$u(0) = \left.\frac{du}{dx}\right|_{x=0} = 0, \tag{4.38a}$$

$$u(L) = \left.\frac{du}{dx}\right|_{x=L} = 0. \tag{4.38b}$$

Eq. (4.37), with the boundary conditions (4.38a), may be solved numerically using any standard numerical procedure (e.g. finite element method (FEM) or finite difference method (FDM)). Once the problem is discretised using a mesh of characteristic size h and the boundary conditions are applied, the nodal displacements $\boldsymbol{u}^h$ are obtained solving the linear system[12]:

$$\boldsymbol{K}^h \boldsymbol{u}^h = \boldsymbol{f}^h. \tag{4.39}$$

In order to test our method, we may proceed by considering Eq. (4.39) as a physical constraint to a data-set of measurements $\mathcal{E} = \{\boldsymbol{u}_i^{h'}\}_{i=1,\cdots,N}$ equally spaced h'. Note that this approach makes sense when $h' << h$ (this may be the case when Eq. (4.39) is computationally expensive to solve with very fine meshes while measurements are easy to obtain). In this case, we deal with complete measurements, but they are subjected to error (bias and noise, for example, due to experimental reasons). The presented method is able to detect how far a given measurement is from the physics of the problem in terms of the distance to the manifold defined by Eq. (4.39). The different measurements will be weighted differently depending on their distance to the manifold. Recall that a standard procedure of averaging all measurements may induce an error if there is a systematic bias in the measurements, which is a well-known problem of mean imputation (Schafer, 1997). The weighting strategy considered associates the bias in the estimation with data quality in a physical sense.

In order to illustrate the application of the methodology, let us solve the defined problem for $q(x) = 10$ kN/m, $L = 10$ m, $E(x)I(x) = 1 \cdot 10^6$ kN $\cdot$ m^2. In that case, the analytical solution is given by $u(x) = -\frac{qL^4}{24EI}\left(\frac{x}{L} - \frac{x^2}{L^2}\right)^2$.

The measurements are randomly generated from the analytical solution sampled in a mesh of size $h' = \frac{1}{m-1}$, $u^k \sim \mathcal{N}(u(x = k\frac{L}{h'}) + b, \sigma)$, where $b = \alpha\bar{u}$ is a bias, $\alpha \sim \mathcal{N}(0.03, 0.03)$, $\sigma = \beta\bar{u}$, $\beta = 0.002$, and $\bar{u} = \frac{1}{30}\frac{qL^4}{EI}$ is the mean value of the analytical solution. For this example, $m = 100$ points are considered along the beam length and $N = 6$ samples are evaluated. The six families of points are shown in Fig. 4.7. Also, and in the same figure, the results computed for different spacings h, used to establish the physical constraint, are shown. In particular, the mean of the samples is represented by the bold red colour line and the true analytical solution is represented by the continuous bold grey line. The RBF function $\phi(u) = \exp\left(-\frac{u^2}{2\varsigma^2}\right)$, with $\varsigma = 5 \cdot 10^{-4}$ m, was selected as the weighting function.

As pointed out above, the method here described partially corrects the bias of the measurements. This correction is done automatically by computing the distance of

[12]It is important to note that this equation is characteristic of a broad family of linear discretised problems in Physics and Engineering, not only the Euler-Bernoulli bending beam

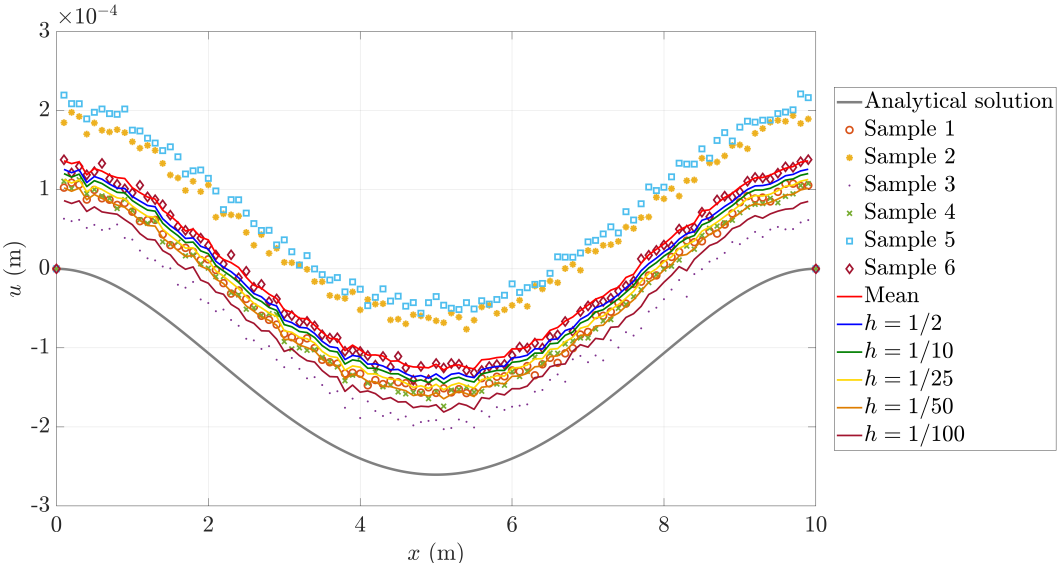


Figure 4.7: Solutions obtained for different mesh sizes in the constrain. Comparison of the solutions obtained using the presented method for different mesh sizes h to define the physical constraints given by Eq. (4.39).

each measure sample to the physical manifold defined by the discretised equilibrium equation and boundary conditions, and transforming this distance using the RBF function. A finer mesh in the discretised physical problem (lower h) takes into account more points to compute the distance from the measure to the physics of the problem. It is clear, however, that the accuracy of the estimation in cases as this one with a systematic positive bias will never be better than the best measurement.

In order to have a deeper understanding of the method, it is worth to make a physical interpretation of this example. For a fixed h, Eq. (4.39) is a constraint relating $n = 1/h + 1$ variables of the m-dimensional space. The manifold $\mathcal{P}^h \simeq \mathbb{R}^n$, of dimension $n \ll m$, defined by these n coordinates is the manifold where the relevant physics of the problem is evaluated. It is easy to figure up what the method does by considering only the projections of the points in $\mathbb{R}^m$ on $\mathcal{P}^h$. In $\mathcal{P}^h$, the physical manifold (that is, the constraint) is given by a single point p: the numerical solution obtained solving Eq. (4.39). Therefore, the distances of the different samples to the physical manifold may be interpreted as Euclidean distances in $\mathcal{P}^h \simeq \mathbb{R}^n$ between sample projections on $\mathcal{P}^h$ (i.e. the consideration of the n coordinates related to the mesh with h spacing) and point P, $d_i = \|\pi_{\mathcal{P}^h}(\boldsymbol{u}_i) - \boldsymbol{u}^h\|$. We could have defined the distance in a more general framework, such as Hilbert spaces, using the FEM approximation, but a simpler and more interpretable norm was selected for illustration purposes. The weighted mean is then computed by using the solution $\boldsymbol{u}^h$ as the reference point in the weights computations, $w_i = \phi(d_i)$. The physics of the problem is inferred outside the manifold $\mathcal{P}^h$, that is, in $(\mathcal{P}^h)^\perp$. In Fig. 4.8, all the described geometric elements are shown for $h = 1/10$: projections over manifold $\mathcal{P}^h$ of m-dimensional samples, numerical solution $\boldsymbol{u}^h$, and associated generalised mean and variance. The analytical solution is also plotted even though it does not appear in the computation.

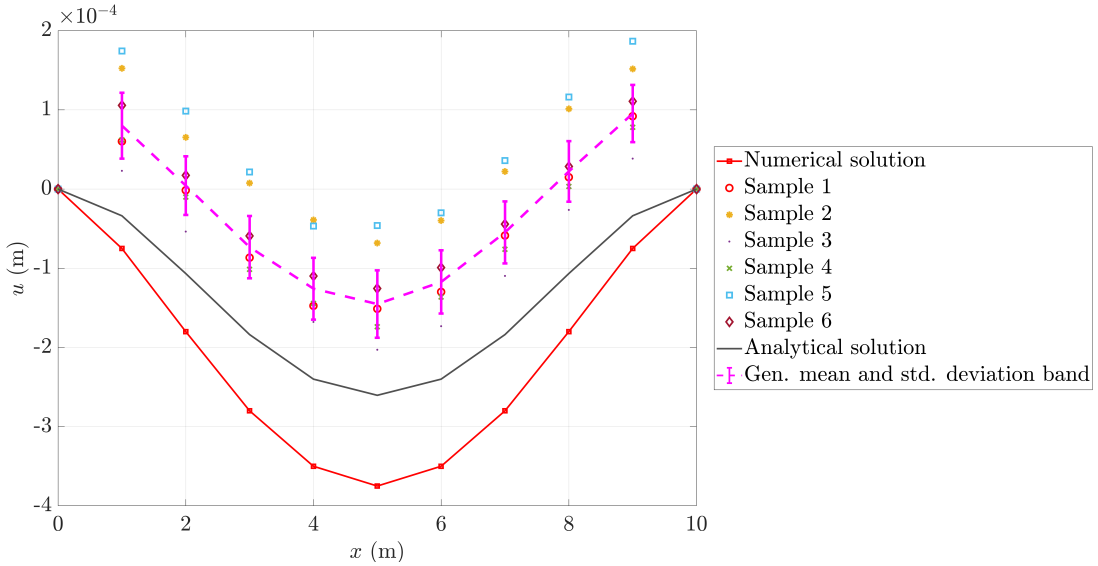


Figure 4.8: Elements of the problem projected on the physical manifold. Projections over the manifold $\mathcal{P}^h$ of all geometric elements used in computations for $h = 1/10$. The dotted magenta line represents the generalised mean (in this case it is equivalent to the weighted mean with complete measurements). The error bands correspond to the projections over the different coordinates lines (manifolds of dimension 1) of the generalised variance ellipsoid (in orange in Fig. 4.1 or Fig. 4.3). Green ellipsoids represented in these figures collapse in the data sample points because samples are complete so the measurement manifolds are points.

4.4.3 Multiscale model-free data-driven problem

4.4.3.1 Contextualization: multiscale analysis in the context of data-driven approach

Aims and scope

Multiscale theory aims to address a problem at different spatial and temporal observation scales: the response of an upper (coarse) scale is seen as the consequence of the evolution of a lower (fine) scale. The reasons for selecting multiscale analysis as a suitable approach are to establish less phenomenological models based on more fundamental principles at different spatio-temporal resolutions and to have access to the distribution of the variables along both the fine and coarse scales. A paradigmatic case where the multiscale approach has been fruitful is the mechanical analysis of heterogeneous materials (Matouš et al., 2017; Nguyen et al., 2011), containing an underlying microstructure, such as natural (biological) or artificial (technological) materials. For instance, many multiscale techniques have been implemented to account for plasticity and the subsequent microstructure evolution (Kouznetsova et al., 2002; Miehe and Bayreuther, 2007), poroelasticity and microstructural fluid circulation (Terada and Kikuchi, 2001) or micromechanical analysis of composite adhesives (Kulkarni et al., 2010; Reina-Romo and Sanz-Herrera, 2011). Bone tissue mechanics is another framework where the multiscale approach has also been applied: loads are transferred at the macroscopic organ scale whereas the driving stimuli of the internal biological evolution processes are local and microstructural (Carter et al., 1988; Claes and Heigele, 1999; Cowin, 2002). Therefore, multiscale models have been used in bone tissue engineering (Sanz-Herrera et al., 2008) and their application in scaffold design (Nguyen et al., 2018; Sanz-Herrera et al., 2009). The reader is addressed to Montero-Chacón et al. (2019) for a review of multiscale techniques applied to continuum models.

Multiscale analysis based on finite elements (FE) is usually called FE^2 (Feyel, 1999; Yuan and Fish, 2008), due to the fact that two meshes are needed for the FE procedure. Indeed, the corresponding simulations have to be performed both at the macro and microlevels, this latter defined at each Gauss point of the macro FE mesh. Even though some remedies have been proposed in the literature to alleviate the CPU time demand (El Halabi et al., 2016), this kind of analysis is usually intractable and with limited applicability to the practical use. Here we exploit the experimental data acquisition of the mechanical characterisation of a bone specimen, available both at the tissue (macro) and microstructural (micro) levels, to propose a multiscale approach in the context of DD. It is clear that the feasibility of this DD technique relies on the availability of test data at both scales. However, once a proper macro and microtissue characterisation is performed, the solution at the microstructure is obtained as a postprocessing in the DD methodology, avoiding prohibitive CPU times. Therefore, if the experimental procedures enable the access to both mechanical field distributions at macro and microstructural levels, the savings proposed by this methodology are evident.

Experimental techniques for Data-Driven methods

The DD technique is inherently linked to the experimental characterisation of the material in hands, such as the bone tissue in this study, prior to DD simulations.

An example of bone tissue is cortical bone, which is found in the diaphysis of long bones as an outer layer of the trabecular bone. At the macroscopic level, it looks like a compact and continuous solid. The mechanical response and properties measured at the macroscopic level of the cortical bone are however influenced by its microarchitecture. This relation between micro and macroscale has been studied for decades. For example, Evans and Vincentelli (1969) related the mechanical properties with the collagen fiber orientation. More recently, Tai et al. (2007) showed that the nanoscale heterogeneity of the mechanical properties of the bone promotes energy dissipation. A multiscale DD approach could take advantage from the data provided by experimental techniques, which allows characterising the cortical bone tissue at the microscopic level.

Among the techniques used to characterise the cortical bone tissue at that microscopic level, histologies, and microscopy techniques allow characterising biological aspects of the bone microarchitecture (Evans and Vincentelli, 1969). More recent techniques, such as microcomputer tomography, provided bone microarchitecture, and other parameters that may be related to mechanical properties, such as the mineral density (Brouwers et al., 2009; Scharmga et al., 2016). Nanoindentation can also provide mechanical properties of a bone surface, which, in combination with microscopic techniques or microcomputer tomography, may define the bone heterogeneity and the spatial variations of its mechanical properties (Mora-Macías et al., 2017; Tai et al., 2007). However, none of these methods is able to measure the micromechanical response (stress and/or strain at the microscopic level) of bone tissue during mechanical experiments.

Another technique, called digital image correlation (DIC), has shown advantages over the aforementioned methods to measure the microscopic strain field, of the order of tens of microns, in real time, in bone samples under different loading conditions. Consequently, the macroscopic strain field, of the order of millimetres, can be obtained by averaging along the domain of the region of interest, or the microstructural representative volume element (RVE). This technique consists in comparing the positions in a speckle pattern of reference, usually spray-painted at the loaded sample surface, where the images are taken. Carriero et al. (2014) and Sztefek et al. (2010) used DIC to compute the strain field in a region of the surface of a mouse tibia, with a resolution of 125 μm distance between the field strain points. Thompson et al. (2007) also used DIC for determining the local mechanical conditions within the early bone callus providing strain fields of the same order of magnitude. Currently, micro-strain fields provided by DIC measurements have been integrated into experimental studies which combine different length scale measurements (*in situ* loading, DIC, and synchrotron X-ray) in order to understand the deformation mechanisms within the bone tissue (Gustafsson et al., 2018). In conclusion, DIC is a contrasted and validated technique to measure microscopic strain fields above the micron scale, and the associated strain stress macroscopic field in bone tissue.

4.4.3.2 Data completion technique

We have proposed new data completion techniques in the context of DD problems. From the above, it is clear that data completion tools are useful in at least two contexts:

1. First, they allow using computational DD algorithms that need to perform searches in complete spaces. This includes to augment data resolution, to expand data coverage and to fill missing data or gaps in the sample.
2. Second, they can be used in an appropriate way as field reconstructors. Indeed, a synthetic field may be computed from an existing data-set, but taking into account reliability and physical admissibility of data.

Here, the proposed data-completion technique is used in both ways.

Data augmentation

On the one hand, at the macroscale level, pairs of strain-stress couples $\boldsymbol{\varepsilon} = (\varepsilon_{xx}, \varepsilon_{yy})$ and $\boldsymbol{\sigma} = (\sigma_{xx}, \sigma_{yy})$ are sampled from an experimental setup. This gives us an unstructured data-set $\mathcal{D} = \{(\boldsymbol{\varepsilon}^i, \boldsymbol{\sigma}^i)\}_{i=1,\dots,N}$, where N is the sample size. However, this amount of data N may not be large enough for some purposes. For instance, if there is no control in one of the sampled variables, such as in longitudinal strains and stresses in our experimental setup, data coverage may be poor. Besides, for computational purposes, the resolution of data is often not high enough, needing some tools to work, sometimes at the continuum level. The methodology presented in this chapter is therefore used in order to obtain, from an unstructured data-set $\mathcal{D}$, a new structured data-set satisfying $u^i = u_{\min} + \frac{i}{n}(u_{\max} - u_{\min})$, $i = 1, \dots, n$, $n = 50$, where $u_{\min}$ and $u_{\max}$ are the minimal and maximal values reached by a variable u with $u = \varepsilon_{xx}, \varepsilon_{yy}, \sigma_{xx}, \sigma_{yy}$. The filling procedure is performed adapting the strategy proposed in this chapter: for each stress state $\boldsymbol{\sigma}^*$, a new strain state $\boldsymbol{\varepsilon}^*$ is computed by solving the minimisation problem:

$$\boldsymbol{\varepsilon}^* = \underset{\boldsymbol{\varepsilon}}{\operatorname{argmin}} \sum_{i=1}^{N} \varphi\left(\|\boldsymbol{\sigma}^i - \boldsymbol{\sigma}^*\|\right) \|\boldsymbol{\varepsilon}^i - \boldsymbol{\varepsilon}\|^2. \tag{4.40}$$

This problem is easily obtained by solving a linear system, as explained in Section 4.2.2. For the data presented in this work, a RBF is selected as activation function, with $\zeta = 80\,\mathrm{N}$ as spread parameter.

Microscopic state reconstruction

On the other hand, the data-completion technique is used for the computation of the microscale strain field, given the macroscale strains and stresses. In this case, the aim is to obtain a microscale field, built from the data-set of possible microscale fields, but compatible with the macroscale pairs of strains and stresses. The approach here is similar, except for the fact that we must impose the constraint relating the macroscopic strain $\boldsymbol{\varepsilon}^M$ and the microscale strain field $\boldsymbol{\varepsilon}(\boldsymbol{x})$ ($\boldsymbol{x} \in \mathcal{V}$, the RVE associated with the macroscopic point) which is none other than:

$$\boldsymbol{\varepsilon}^{\mathrm{M}} = \frac{1}{V} \int_{\mathcal{V}} \boldsymbol{\varepsilon}(\boldsymbol{x})\, dV. \tag{4.41}$$

As only the microscopic strains fields are given in the data-set, these are the only data that may be used in the minimisation procedure for obtaining a microscale

field. Therefore, the problem is here reduced to computing $\boldsymbol{\varepsilon}^*(\boldsymbol{x})$ by solving the minimisation problem:

$$\begin{aligned} \varepsilon^*(\boldsymbol{x}) = \underset{\boldsymbol{\varepsilon}(\boldsymbol{x})}{\operatorname{argmin}} \sum_{i=1}^{N} \varphi\left(\| \frac{1}{V} \int_{\mathcal{V}} \boldsymbol{\varepsilon}^i(\boldsymbol{x})\, dV - \boldsymbol{\varepsilon}^{\mathrm{M}} \| \right) \| \boldsymbol{\varepsilon}(\boldsymbol{x}) - \boldsymbol{\varepsilon}^i(\boldsymbol{x}) \|^2 \\ \text{subject to} \quad \frac{1}{V} \int_{\mathcal{V}} \boldsymbol{\varepsilon}(\boldsymbol{x})\, dV = \boldsymbol{\varepsilon}^{\mathrm{M}}. \end{aligned} \tag{4.42}$$

Note that $\frac{1}{V}\int_{\mathcal{V}} \boldsymbol{\varepsilon}^i(\boldsymbol{x})\, dV$ is the macroscale strain associated with the microscale field $\boldsymbol{\varepsilon}^i(\boldsymbol{x})$, $\boldsymbol{\varepsilon}^{\mathrm{M}}$ is the macroscopic strain whose microscale strain field is intended to be obtained, and $\mathcal{D} = \{\boldsymbol{\varepsilon}^i(\boldsymbol{x})\}_{i=1,\dots,N.}$ is the learning data-set. As the constraint is linear, this problem is easily solved following the two-step strategy described in Section 4.2.2, where first a linear system is solved and then the solution is projected onto a linear manifold, obtaining the final result. Of course, the accuracy of the result depends on the coverage of the data-set $\mathcal{D}$.

Application to multiscale simulations

We have seen that the previous data-completion technique may be used for the computation of the microscale strain field, given the macroscopic strain state. This enables the possibility of plugging the computation of the microscale in any DD simulation, as the ones presented in Chapter 3, provided that we have a micro-macro data-set relating the strain fields of both scales. The procedure is illustrated schematically in Fig. 4.9.

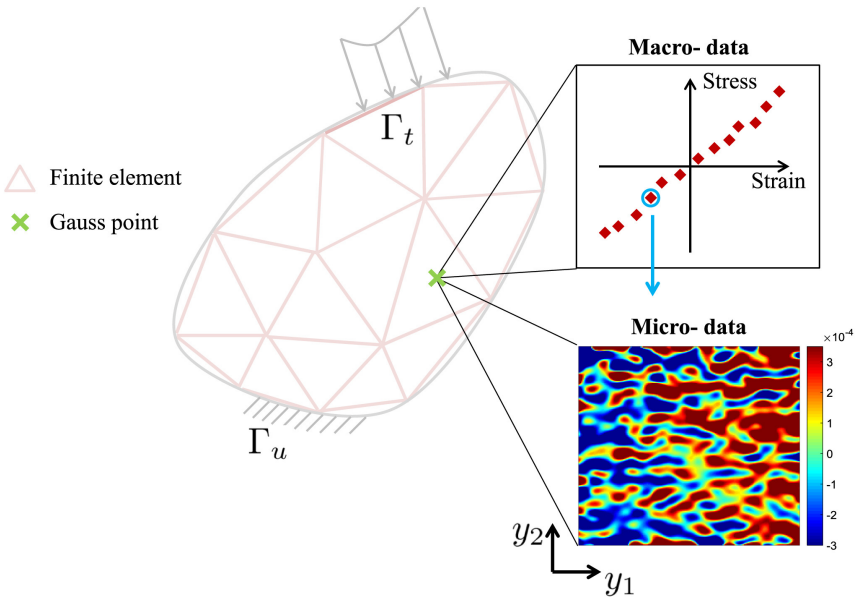


Figure 4.9: Multiscale DD simulation scheme. The pair of macroscopic strain-stress variables at each Gauss point are used as input for the reconstruction of the microscopic strain field, using the completion procedure described in the previous sections.

Note that, using this strategy, the computation of the microscopic fields is carried out as a post-process. Therefore, the computational cost of our study is that of the

DD techniques. Next, we discuss the computational cost of a DD technique versus a multiscale one in a generic framework.

The DD algorithm turns into an iterative solver solution as exposed by Kirchdoerfer and Ortiz (2016b). The cost of each iteration is the one of a FEM solver for a single load state. Hence, the DD computational cost is of the order $\omega(N_D) \times \mathcal{O}(N) \times N_{\text{iter}}$, where $\mathcal{O}(N)$ is the computational cost of a FEM iteration and N is the size of the FEM macroscopic mesh. $\omega(N_D) \times \mathcal{O}(N)$ is the computational cost of a FEM iteration corrected by the searching algorithm of the data-set at each iteration, that depends on the data coverage, and on the data-set size N_D. It has been estimated to be of order 2 for a computer with core processor i7 @3.4 GHz and RAM 32 GB, using the Matlab searching algorithm `knnsearch`. Finally N_{iter} is the number of iterations, which is of the order of 100–1000 in a DD approach (Kirchdoerfer and Ortiz, 2017).

On the other hand, the cost of a FE^2 (generic nonlinear) multiscale approach is, according to Feyel (1999), of the order $\text{NL}_{\text{iter}} \times \text{nL}_{\text{iter}} \times N_G \times \mathcal{O}(n)$ where N_G is the number of Gauss points of the macroscopic mesh, NL_{iter} and nL_{iter} are the macroscopic and microscopic nonlinear iterations (for instance, using the Newton-Raphson algorithm) and $\mathcal{O}(n)$ is the computational cost of a FEM iteration, being n the size of the FEM microscopic mesh.

Therefore, the computational cost of the proposed DD multiscale method is competitive versus multiscale numeric schemes if:

$$\omega(N_D)\mathcal{O}(N)N_{\text{iter}} \gg \text{NL}_{\text{iter}} \cdot \text{nL}_{\text{iter}} \cdot N_G \cdot \mathcal{O}(n). \tag{4.43}$$

Usually we have $N_G \sim N$, $\text{NL}_{\text{iter}} \sim 10$, $\text{nL}_{\text{iter}} \sim 10$, $N_{\text{iter}} \sim 2000$ in a conservative scenario. Then, Eq. (4.43) yields:

$$N\mathcal{O}(n) \gg 20 \cdot \mathcal{O}(N). \tag{4.44}$$

Moreover, assuming that the macro and micro meshes are comparable in a conservative case, that is $n \sim N$ (in fact, the microstructure mesh is usually more detailed in a multiscale approach), Eq. (4.44) yields:

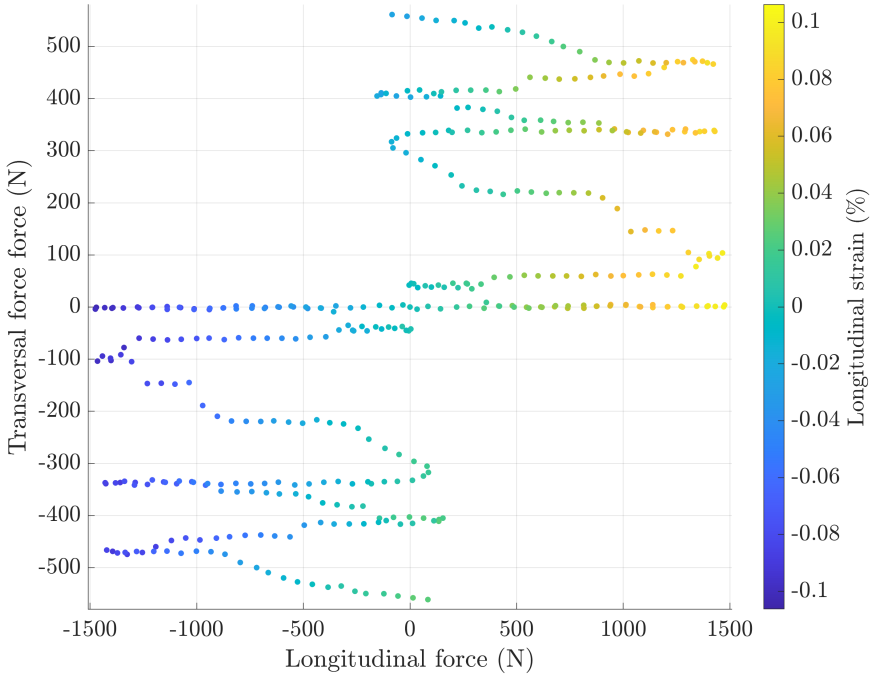
$$N \gg 20. \tag{4.45}$$

This last condition is usually satisfied, which justifies the proposed multiscale DD approach.

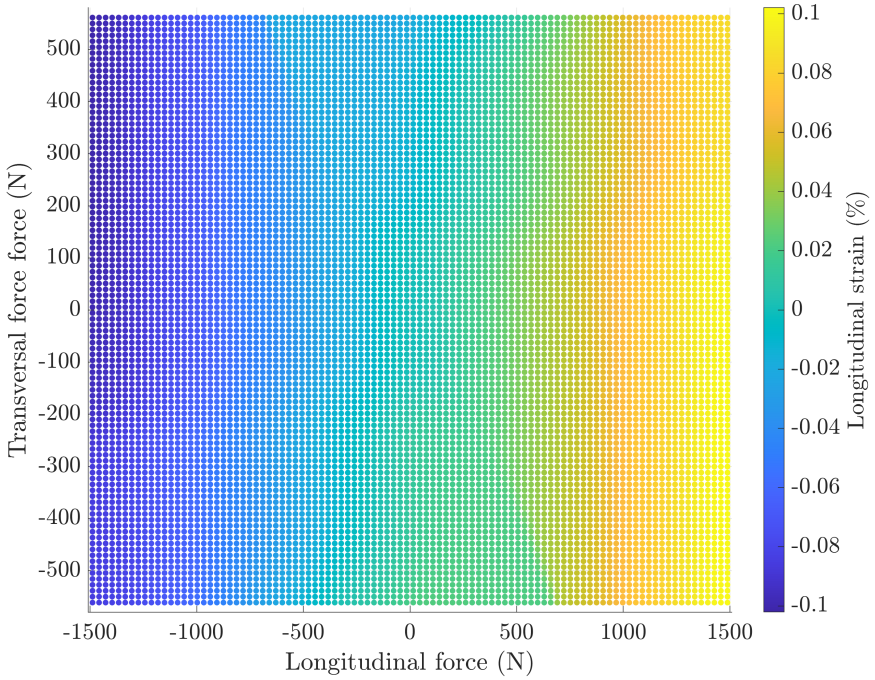
4.4.3.3 Results

Data augmentation

First, we shall explore the results of the data-completion technique for data augmentation purposes. The initial data were sampled pairs $(\boldsymbol{\sigma}, \boldsymbol{\varepsilon})$, as shown in Figs. 4.10a (longitudinal strain) and 4.11a (transversal strain). These data are expanded by means of the described filling procedure. The macroscopic upsampled data are presented in Figs. 4.10b (longitudinal strain) and 4.11b (transversal strain) using a sampling size of $n = 100 \times 100$. The upsampled macroscopic data-set is presented

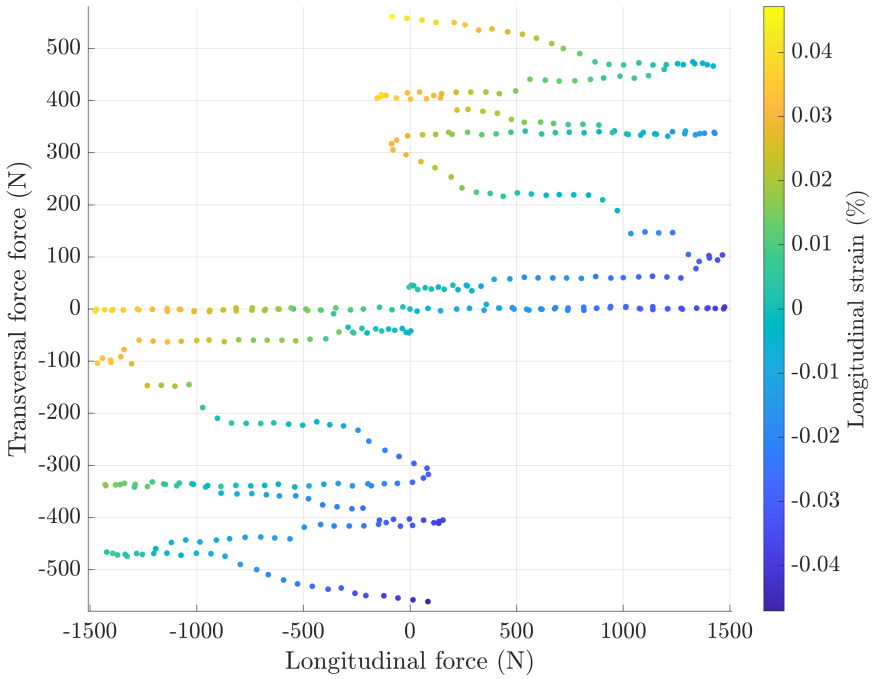


(a) Original longitudinal strain, $\varepsilon_{xx} = f(\sigma_{xx}, \sigma_{yy})$

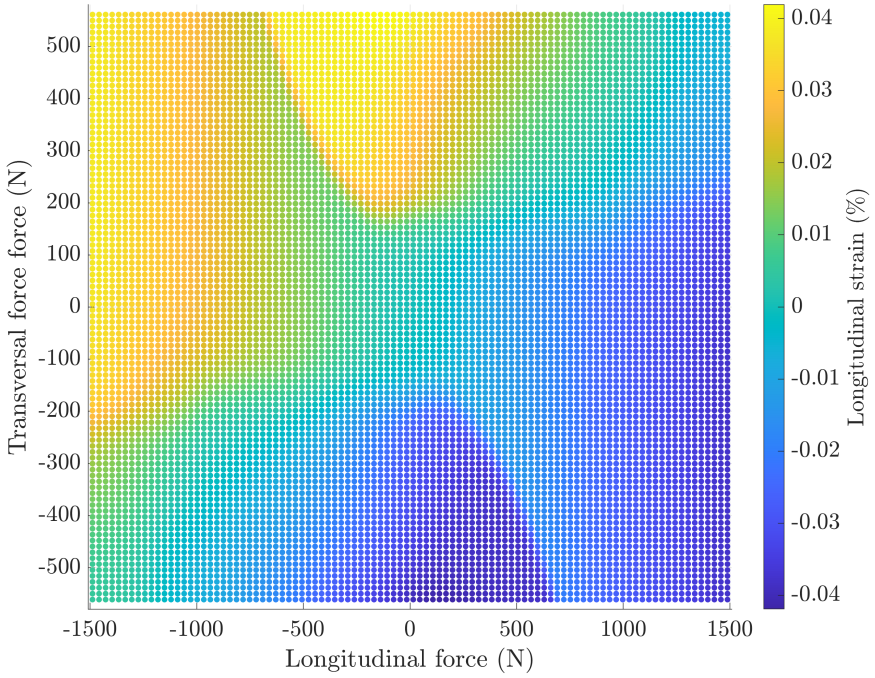


(b) Upsampled longitudinal strain, $\varepsilon^*_{xx} = f(\sigma_{xx}, \sigma_{yy})$

Figure 4.10: Original and upsampled longitudinal strain field. The filled data will be more reliable in the regions close to original data. Outside, the method makes an extrapolation but using some of the data learned structure.



(a) Original transversal strain, $\varepsilon_{yy} = f(\sigma_{xx}, \sigma_{yy})$



(b) Upsampled transversal strain, $\varepsilon^*_{yy} = f(\sigma_{xx}, \sigma_{yy})$

Figure 4.11: Original and upsampled transversal strain field. The filled data will be more reliable in the regions close to original data. Outside of them, the method performs an extrapolation but using some of the data learned structure as it might be observed for example at the region $(\sigma_{xx}, \sigma_{yy} \in [-1500\,\mathrm{N}; 0\,\mathrm{N}] \times [0\,\mathrm{N}; 500\,\mathrm{N}]$, where the reconstructed data generates a valley.

over the measured macroscopic data-set showing good agreement between measured and synthetic (upsampled) data.

In order to compare the reconstructed data with the original ones, something that is only possible for points close to the sampled data, Fig. 4.12 compares both datasets, the original and the upsampled one. The reconstruction is coherent in the sense that the estimated value is similar to the actual sampled one for points close to the sampled data.

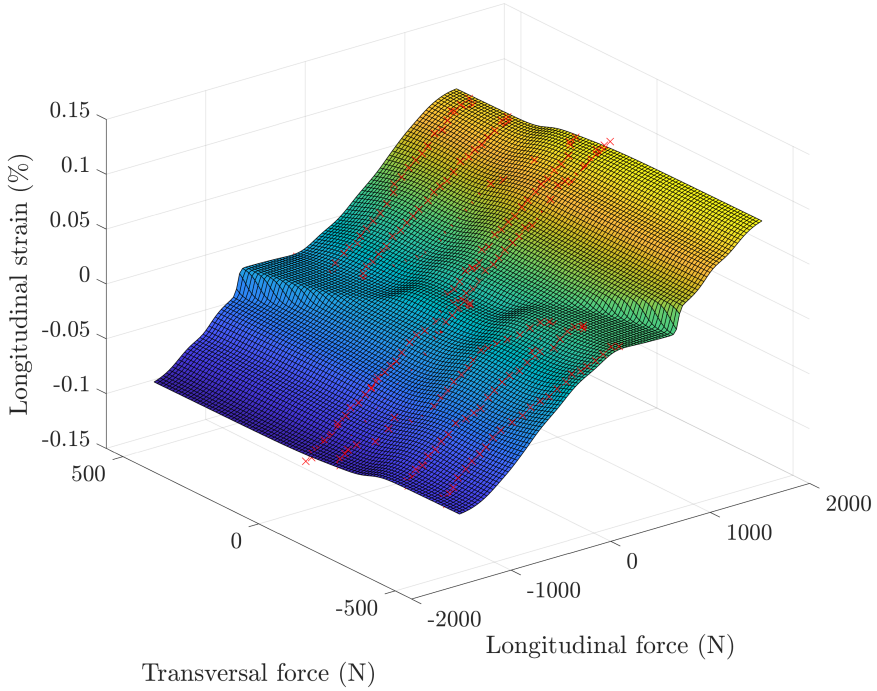
Microscopic state reconstruction

The associated microscopic data-set is also upsampled together with their corresponding upsampled macroscopic data-set. In this case, Fig. 4.13 shows the estimated longitudinal and transversal microscopic strain fields along the RVE that are associated with a certain prescribed macroscopic strain state ($\varepsilon^{\mathrm{M}}_{xx} = -2.000 \times 10^{-3}$ and $\varepsilon^{\mathrm{M}}_{yy} = -2.000 \times 10^{-3}$). For comparison purposes, the figures show the closest measured microscopic strain fields from the data, associated with the specified macroscopic strain state.

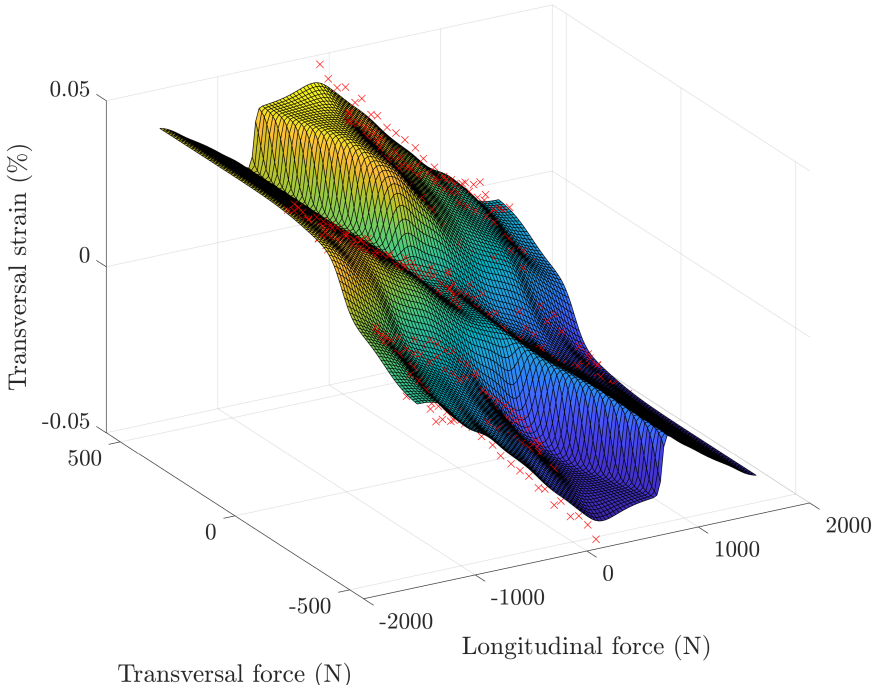
Moreover, in order to test the performance of the data-set, the following validation test was established: we removed random points of the data-set and used the data completion algorithm to reconstruct the microscopic strain fields for those points. The comparison of these points versus the real ones measured by means of DIC is given in Fig. 4.14 and Fig. 4.15 for 2 removed points (although conclusions are extended for the remaining 2 points). Qualitatively, a good agreement can be observed in these figures for the microscopic fields. Of course, the mean (macroscopic) value of the strains is exactly the same for both real and estimated, since the query strain field in the algorithm is constrained to fulfil the mean input value according to Eq. (4.41). Since the micromechanical fields in the RVE have to be interpreted statistically and in average terms (Suquet, 1985), we consider that the approach given by the data completion algorithm is accurate enough.

Application to multiscale simulations

A multiscale DD simulation is then performed using the experimental data-set corresponding to cortical bone test samples obtained by Mora-Macías et al. (2020). The procedure is analogous to the one explained in Chapter 3 for the concrete example, except for the fact that we use another (macroscopic) data-set $\mathcal{D} = \{(\boldsymbol{\varepsilon}^{\mathbf{M},i}, \boldsymbol{\varepsilon}^{\mathbf{M},i}), i = 1, \ldots, N\}$, where $N = 100 \times 100$ is the augmented data-set. Note that without this data-augmentation, the DD would have failed, as the data scarcity would have obstructed an appropriate local search in Algorithm 1. The geometry corresponding to the example consists of a biaxially loaded plate with an elliptical inclusion oriented along a 45° axis (see Fig. 4.16). The selected example of application fulfils the length-scale separation hypothesis for multiscale analysis, since the order of magnitude of the heterogeneity considered in the microscale is quite below the order of magnitude of the characteristic scale of centimetres of bone tissue. The problem is then a 2D plane stress situation since it is conditioned to the availability of data, which in our case is restricted to 2D biaxial tests. The plate is subjected both to longitudinal and transversal compression stresses LF = 5.2 MPa and TF = 2.6 MPa, respectively, and

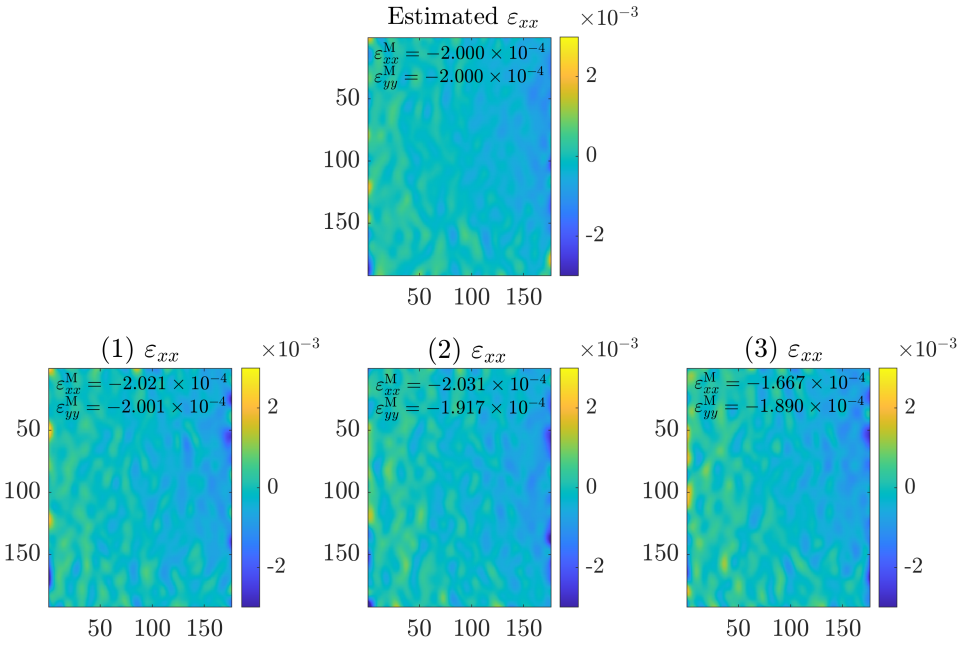


(a) Longitudinal strain, ε_{xx}.

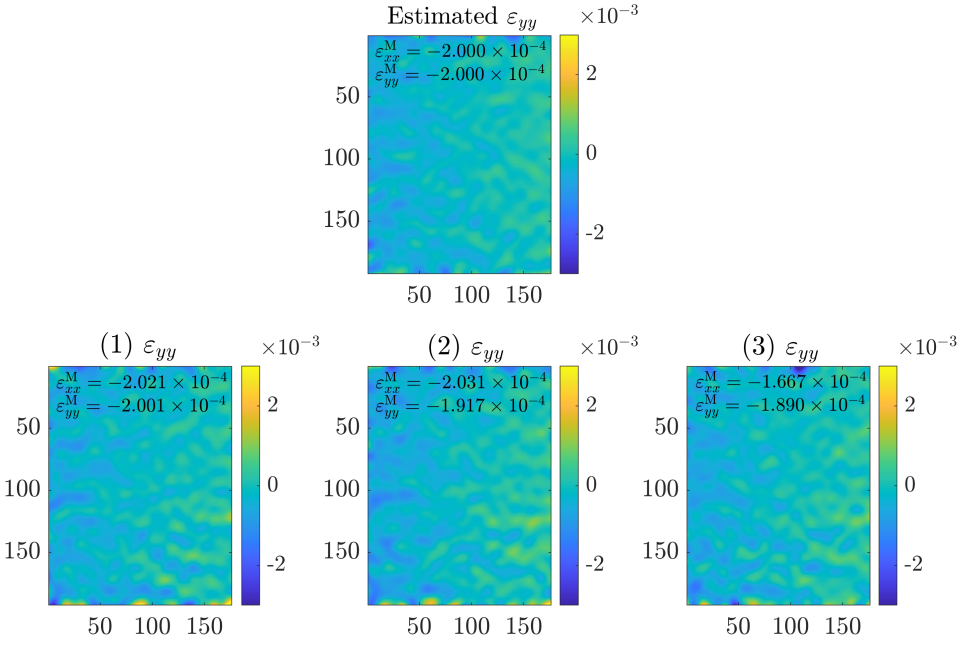


(b) Transversal strain, ε_{yy}.

Figure 4.12: Original and upsampled strain fields. The reconstructed data shows a good agreement with the original one, specially close to the sampled data.



(a) Microscopic longitudinal strain, ε_{xx}.



(b) Microscopic transversal strain, ε_{yy}.

Figure 4.13: Reconstructed microscopic strain fields for a given macroscopic data. The reconstructed microscopic strain fields are compared with the three closest microscopic strain fields from the dataset, (1), (2) and (3), which gives an idea of the information used for the reconstruction.

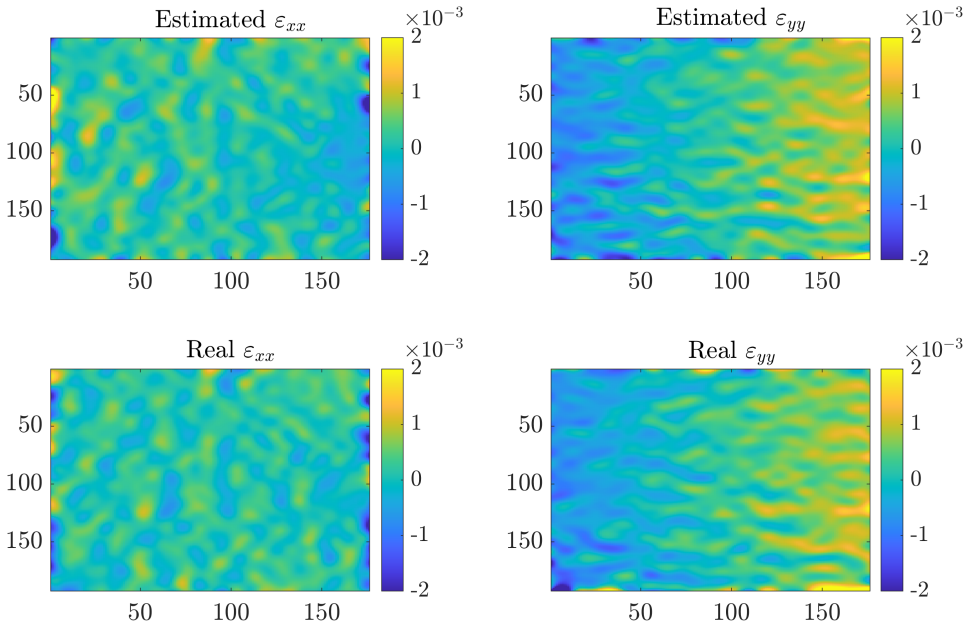


Figure 4.14: Reconstructed microscopic longitudinal strain field associated with the macroscopic one. The microscopic field reconstructed from the macroscopic state ($\varepsilon_{xx} = -1.693 \times 10^{-4}$, $\varepsilon_{yy} = -1.625 \times 10^{-5}$) is compared with the real and removed one from the original data-set.

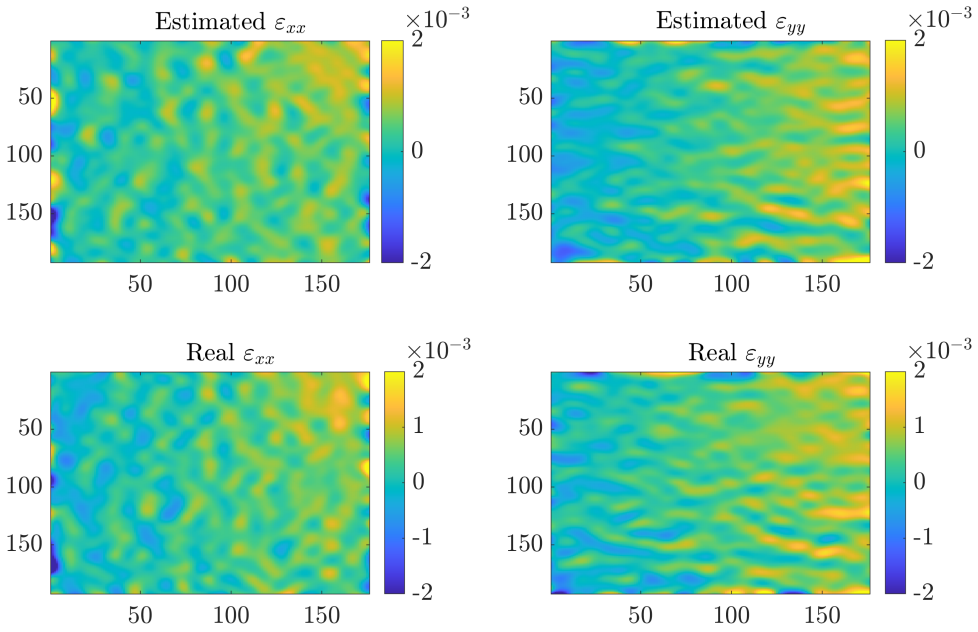


Figure 4.15: Reconstructed microscopic transversal strain field associated with the macroscopic one. The microscopic field reconstructed from the macroscopic state ($\varepsilon^{\mathrm{M}}_{xx} = -4.447 \times 10^{-4}$, $\varepsilon^{\mathrm{M}}_{yy} = -3.712 \times 10^{-5}$) is compared with the real and removed one from the original data-set.

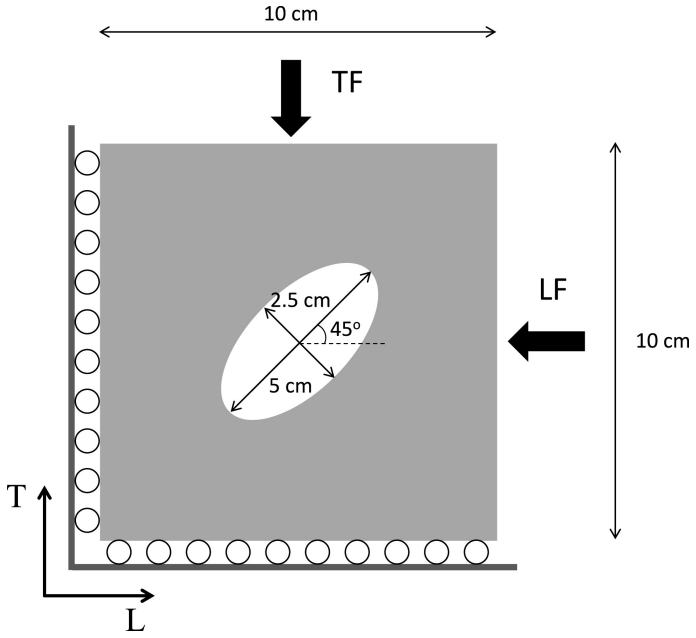


Figure 4.16: Geometry of the DD multiscale simulation. The geometry, applied forces and boundary conditions are illustrated, where L is associated to the x axis and T to the y axis.

to prescribed normal displacements as described in Fig. 4.16. The FE mesh was composed of 1240 triangular linear elements.

The macroscopic longitudinal and transversal components of the displacement field are shown in Fig.4.17. Moreover, the macroscopic longitudinal and transversal components of the strain and stress tensors are shown in Figs. 4.18 and 4.19, respectively.

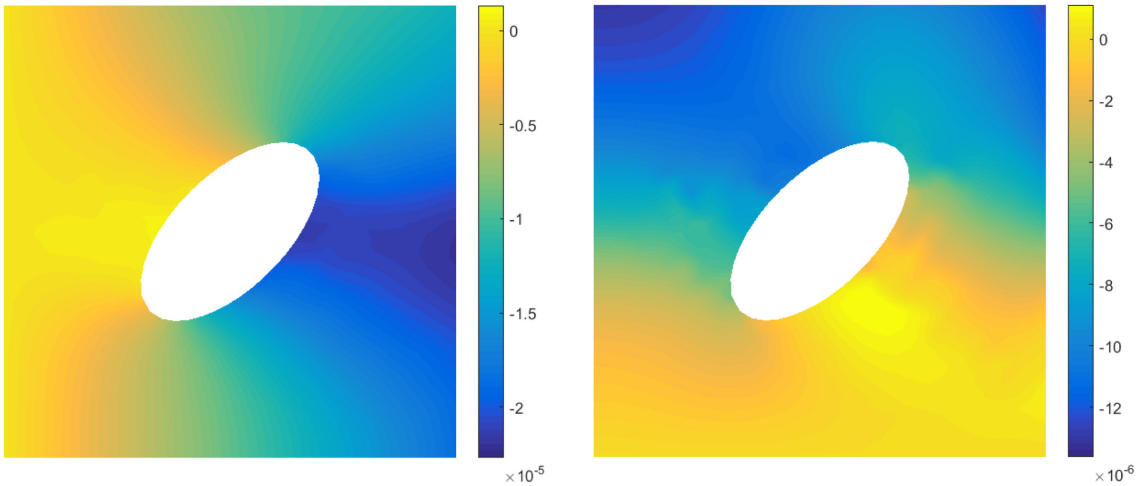


Figure 4.17: Macroscopic displacement field (in m). At the left panel the macroscopic longitudinal displacement and at the right panel the macroscopic transversal displacement are illustrated.

Multiscale results are presented along the microstructure, i.e. RVE, at three selected points in the macroscale. The distribution of the microscopic strain field in the RVE

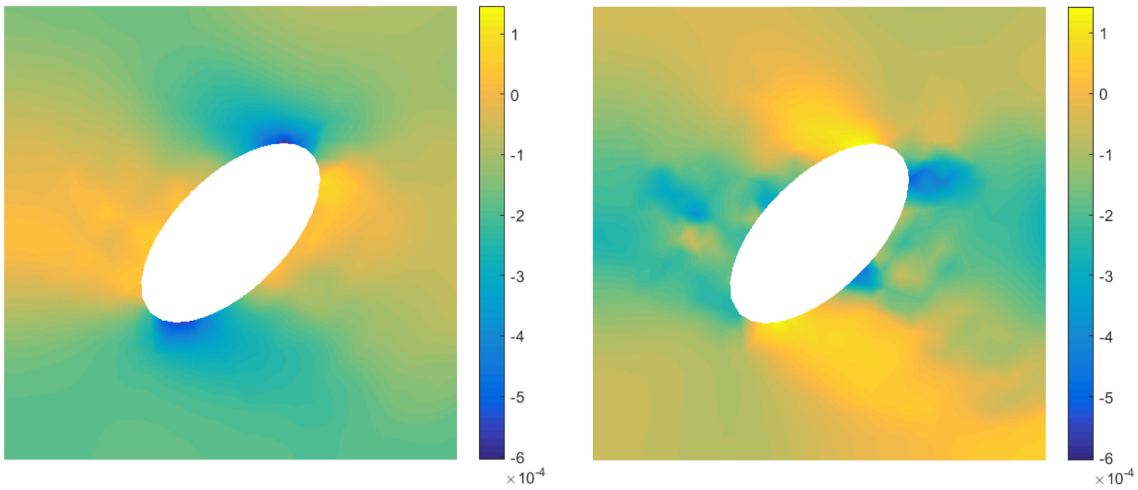


Figure 4.18: Macroscopic strain field. At the left panel the macroscopic longitudinal strain and at the right panel the macroscopic transversal strain are illustrated.

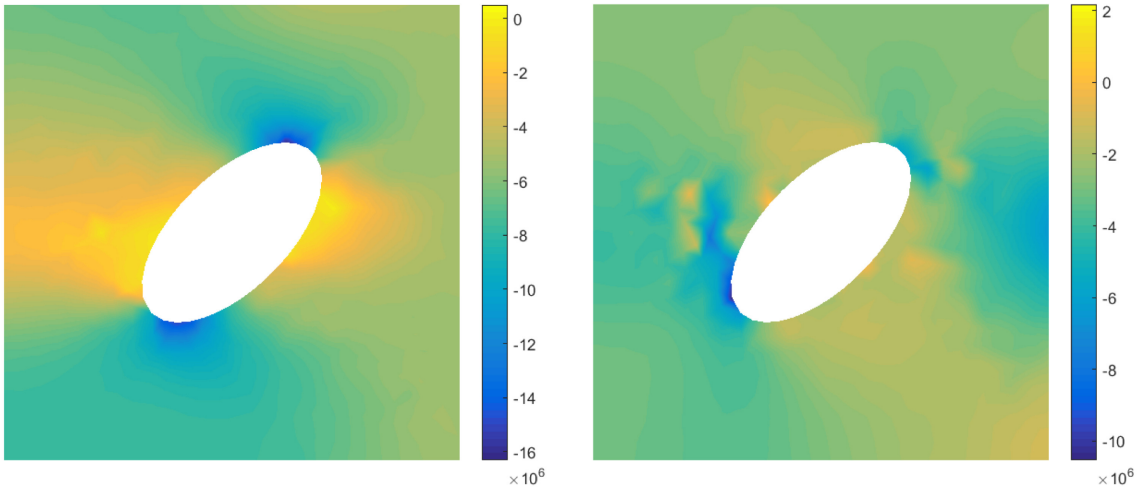


Figure 4.19: Macroscopic stress field (in MPa). At the left panel is illustrated the macroscopic longitudinal stress and at the right panel the macroscopic transversal stress.

represents the fluctuation of the strain field as a consequence of the heterogeneity of the domain at this scale. Fig. 4.20, Fig. 4.21 and Fig. 4.22 show three points in the macroscale with the associated macroscopic strain fields and their underlying microstructural distribution of the corresponding strain fields in the RVE.

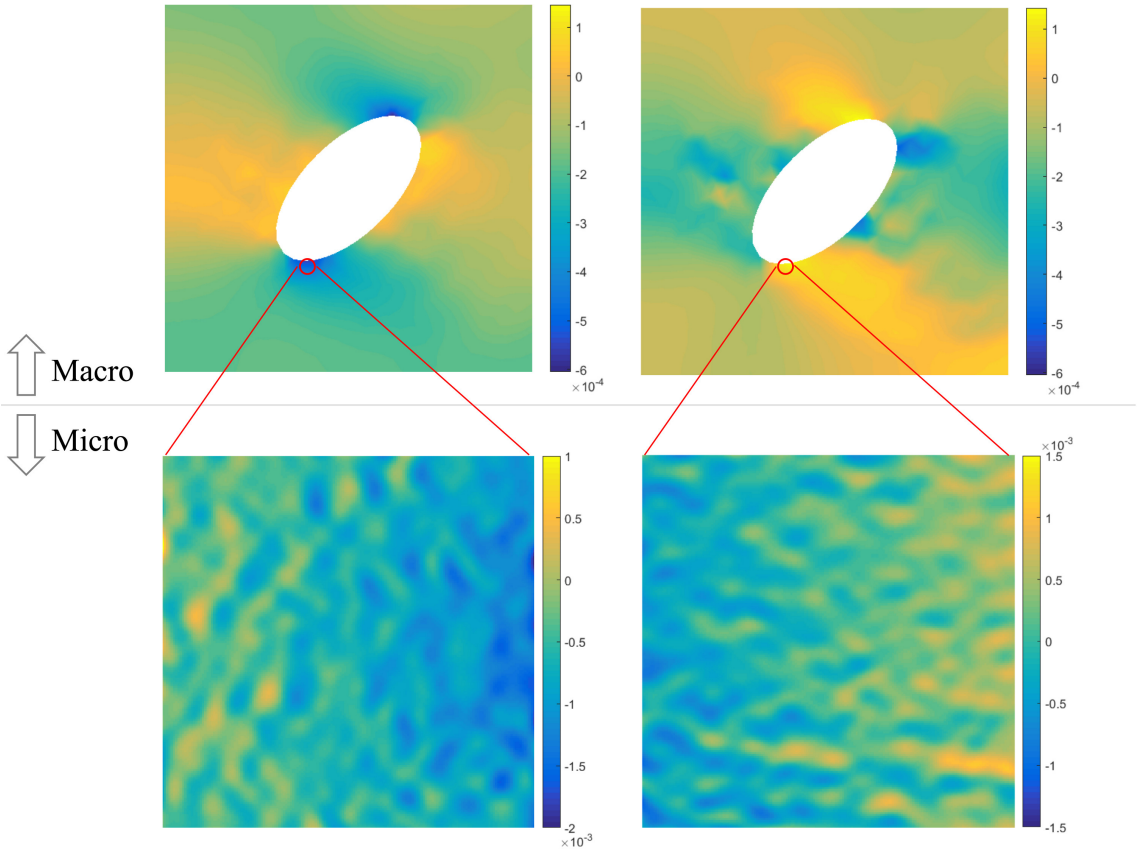


Figure 4.20: Multiscale results (1). Top: Macroscopic strain field [–] of the example of application5. Bottom: Microscopic strain field in the RVE for the highlighted point at the macroscale. Left: longitudinal strain component. Right: transversal strain component.

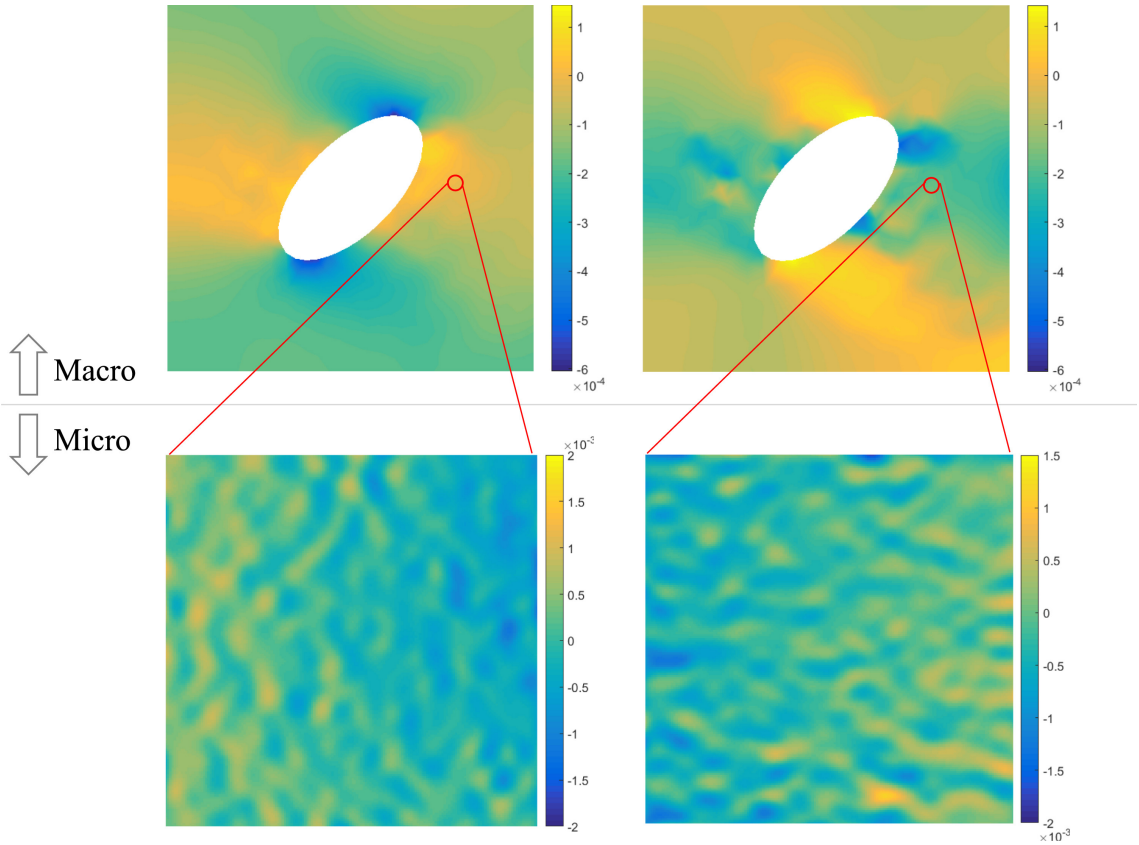


Figure 4.21: Multiscale results (2). Top: Macroscopic strain field [–] of the example of application. Bottom: Microscopic strain field in the RVE for the highlighted point at the macroscale. Left: longitudinal strain component. Right: transversal strain component.

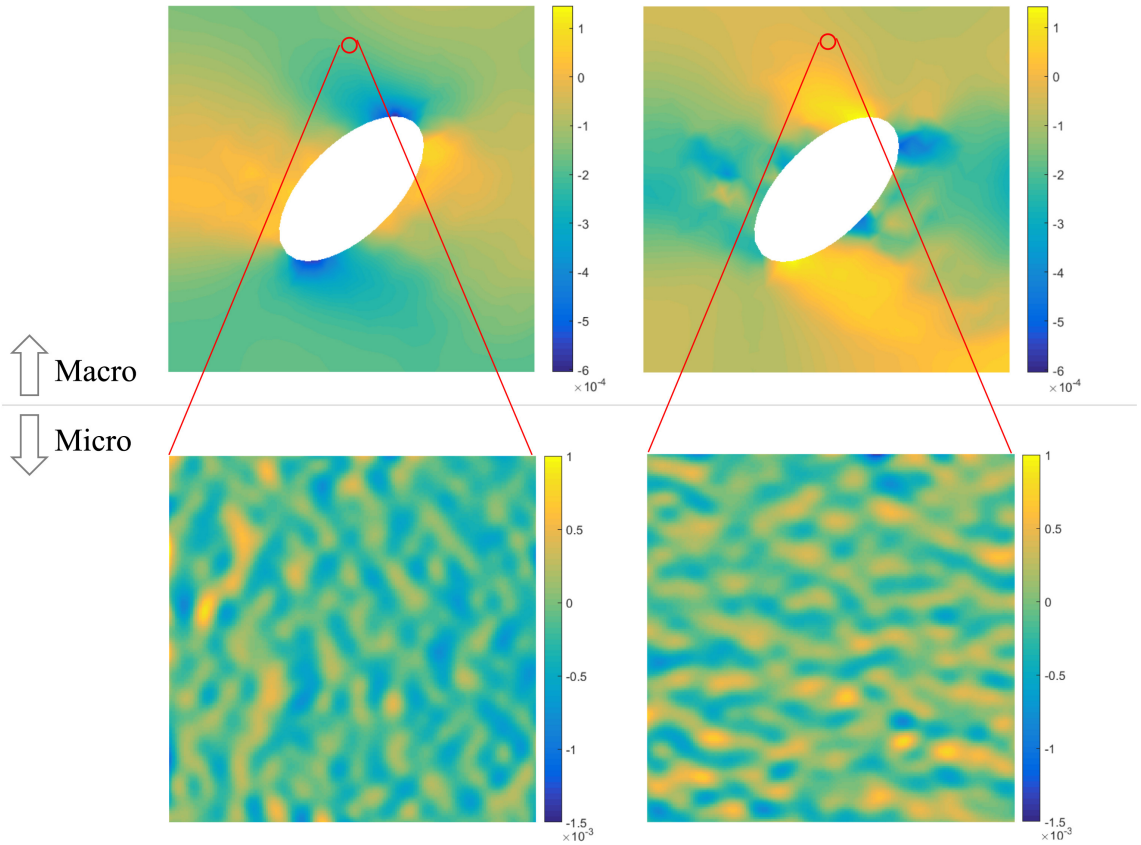


Figure 4.22: Multiscale results (3). Top: Macroscopic strain field [–] of the example of application. Bottom: Microscopic strain field in the RVE for the highlighted point at the macroscale. Left: longitudinal strain component. Right: transversal strain component.

4.5 Discussion and conclusions

In this chapter, a new completion data method has been presented, adapted to the DDSBES framework. The method can be seen as a generalisation of the classical mean imputation. Indeed, the presented method works when each of the data points is constrained to an oblique or even nonlinear manifold, whereas the mean imputation considers the points in the canonical coordinate manifolds. The presented method is based on the definition of a generalised weighted mean as the solution of the constrained minimisation problem:

$$\min_{\boldsymbol{x}\in\mathcal{M}} \sum_{j=1}^{N} w(\mathcal{M}_j) d^2(\boldsymbol{x}, \mathcal{M}_j). \tag{4.46}$$

Here, $\mathcal{M}$ is the manifold of points fulfilling a certain underlying structure, and $\mathcal{M}_j$, $j = 1, \ldots, N$ represent the incomplete data-sets. When $\mathcal{M} = \mathbb{R}^n$ and $\mathcal{M}_j = \{\boldsymbol{p}_j\}$, that is, points, we recover the usual definition of the mean. With this generalisation, the mean can be defined in non-Euclidean frameworks and the imputation of the missing values can be consistent with complex geometrical structures.

However, the presented method is more than a simple imputation method. The abstract framework in which it is formulated facilitates its use in combination with a DD approach for the resolution of simulation-based problems when the data used to feed the DD algorithm are incomplete. The physical interpretation of the data, in terms of state variables belonging to a given local structure (physical manifold), is compatible with the generalised defined imputation. Moreover, if some global physical conditions must be fulfilled, that is, state variables are embedded in a more manageable space and/or the data involve uncertainty, a weighting strategy is proposed in order to take all these considerations into account.

It has been shown that the presented imputation method, though it is used in the usual framework (with the canonical coordinate manifolds), improves the classical imputation approaches when the desired prediction can be stated in a framework including some extra constraints. The first example, of a pure Data Science nature, illustrates how the weighting strategy can be used to quantify the admissibility of a point in the imputation method. The second example illustrates how the presented methodology incorporates the physics of the problem to computations. It is pointed out that the method may be used for any physical problem where some fundamental physical constraints are invoked in combination with experimental measurements. Here, the method provides an alternative to highly demanding computational solutions based on numerical procedures, when experimental measurements can be easily obtained. The presented algorithm takes into account the physical quality of the data and, therefore, is more robust in problems with experimental bias. The last example, much richer, illustrates the full power of the presented method. Here, all features of the methodology are taken into account for the solution of a model-free approach to a multiscale problem in bone tissue biomechanics. First, a DD approach has been possible thanks to the data augmentation procedure developed, based on the algorithms described in this chapter. Besides, multiscale results give access to the microscopic strain distribution in the RVE. Fig. 4.20, Fig. 4.21 and Fig. 4.22

show that the peak fluctuation of the strain fields in the finer (microscopic) scale is an order of magnitude higher than the averaged (homogenised) strain fields. Since many of the physiological processes in the bone tissue are microstructural and local, including microcracking, bone fracture, and bone microarchitectural disorders (as menopause disease); information of this mechanical variable at the microscopic level is very important when analysing the referred phenomena with mathematical models. Moreover, it has been proposed that microstrains are the driving force in many remodelling theories (Claes and Heigele, 1999; Cowin, 2002; Isaksson et al., 2007). Therefore, this multiscale approach may be useful to quantify mechanical variables at this level. Indeed, the presented multiscale DD approach may be seen as a remedy to intractable large size multiscale FE methods, since the solution at the microscale is obtained as a post-process step, enabling important savings in the computational resources, when Eq. (4.45) is fulfilled.

The filling data methodology presented is conceptually simple, because it is based on the minimisation problem (4.46). However, this constrained minimisation problem is in general nonlinear and not always smooth. Here, a computational expression for the solution of (4.46) is derived in the linear case and an iterative algorithm is presented for nonlinear problems, based on tangent linearisation, the application of the linear solution and a standard strategy for returning to the manifold. No mathematical results are presented in this work about the convergence of the presented algorithm, which is crucial, and depends on the geometry of the problem, particularly the convexity of the data manifolds. Fortunately, the existence of extensive software for solving constrained optimisation problems can save this inconvenience in many problems, but the selected solution method would be, also, context dependent.

To conclude, the presented imputation method is a starting point for a new domain in Engineering, which responds to the need of DDSBES, that is, the adaptation of the classical statistical tools to engineering problems where some physics defines the geometric structure of the problem and has to be fulfilled. This domain, which could be named as data-driven simulation based statistics (DDSBS) is no more than the meeting point between Mathematical Physics, whose mathematical language is differential geometry, and Statistics, whose mathematical language is measure theory.

Scientific outcomes As a result of this work, we published the following scientific articles:

- Ayensa-Jiménez J., Sanz-Herrera, J. A., Doweidar M. H., Doblaré M. "An unsupervised data completion method for physically-based data-driven models", *Computer Methods in Applied Mechanics and Engineering*, 2019.
- Mora-Macías J., Ayensa-Jiménez J., Reina-Romo E., Doweidar M. H., Domínguez J., Doblaré M., Sanz-Herrera J. A. "A multiscale data-driven approach for bone tissue biomechanics", *Computer Methods in Applied Mechanics and Engineering*, 2020.
- Sanz-Herrera J. A., Mora-Macías J., Ayensa-Jiménez J., Reina-Romo E., Doweidar M. H., Domínguez J., Doblaré M. "Data-Driven Computational Simulation in Bone Mechanics", *Annals of Biomedical Engineering*, 2021.

5. Physically-Guided Neural Networks with Internal Variables

Contents

5.1 Introduction

As discussed in Chapter 2, a new scientific paradigm is rising, based on our increasing ability to collect, store, analyse, and extract information from high volumes of data. This is due to the huge amount and variety of data available, the power of non-structured data preprocessing techniques, the increasing computing power with dedicated capacities (LeCun, 2019) and a new generation of software tools that simplify and optimise the construction of machine learning (ML) models and their training process, such as TensorFlow and Keras (Géron, 2019; Gulli and Pal, 2017), Theano (Bastien et al., 2012; Bergstra et al., 2011) or Pytorch (Paszke et al., 2019).

In previous chapters, we have seen that scientific knowledge has been progressively incorporated into many different ML approaches. This philosophy tries to overcome the paucity of data in many scientific problems, involving a large number of parameters and variables, which usually causes poor predictive capability of purely data-based approaches in problems far from the training set (Xue, 2019). Additionally, a physically-based model is not only useful for making predictions, but it is also expected to help gain knowledge by the interpretation of its structure, parameters, and mathematical properties. In fact, physical interpretability is, in many cases, at least as important as predictive performance. Accordingly, scientists are struggling to "whiten" the "black-box" inherent to current ML predictive algorithms (Shwartz-Ziv and Tishby, 2017; Xu et al., 2019).

In Chapter 3 and Chapter 4 we have seen that one possible solution to this downside is the combination of raw or processed data obtained from experimental tests with the scientific consistency and interpretability of sound physically-based models, a field coined as data-driven simulation-based engineering and sciences (DDSBES) (Ayensa-Jiménez et al., 2018; Kirchdoerfer and Ortiz, 2016b, 2017; Leygue et al., 2018; Stainier et al., 2019). However, one of the most important drawbacks in current data-driven (DD) approaches is the need for explicitly defining the cloud of experimental values that identifies the internal state model (e.g. material constitutive equations in solid mechanics) with a sufficient number of points in the whole range of interest (Ayensa-Jiménez et al., 2018). This forces us to perform extensive experimental campaigns that are costly in time and money and whose results rely on strong assumptions on the experimental model itself (e.g. uniform distribution of stresses in uniaxial tests), that cannot be overcome due to the non-observable (non-measurable) character of some of the state variables (e.g. stresses) (Sanz-Herrera et al., 2021).

An opposite perspective is to integrate physical knowledge into Data Science models, that is, to constrain the prediction domain of the standard data model by physical constraints. This approach has been coined as physically-informed data science (PIDS). Of course, this approach may be extended to any known relation between the input-output variables. One simple example of this addition of physical knowledge was presented in Chapter 4, to fill incomplete data-sets coming from experimental campaigns in a reliable way. Another more powerful approach is using physical knowledge to inform and improve the data prediction capability of artificial neural networks (ANNs) (Karpatne et al., 2017c; Lu et al., 2021b, 2019; Mao et al., 2020; Raissi, 2018; Raissi et al., 2019, 2020; Yazdani et al., 2020). However, in these works, the physical information was introduced directly as relations between the input and output layers. Only in Lu et al. (2019); Raissi et al. (2019) a first attempt was made to provide the network with some explanatory capacity by adding some of the parameters associated with the internal state model as free parameters adjusted during the training step.

Our aim is to extend such explanatory capability by establishing a general approach in which we distinguish between the universal physical laws and the internal state equations. The former are treated as constraints imposed by the particular Physics between neuron values in the ANN, with an appropriate topology, while the latter are derived as a direct ANN outcome. This will permit us to identify some of the internal neurons with internal (in general, non-observable) state variables. The objective of

this work is, therefore, to introduce this new methodology of physically-guided neural networks with internal variables (PGNNIV) to predict the input-output relation in a physical system from a sufficient set of data, as well as to infer knowledge on the system internal structure, but always considering the constraints imposed by the Physics. The corresponding ANN is trained only with observable data (e.g. displacements or velocities, forces, etc.), extracting the internal non-observable values (e.g. stresses) from the ANN structure. In this way, we relax as much as possible the internal state model[13], thus "letting the data speak for themselves" (Gould, 1981). Thinking this way, this latter explanatory approach would be seen as a particular DD algorithm, in which the state model is derived from the available data, while the universal laws are applied as constraints in the ANN, instead of solving the equivalent mathematical equations by means of appropriate numerical solvers.

Moving to continuum Physics (deformable solids and fluid mechanics, electromagnetism, energy and mass transport problems, etc.) the universal physical principles are written in terms of partial differential equations (PDEs), which are currently computationally solved by means of numerical methods (Larsson and Thomee, 2009; Ruas, 2016). These use a previous discretisation step in space, leading to an algebraic, in general non-linear, system, that is then solved by means of standard matrix manipulation. Also, for time-dependent (evolution) problems, another discretisation step in time is required to transform the time-continuum problem into a discrete one. This includes the selection of a suitable time integrator (i.e. Euler, Multi-step, Runge-Kutta, among many others) (Larsson and Thomee, 2009). Alternatively, automatic differentiation may be used (Griewank et al., 1989), as is done by Raissi et al. (2019) in their physics-informed neural networks (PINNs). In the PGNNIV approach, solving this problem is straightforward by working with the discretised version of the problem, assigning an internal neuron to each nodal variable.

Several reasons support the interest in using PGNNIV in continuum Physics. On the one hand, the multiple problems of interest in this area, in many disciplines. Secondly, the increasing use of images and videos as the main method for data provision, which is equivalent to having a continuous distribution of data in space, previously discretised (pixels in 2D images and voxels in 3D ones), and also in time (time frames in a video). In third place, the tremendous effort made in treating images and in the associated artificial intelligence (AI) tools to make predictions from such images (Anwar et al., 2018; McCann et al., 2017; Pujari et al., 2018; Rawat and Wang, 2017; Yoo, 2015). Many of the mathematical operators now used to treat images are easily extrapolated to the standard differential operations in continuum Physics, thus allowing PGNNIV to leverage all current possibilities in image treatment to predict the evolution of a physical system (predictive capacity of PGNNIV) as well as to extract information on its structure (explanatory capacity of PGNNIV). The similarity in the mathematical language of image treatment and image-based ANN predictions (convolutional filters) and of discretised continuous physical problems (discretised differential operators) may be exploited to find a common framework for both.

[13]Sometimes, however, it is better to constrain that model, what can be done by additional constraints on the internal physically-meaningful layers of the ANN.

The PGNNIV methodology provides a general and systematic way of addressing the two objectives of simulation: predictive capacity and explainability, but now using the full power of the immense amounts of data available, together with the extensive battery of well-contrasted physical principles derived throughout centuries. The methodology is presented first for general systems defined by algebraic equations, and is next adapted to continuum problems, where these equations are the result of a previous discretisation. This observation requires to be exploited in detail, as during the last decades, the ANN community has developed a broad family of architectures and machinery that may be correlated with the nature of different physical operators. Indeed, the similarities between PDE operators and those used in image treatment (e.g. convolutional neural network (CNN)) are highlighted.

In summary, in addition to presenting the PGNNIV methodology, we explain how to adapt it to continuum problems, showing its predictive capacity to get the input-output relation, as well as its unravelling (explanatory) ability to extract knowledge on the system internal structure, including the values of the internal state variables. And this, by using only observable (measurable) variables in the training set (here related to continuous distributions of data values in a spatial and/or temporal domain).

After this introduction, we state the fundamentals of the methodology and review several families of problems in which it can be applied. The general methodology is then adapted to study continuum Physics under the PGNNIV framework, that is, we reformulate the mathematical foundations of continuum Physics in the domain of ANNs. Both the fields and the operators are recast in a standard ANN language, as it is TensorFlow. Then, we apply the proposed methods to two problems:

- A simple although illustrative problem coming from hydraulics, with some of the physical driving phenomena (conservation laws) are assumed and some other are stated empirically (friction and turbulent losses). This example is extended in several ways (inclusion of variable geometry, enrichment of the physics with an internal state assumption etc.) to illustrate the flexibility of the methodology. Next, we discuss some relevant properties of the approach, comparing the results of the same example with similar unconstrained standard ANNs. We show that the presented methodology shows a better performance in terms of faster convergence, fewer data needs, data noise filtering and bias correction, and extrapolation capacity.
- A continuum problem, specifically the (possibly nonlinear) diffusion equation, to show the potential of the methodology to deal with PDEs. We show several validation examples where the methodology is fully illustrated and its performance is analysed when dealing with heterogeneous and nonlinear problems. The predictive and explanatory capacity of the methodology are here revealed. Finally, we perform some numerical experiments to demonstrate the performance and the main features of the methodology and how they depend on the training data-set size and noise level and on the ANN structure and capacity.

Next, we present a discussion of the possibilities, limitations and requirements of the approach and the chapter finishes with the main conclusions.

5.2 Mathematical formulation

5.2.1 Physically-Guided Neural Networks with Internal variables

5.2.1.1 Concept of Physically-Guided Neural Networks with Internal Variables

In the PGNNIV framework, the state variables of the system are identified with neurons of the ANN. Some of them (e.g. the observable variables) are directly related to the input or output layers while the rest (non-observable) correspond to specific internal layers. Following the formalism of this thesis, we can write the rate evolution equation as in Eqs. 2.8, that we repeat here for completeness:

$$\boldsymbol{F}(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{f}) = \boldsymbol{0}, \tag{5.1a}$$

$$\boldsymbol{G}(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{g}) = \boldsymbol{0}, \tag{5.1b}$$

$$\boldsymbol{H}(\boldsymbol{u}, \boldsymbol{v}) = \boldsymbol{0}. \tag{5.1c}$$

To predict the value of the essential variables (at time $t+\Delta t$ for the transient problem or the steady-state solution), in what is called the *direct problem*, we define the architecture of our PGNNIV according to the following elements:

1. **Output identification:** It is the variable we want to predict. For regular problems (*direct problem*), we identify the output layer with (some or all) the spatially discretised (if required) values of the essential field $\boldsymbol{u}$, other variables associated with the essential field, $\boldsymbol{\varepsilon} = \boldsymbol{E}(\boldsymbol{u})$ or some other quantity of interests (QoIs) $\boldsymbol{q} = \boldsymbol{Q}(\boldsymbol{u})$. The important point here is that these variables have to be measurable.
2. **Input identification:** It is the variable from which we want to make predictions. Usually, the input layer will correspond to known values such as the external stimuli, $\boldsymbol{f}$ and the natural boundary conditions $\boldsymbol{g}$.
3. **Prescribed internal layers:** We identify some predefined internal layers (PILs) associated with the internal and non-measurable state variables $\boldsymbol{v}$. The values of such neurons may be recovered after convergence, getting the values of the internal state variables from the solution of the system for a particular input. The difference between PILs and common internal layers is that mathematical constraints will be applied to the neurons of the former, *guiding* the learning process.
4. **Neural Network architecture:** The rest of the internal layers, connecting the input, output and PILs, follows the standard approach in neural networks (NN), so they are able to "discover" the complex relations hidden in the function $\boldsymbol{H}$.
5. **Universal physical constraints:** The universal laws stated in $\boldsymbol{F}$ and the constraints expressed by $\boldsymbol{G}$ are established in the ANN as constraints between input, output and internal layers. Thus, the ANN is built to learn the relationship $\boldsymbol{H}$, assumed unknown, and that will be denoted for that purpose as $\boldsymbol{v} = \mathsf{H}(\boldsymbol{u})$.
6. **State equation relaxation:** Since the interest of this methodology is both *to predict* new values of the variable $\boldsymbol{y}$ and *to unravel* the constitutive model $\boldsymbol{H}$, the latter is generally only partially known. That is, we may partially know its functional structure, or some of the associated parameters. Therefore, the model is replaced by the subnetwork $\boldsymbol{v} = \mathsf{H}(\boldsymbol{u})$. Some guidelines for the set-up of this H are given hereafter.

7. **Extra knowledge about the state equation:** It is also possible to add any additional information (for example, associated with the structure of $\boldsymbol{H}$). This may be done by including additional constraints between layers (e.g. model symmetries), by specifying the topology of the network H (e.g. sparsity), by defining a parametric state equation $\boldsymbol{v} = \boldsymbol{H}(\boldsymbol{u}; \boldsymbol{\lambda})$ (parametrisation) or even by stating an explicit state model $\boldsymbol{v} = \boldsymbol{H}(\boldsymbol{u})$. Indeed, the state model may be known completely or partially (e.g. the material is isotropic but the parameters are unknown). In those cases, additional constraints or conditions may be designed to include this knowledge. We will come back to it in Section 5.2.4.2 for some interesting examples.

We have so far described the *direct problem*. It is also possible to choose as input the essential variables $\boldsymbol{u}$ and as output the stimuli $\boldsymbol{f}$ and $\boldsymbol{g}$, in what we call the *inverse problem*, following a similar approach, even though this is a less common problem as, generally, we are interested in predicting the solution field $\boldsymbol{u}$.

Now, we can write the formal input-output formulation, $\boldsymbol{x} = \boldsymbol{Y}(\boldsymbol{y})$, with:

$\boldsymbol{x} = \boldsymbol{I}(\boldsymbol{u}, \boldsymbol{f}, \boldsymbol{g})$,	identification of the input variables $\boldsymbol{x}$,
$\boldsymbol{y} = \boldsymbol{O}(\boldsymbol{u}, \boldsymbol{f}, \boldsymbol{g})$,	identification of the output variables, $\boldsymbol{y}$,
$\boldsymbol{F}(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{f}) = \boldsymbol{0}$,	constraints associated with the universal laws,
$\boldsymbol{G}(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{g}) = \boldsymbol{0}$,	constraints associated with the boundary conditions,
$\boldsymbol{v} = \boldsymbol{H}(\mathbf{u})$,	model network, defined in terms of a given network architecture.

We have then defined the different ingredients:

- **Problem variables**:
  - $\boldsymbol{x}$ are the input variables, supplied in the training process.
  - $\boldsymbol{y}$ are the output variables, supplied in the training process.
  - $\boldsymbol{u}$ are the essential variables.
  - $\boldsymbol{v}$ are the internal state variables.
  - $\boldsymbol{f}$ are the external stimuli.
  - $\boldsymbol{g}$ are the boundary conditions.
- **Problem functions**:
  - $\boldsymbol{I}$ identifies the input variables.
  - $\boldsymbol{O}$ identifies the output variables.
  - $\boldsymbol{F}$ are constraints defining the universal laws of the problem, related to the physics of the problem.
  - $\boldsymbol{G}$ are constraints defining the boundary conditions of the problem, related also to the physics of the problem.
  - $\boldsymbol{H}$ are functions that establish the state equations relating the internal state variables $\boldsymbol{v}$ to the essential ones $\boldsymbol{u}$.

The combination of $\boldsymbol{F}$ and $\boldsymbol{G}$ provides the variables $\boldsymbol{v}$ with their physical meaning.

As a result of all these considerations, we may interpret a PGNNIV as the combination of two ANNs:

1. **A *predictive* network:** This network is able to *predict* the value of the variable $\boldsymbol{y}$ from the value of the variable $\boldsymbol{x}$:

$$\boldsymbol{y} = \mathsf{Y}(\boldsymbol{x}). \tag{5.2}$$

2. **An *explanatory* network:** This network is able to *unravel* the state equation relating the measurable variable $\boldsymbol{u}$ and the internal variable $\boldsymbol{v}$, that is, the geometry of the phase space $\mathcal{M} \times \mathcal{E}$:

$$\boldsymbol{v} = \mathsf{H}(\boldsymbol{u}). \tag{5.3}$$

Note that the data-set used for the learning process, $\mathcal{D} = \{(\boldsymbol{x}^i; \boldsymbol{y}^i)\}_{i=1,\dots,N}$ considers only a subset of the measurable variables, that is, $\boldsymbol{u}$, $\boldsymbol{f}$ and $\boldsymbol{g}$, and does not include the values of the internal and non-measurable variable $\boldsymbol{v}$.

As a result of the learning process, the relationship $\boldsymbol{u} \to \boldsymbol{v}$, given by Eq. (5.2), is learned, and since $\boldsymbol{I}$ and $\boldsymbol{O}$ are known, the relationship $\boldsymbol{x} \to \boldsymbol{y}$, given by Eq. (5.3), is also learned. Therefore, PGNNIV have both predictive and explanatory capacity.

It has to be remarked that any additional physical knowledge of the system that may be expressed in mathematical terms relating state variables and stimuli may be imposed in a similar way. For example, physical inequalities such as the second principle of thermodynamics or any inequality constraint on the parameters (e.g. positivity of the elastic modulus or the physical range for Poisson coefficient in an isotropic material $-1 < \nu < 0,5$ in solid mechanics). The details about the inequality implementation are given in the next section.

Several important facts must be highlighted about this approach:

- This representation opens a new paradigm in the characterisation of a given constitutive model. The model is not anymore characterised by an explicit functional relationship $\boldsymbol{v} = \boldsymbol{H}(\boldsymbol{u}; \boldsymbol{\lambda})$, depending on the field variables and additional parameters that have to be determined by classical fitting of a set of experimental tests (*parametric functional framework*), but by an ANN topology and the corresponding activation functions, $\boldsymbol{v} = \mathsf{H}(\boldsymbol{u}) = \boldsymbol{H}(\boldsymbol{u}; \boldsymbol{\Lambda})$, where $\boldsymbol{\Lambda}$ represents the weights and biases of the neural network that are tuned during the training process (*neural network framework*). Indeed, the universal approximation theorem guarantees that a regular enough function may be approximated by a specific neural network with a sufficient number of layers and neurons and convenient activation functions (Cybenko, 1989; Hanin, 2017; Hornik, 1991; Lu et al., 2017), so the second approach is, at least, as general as the first one (Pinkus, 1999). This compositional approach, instead of the additional one that characterises standard parametric fitting (Han et al., 2018), unfolds the benefits of the ANN hardware (fast computation with graphical processing units

(GPU) and tensorial processing units (TPU) (Jouppi et al., 2017), cloud and distributed computing...) and software such as Keras and TensorFlow (modularity, pluggability, fast generalisation capability,...). All this also allows for high performance computing capabilities and scalability (Strigl et al., 2010), a major model flexibility that allows capturing strong non-linearities (Lee et al., 1993; Sarle, 1994) and soundness with respect to statistical data (heteroscedasticity, non-normality...) (Guh, 2002; Matías et al., 2010; McAleer et al., 2008).

- This framework allows the scientist to work only with measurable variables and fields $\boldsymbol{u}$. This is fundamental, as internal (non-measurable) variables $\boldsymbol{v}$ are, indeed, mathematical constructs that are determined now as a byproduct of the predictive problem. In that sense, the proposed approach is not only valid as a parameter or output predictor, but also as a method for building theoretical frameworks based on a prescribed field theory (for instance, the continuum solid mechanics theory based on the field theory of internal stresses, $\boldsymbol{\sigma}$). A detailed formulation of the different classical field theories in continuum physics using PIDS is given in Chapter 2.
- The variables involved in the problem, that is, $\boldsymbol{u}$, $\boldsymbol{v}$, $\boldsymbol{f}$ and $\boldsymbol{g}$, may be spatial and/or temporal fields depending on the location $\boldsymbol{x}$ and/or time t. In those cases, we shall consider that a previous discretisation step has been applied, so the time-position independent interpolating variables are those of the associated discretised problem. Therefore, the same approach can be used both for non-transient problems or for transient ones, using as variables $\boldsymbol{u}$ the algebraic values that define the approximated field at a certain time and interpolation point, following a step-by-step continuation approach.
- If no constraint is applied and no PIL is defined, we recover the classical ANN framework. If the constraints are applied to the input or output layers, we recover the formulation developed by other authors for physically-guided neural networkss (PGNNs) (Karpatne et al., 2017a) or PINNs (Raissi et al., 2019) among other similar approaches, as the one exposed in Yadav et al. (2015). The work developed in Raissi et al. (2019) for PDEs, which is nowadays considered almost foundational, includes some explanatory capacity by parametrising the relation given by $\boldsymbol{H}$, in terms of some explanatory parameters related to *a priori* defined state models but without using the PIL concept, that is the seed of the constitutive equation's ANN framework, that is original of this thesis, up to the authors' knowledge.

In summary, this framework allows the scientist to work with directly measurable variables and fields, without the need for establishing any *a priori* assumption on the expression of the internal variables or the relation between them, unless otherwise desired. The approach is, therefore, model-free. This model-free character is *modular*, as the degree of knowledge may be included partially or totally, thus leveraging the information about the system. As special cases, we have the no information paradigm (standard ANN framework), the total information paradigm (classical simulation-based engineering and science (SBES) using PINNs), the inverse problem analysis (parametrisation of constitutive equations encoded in $\boldsymbol{H}$), model checking and structural analysis (comparison of different models for $\boldsymbol{H}$), among many other possibilities, as we will detail in Section 5.2.2.

5.2.1.2 Construction of the Physically-Guided Neural Network with Internal Variables

In this section, we build the PGNNIV from the known concept of ANN. Although nowadays there exist a lot of architectures of different complexity (see Sengupta et al. (2020); Shrestha and Mahmood (2019) for some interesting reviews), for the sake of simplicity we illustrate the construction of the PGNNIV starting from the structure of a multilayer perceptron (MLP), based on the original idea developed by Rosenblatt (1958), even if the procedure may be adapted to more complex architectures.

Having said that, and denoting the input variable $\boldsymbol{x}$ as $\boldsymbol{y}_0$, each hidden layer of n_i neurons, $\boldsymbol{y}_i$, $i = 1, \dots, L$ is defined by a functional relation:

$$\boldsymbol{y}_i = \phi(\boldsymbol{y}_{i-1}\boldsymbol{W}_i + \boldsymbol{b}_i), \tag{5.4}$$

where $\boldsymbol{W}_i$ and $\boldsymbol{b}_i$, $i = 1, \cdots, L$, are the weights and biases, the parameters of the model, and $\phi : \mathbb{R}^{n_i} \to \mathbb{R}^{n_i}$ is an activation function. With this notation, the output variable is $\boldsymbol{y} = \boldsymbol{y}_L$. The network is symbolically represented by the relationship $\boldsymbol{y} = \mathsf{Y}(\boldsymbol{x})$ or, denoting by $\boldsymbol{\Lambda}$ the whole set of weights and biases for a given network topology, $\boldsymbol{y} = \boldsymbol{Y}(\boldsymbol{x}; \boldsymbol{\Lambda})$. Given a set of *ground truth* data points $\mathcal{D} = \{(\bar{\boldsymbol{x}}^i, \bar{\boldsymbol{y}}^i) | i = 1, \cdots, N\}$, we use a loss function $\mathcal{L}$ to evaluate the network performance, for instance, the mean square error (MSE)[14]:

$$\mathcal{L}(\boldsymbol{\Lambda}|\mathcal{D}) = \mathrm{MSE}(\boldsymbol{\Lambda}|\mathcal{D}) = \frac{1}{N}\sum_{i=1}^{N} \left\| \bar{\boldsymbol{y}}^i - \boldsymbol{Y}(\bar{\boldsymbol{x}}^i; \boldsymbol{\Lambda}) \right\|^2. \tag{5.5}$$

Now, it is possible to train the neural network by minimising the loss function $\mathcal{L}$, getting the optimal set of weights and biases $\boldsymbol{\Lambda}$. This is what standard ANNs do. Let us now impose some constraints such that some neuron values satisfy some (physically-based) equations, including universal laws in manifold constraints and boundary conditions. These constraints are formulated in terms of functions F_j, G_j, H_j as:

$$F_j(\boldsymbol{y}_0, \dots, \boldsymbol{y}_L) = \boldsymbol{0}, \quad j = 1, \dots, r_F, \tag{5.6a}$$
$$G_j(\boldsymbol{y}_0, \dots, \boldsymbol{y}_L) = \boldsymbol{0}, \quad j = 1, \dots, r_G, \tag{5.6b}$$
$$H_j(\boldsymbol{y}_0, \dots, \boldsymbol{y}_L) = \boldsymbol{0}, \quad j = 1, \dots, r_H. \tag{5.6c}$$

Recall that F_j, $j = 1, \dots, r_F$, define the universal physics on the ambient manifold, G_j, $j = 1, \dots, r_G$, are related with boundary conditions and H_j collect the partially or totally known constitutive model information. Without loss of generality, we denote all these functions by $\boldsymbol{R}_j, j = 1, \dots, r$, with $r = r_F + r_G + r_H$:

$$\boldsymbol{R}_j(\boldsymbol{y}_0, \dots, \boldsymbol{y}_L) = \boldsymbol{0}, \quad j = 1, \dots, r. \tag{5.7}$$

We can then adapt the minimisation problem to:

$$\min_{\boldsymbol{\Lambda}} \quad \mathcal{L}(\boldsymbol{\Lambda}|\mathcal{D}) \qquad \text{s.t.} \qquad \boldsymbol{R}_j(\boldsymbol{\Lambda}|\mathcal{D}) = 0, \quad j = 1, \dots, r. \tag{5.8}$$

[14] Of course, it is possible to choose another regression loss function such as the mean absolute deviation (MAD) or mean square logarithmic error (MSLE).

With this formulation, $\boldsymbol{x} = \boldsymbol{I}(\boldsymbol{y}_0, \ldots, \boldsymbol{y}_L) = \boldsymbol{y}_0$ and $\boldsymbol{y} = \boldsymbol{I}(\boldsymbol{y}_0, \ldots, \boldsymbol{y}_L) = \boldsymbol{y}_L$. The variables associated with $\boldsymbol{u}$, $\boldsymbol{v}$, $\boldsymbol{f}$ and $\boldsymbol{g}$ are automatically determined by the nature and structure of $\boldsymbol{R}$, that is context dependent, so that:

$$\boldsymbol{u} = \boldsymbol{u}(\boldsymbol{y}_0, \ldots, \boldsymbol{y}_L), \tag{5.9a}$$
$$\boldsymbol{v} = \boldsymbol{v}(\boldsymbol{y}_0, \ldots, \boldsymbol{y}_L). \tag{5.9b}$$

Thus, the explanatory network $\boldsymbol{v} = \mathsf{H}(\boldsymbol{u})$ is implicitly defined by considering the dependence $\boldsymbol{u} \to \boldsymbol{v}$.

If some of the constraints are expressed by an explicit equation, we can modify the problem (5.8) to make this constraint disappear from the general formulation. This is easily done by explicitly defining part of the architecture of the network without the inclusion of learning parameters. Fig. 5.1 illustrates a PGNNIV for a three hidden-layered neural network.

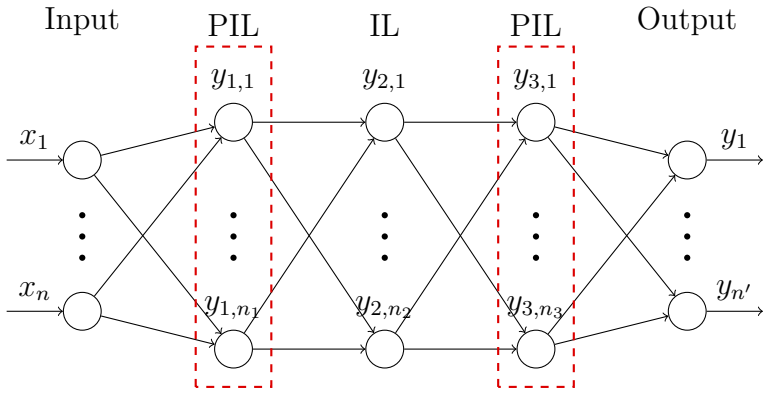


Figure 5.1: Physically-Guided Neural Network for a three hidden-layered network. The red dashed rectangles indicate the neurons in which a certain constraint is applied (PILs). The number of internal layers between the input layer and layer 1, layer 1 and layer 3, and between layer 3 and the output layer can be increased to allow more complex models. It is the physical constraint that provides PILs 1 and 3 with a physical interpretation as internal state variables.

It is possible to reformulate (5.8) using a penalty approach:

$$\min_{\boldsymbol{\Lambda}} \quad \mathcal{L}(\boldsymbol{\Lambda}|\mathcal{D}) + \mathcal{L}_{\text{PEN}}(\boldsymbol{\Lambda}|\mathcal{D}), \tag{5.10}$$

with $\mathcal{L}_{\text{PEN}}$ verifying that $\mathcal{L}_{\text{PEN}}(\boldsymbol{\Lambda}) = \boldsymbol{0}$ if $\boldsymbol{R}_j(\boldsymbol{\Lambda}) = \boldsymbol{0}$. For instance, we may define again a quadratic loss function and a Gramm (scalar product) matrix $\mathbf{G}$:

$$\mathcal{L}_{\text{PEN}}(\boldsymbol{\Lambda}) = \boldsymbol{R}(\boldsymbol{\Lambda}|\mathcal{D})^{\intercal}\mathbf{G}\boldsymbol{R}(\boldsymbol{\Lambda}|\mathcal{D}). \tag{5.11}$$

In particular, we can define r penalty parameters p_j and state:

$$\mathcal{L}_{\text{PEN}}(\boldsymbol{\Lambda}) = \sum_{j=1}^{r} p_j \|\boldsymbol{R}_j(\boldsymbol{\Lambda}|\mathcal{D})\|^2, \tag{5.12}$$

where p_j, $j = 1, \ldots, r$ are penalty parameters. This approach allows the implementation of the problem in a standard ANN framework (i.e. TensorFlow@Python) by just defining an adapted loss function that includes the penalty term $\text{OF} = \mathcal{L} + \mathcal{L}_{\text{PEN}}$.

Note that Eq. (5.10) may be interpreted as an auxiliary ANN with input $\boldsymbol{x}$ and output $\hat{\boldsymbol{y}} = (\boldsymbol{y}; \boldsymbol{y}_{\text{PEN}})$, being $\boldsymbol{y}_{\text{PEN}}$ a new set of output variables, $\boldsymbol{y}_{\text{PEN}} = (\boldsymbol{R}_1, \ldots, \boldsymbol{R}_r)$ with physical meaning, whose *ground-truth* value is always 0, that is, $\bar{\boldsymbol{y}}_{\text{PEN}} = \boldsymbol{0}$, as illustrated in Fig. 5.2.

The inclusion of inequalities in the presented framework is possible using the rectified linear unit (ReLU) function. Indeed, the inclusion of a term in the penalty function with the structure $p\text{ReLU}(f(\boldsymbol{\Lambda}|\mathcal{D}))$ guarantees that if p is high enough, $\text{ReLU}(f(\boldsymbol{\Lambda}|\mathcal{D}))$ has to be the smallest possible, ensuring that $\text{ReLU}(f(\boldsymbol{\Lambda}|\mathcal{D})) \to 0$ and therefore $f(\boldsymbol{\Lambda}|\mathcal{D}) \leq 0$.

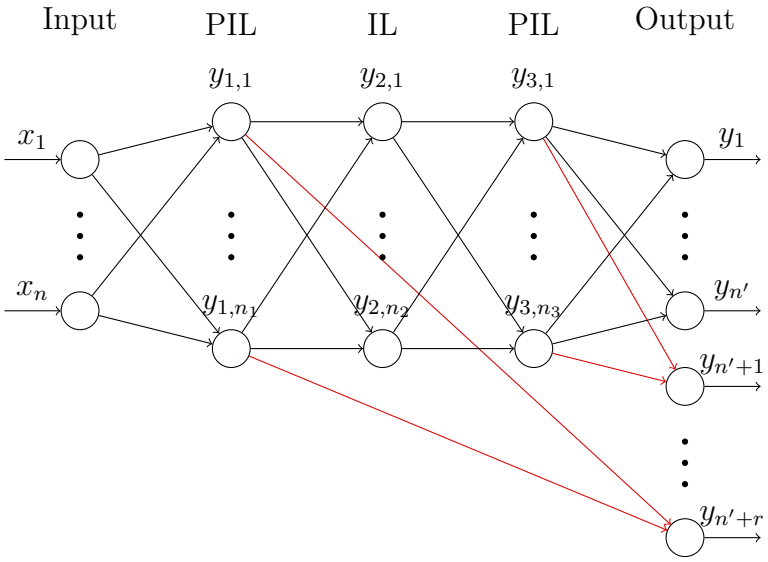
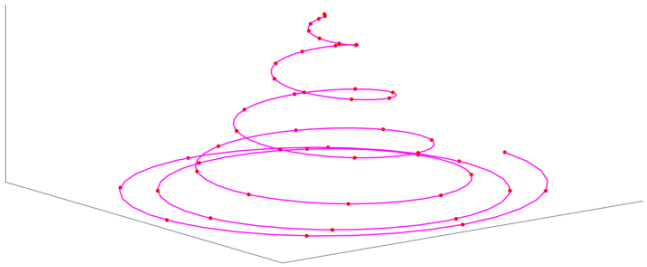


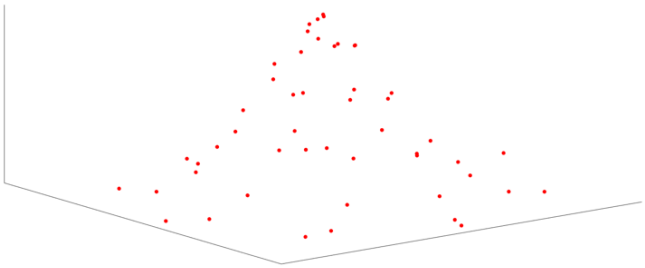
Figure 5.2: Augmented Neural Network equivalent to the Physically-Guided Neural Network. Each constraint is replaced by an extra output representing the value of the constraint that, ideally, should be null.

This approach has two main advantages that match the two spearheads against ANN methods:

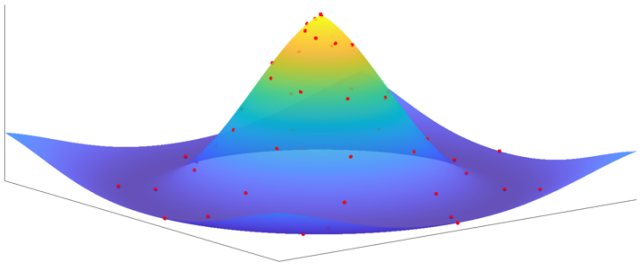
1. From a physical point of view, we postulate some extra conditions onto the hidden variables, which allow us to interpret them as true physically-based features, that is, as state variables of the physical problem, overcoming the *black-box* problem of neural networks (Castelvecchi, 2016; Papernot et al., 2017; Samek et al., 2017) and placing this work within the framework of explainable artificial intelligence (XAI) (Adadi and Berrada, 2018; Emmert-Streib et al., 2020).
2. As the search space is reduced via constraints, the optimisation algorithm is expected to learn faster, with less information, to filter the noise incompatible with the problem physics and to discard solutions without physical sense. Fig. 5.3 illustrates the idea behind this argument.



(a) Sampled data and latent structure.



(b) Data without physical structure.



(c) Data with physical structure.

Figure 5.3: Geometrical idea behind the PGNNIV performance improvement: Suppose that the aim is to discover the latent structure (magenta line) of the data (red dots). The data is supposed to fulfil the problem physics, represented by the surface, so the search space has indeed a lower dimension ($d = 2$) than the ambient space ($d = 3$).

5.2.2 Types of problems in which Physically-Guided Neural Networks with Internal variables may be applied

The idea is, therefore, to build an ANN that explicitly includes the PILs associated with the internal state variables and the known physical constraints. This network will be trained by a set of data to derive the relations and, from them:

- To predict a set of output measurable variables from the input measurable ones.
- To predict the values of the internal non-measurable variables.
- To get knowledge on the internal relations that define the constitutive model itself and even to obtain an implicit model from the relations between PILs or to derive the value of some structural parameters related to the constitutive model.

We can think, therefore, of several families of problems, namely:

1. **Prediction problems:** The goal here is to predict the value of a given dependent output variable $\boldsymbol{y}$ from other independent measurable one $\boldsymbol{x}$. The material constitutive model or state equation is assumed to be frozen and, therefore, there exists an unknown relationship, $\boldsymbol{H}$, whose functional form or properties have to be revealed.

   - **Direct problem:** Here we are interested in defining the output $\boldsymbol{y}$, that is, the dependent variable of the model $\boldsymbol{y} = \mathsf{Y}(\boldsymbol{x})$, usually identified with the essential variable $\boldsymbol{u}$, which corresponds to the solution field (continuum) or variable (discrete) in most physical problems:

     $$\boldsymbol{y} = \boldsymbol{u}, \quad \text{output definition.} \tag{5.13}$$

     Similarly, the input variable $\boldsymbol{x}$ will be the independent variable of the predictive model $\boldsymbol{y} = \mathsf{Y}(\boldsymbol{x})$. For a wide range of problems, it corresponds to the value of the source term (or external stimuli) $\boldsymbol{f}$ together with the boundary conditions $\boldsymbol{g}$. For evolution problems, the values of the fields and variables at the current time t (memoryless problems) or their whole history until time t (problems with memory) have to be supplied as input too and are incorporated as part of the variables $\boldsymbol{f}$:

     $$\boldsymbol{x} = (\boldsymbol{f}, \boldsymbol{g}), \quad \text{input definition.}$$

     Therefore, in our framework:

     $$\boldsymbol{I}(\boldsymbol{u}, \boldsymbol{f}, \boldsymbol{g}) = (\boldsymbol{f}, \boldsymbol{g}), \tag{5.14a}$$
     $$\boldsymbol{O}(\boldsymbol{u}, \boldsymbol{f}, \boldsymbol{g}) = \boldsymbol{u}. \tag{5.14b}$$

   - **Inverse problem:** Now the output variable $\boldsymbol{y}$ is the external stimulus:

     $$\boldsymbol{y} = \boldsymbol{f}, \tag{5.15}$$

and the input variable $\boldsymbol{x}$ the solution field. As in that case, since we know the essential field $\boldsymbol{u}$, the specification of the essential boundary conditions $\boldsymbol{g}$ is redundant:

$$\boldsymbol{x} = (\boldsymbol{u}, \boldsymbol{g}) = \boldsymbol{u}. \tag{5.16}$$

Therefore:

$$\begin{aligned} \boldsymbol{I}(\boldsymbol{u}, \boldsymbol{f}, \boldsymbol{g}) &= \boldsymbol{u}, \\ \boldsymbol{O}(\boldsymbol{u}, \boldsymbol{f}, \boldsymbol{g}) &= \boldsymbol{f}. \end{aligned} \tag{5.17a}$$

Mixed formulations can be defined between the direct and the inverse problems, assuming the problem is well-posed.

When solving prediction problems, many approaches can be followed:

(a) **Pure predictive problem**. We establish a direct correspondence between observable variables (e.g. prediction of the displacements from the external forces or *viceversa*) for a fixed system (same geometry and internal structure), without any constraint or explicit establishment of PILs. In this case, we do not impose any constraint and use the ANN in the standard "black-box" manner to get the correlation between the input and output variables to predict, after training, the latter for a particular input, without any knowledge of the physical system. The objective is, therefore, to solve with the ANN the implicit relation $\boldsymbol{R}(\boldsymbol{u}, \boldsymbol{v}(\boldsymbol{u}), \boldsymbol{g}, \boldsymbol{f}) = \boldsymbol{0}$, where the input and output variables will depend on whether we consider the direct or inverse problem:

$$\text{GET} \quad \boldsymbol{y} = \mathsf{Y}(\boldsymbol{x}), \tag{5.18}$$

where Y is the representation of some appropriate ANN (architecture, connectivity and metaparameters). This has been done frequently in the last decades (see Hambli et al. (2006); Pathak et al. (2005); Thibault and Grandjean (1991) or Chapter 2 for more examples) using both real and synthetic (*in silico*) data-sets. As an illustrative example, this is the case in solid mechanics when relating directly the applied forces to the displacements.

(b) **Pure predictive problem with input-output constraints**. The only difference with the above is the assumption (or knowledge) of some input-output relations. Those relations are imposed via external constraints onto the objective function of the ANN via penalty terms without using the concept of internal state variables or PILs:

$$\begin{aligned} &\text{GET} \quad \boldsymbol{y} = \mathsf{Y}(\boldsymbol{x}), \\ &\text{s.t.} \quad \boldsymbol{R}(\boldsymbol{x}, \boldsymbol{y}) = \boldsymbol{0}, \end{aligned} \tag{5.19}$$

being $\boldsymbol{R}$ a prescribed function that relates the input and output layers and encodes the problem physics. Some particular problems have been solved using this methodology (Karpatne et al., 2017c), and as specially important is the case of solving PDEs, when we identify $\boldsymbol{x}$ with the spatial coordinate and $\boldsymbol{y} = \boldsymbol{u}(\boldsymbol{x})$ is the solution field (Raissi et al., 2019).

(c) **Pure predictive problem with internal derived variables**. In this case, we use a particular PIL associated with rates and gradients of the essential variable (e.g. strains, velocities, temperature change rate) and include their definition $\boldsymbol{\varepsilon} = \boldsymbol{\varepsilon}(\boldsymbol{u})$ as constraints between such layer and the measurable one ($\boldsymbol{u}$), thus helping the ANN to "know" that the dependence between the output and input is through those intermediate variables. For instance, this is the case in solid mechanics when relating directly the applied forces to the strains, defined in terms of the displacement gradient, leaving free the material constitutive model. The formulation of the problem is now:

$$\begin{aligned} &\text{GET} \quad \boldsymbol{y} = \mathsf{Y}(\boldsymbol{\varepsilon}, \boldsymbol{x}), \\ &\text{s.t.} \quad \boldsymbol{\varepsilon} = \boldsymbol{E}(\boldsymbol{u}), \end{aligned} \tag{5.20}$$

where $\boldsymbol{E}$ is a prescribed function (obtained from the discretisation of a functional) that will improve the learning capability of Y. The relationship $\boldsymbol{\varepsilon} = \boldsymbol{E}(\boldsymbol{x})$ may be stated explicitly or via the constraint $\boldsymbol{R} = \boldsymbol{\varepsilon} - \boldsymbol{E}(\boldsymbol{x}) = \boldsymbol{0}$.

(d) **Predictive problem with internal hidden variables and physical constraints**. We add, instead, a PIL with the physical meaning of internal state non-observable variables (e.g. stresses, heat flux, piezometric losses in fluids, ...) and include, explicitly, the physical laws between them and the directly observable external stimuli, and the boundary conditions by constraining the ANN, thus helping the network to "know" that the system should fulfil such conservation laws. The unknown constitutive model is then obtained as an implicit relation between such internal state variables and the input ones. This is the case in solid mechanics when relating the applied forces to the stresses, leaving free the material constitutive model that is implicitly established between stresses and displacements:

$$\begin{aligned} &\text{GET} \quad \boldsymbol{y} = \mathsf{Y}(\boldsymbol{x}); \quad \boldsymbol{v} = \mathsf{H}(\boldsymbol{u}), \\ &\text{s.t.} \quad \boldsymbol{R}(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{u}, \boldsymbol{v}) = \boldsymbol{0}, \end{aligned} \tag{5.21}$$

where $\boldsymbol{R}$ is a function encoding the universal principles driving the problem physics and the extra knowledge incorporated into the system. The network learning capability is now located at the two ANN relations $\boldsymbol{v} = \mathsf{H}(\mathbf{x})$ (explanatory network) and $\boldsymbol{y} = \mathsf{Y}(\boldsymbol{v})$ (predictive network).

(e) **Predictive problem with internal derived and hidden variables and physical constraints**. We combine the two previous problems, adding both PILs, thus helping the ANN to "know" that the system should fulfil the conservation laws and that the unknown constitutive model is composed of implicit relations between the internal hidden variables and the new observable ones. This is the case in solid mechanics when relating the applied forces with the stresses, leaving free the material constitutive model that is implicitly established between stresses and strains. The mathematical formulation is, therefore:

$$\begin{aligned} &\text{GET} \quad \boldsymbol{y} = Y(\boldsymbol{x}); \quad \boldsymbol{v} = \mathsf{H}(\boldsymbol{\varepsilon}), \\ &\text{s.t.} \quad \boldsymbol{\varepsilon} = \boldsymbol{E}(\boldsymbol{u}); \quad \boldsymbol{R}(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{\varepsilon}, \boldsymbol{v}) = \boldsymbol{0}, \end{aligned} \tag{5.22}$$

where, again, the relationship $\boldsymbol{E}(\boldsymbol{x}) = \boldsymbol{\varepsilon}$ may be imposed in an implicit or explicit form. Once the model $\boldsymbol{H}$ is learned, it can be exported as an input-output relationship and plugged into another PGNNIV, reducing the learning needs of the network, thus simplifying its architecture. The model $\boldsymbol{H}$ may even be exported to a commercial simulation software, as an $\boldsymbol{\varepsilon} \rightarrow \boldsymbol{v}$ function[15]. Moreover, the exportability allows an *a posteriori* analysis and evaluation of the model (symmetries, complexity, sensitivity analysis) and the possibility of generating *in silico* experiments for further fitting procedures (classical parametric models, response surfaces...) for a given set of model parameters $\boldsymbol{\lambda}$, $\boldsymbol{v} = \boldsymbol{H}(\boldsymbol{\varepsilon}; \boldsymbol{\lambda})$.

(f) **Explanatory problem to identify fixed internal parameters**. We add to the previous problem additional information on the structure of the constitutive model via new constraints between both PILs, adding (or not) information on the values or ranges of the constitutive parameters. In this case, it is possible to ask the ANN to predict the particular values of the parameters for a certain predefined explicit constitutive model structure. In this family of problems, the model has to be postulated *a priori* (e.g linear, isotropic, ...) and also, if required, some of the known parameters, obtaining the rest as output and reaching some explanatory capacity. We consider that all training data correspond to the same material and, therefore, all have the same material parameters to be determined. Thus, we have the same mathematical formulation as in the previous case except for the fact that the network relationship $\boldsymbol{v} = \mathsf{H}(\boldsymbol{\varepsilon})$ is not general, but is postulated using material symmetries and properties. One possible way is to define:

$$\mathsf{H}(\boldsymbol{u}) = \boldsymbol{H}(\boldsymbol{u}; \boldsymbol{\lambda}), \tag{5.23}$$

where $\boldsymbol{\lambda}$ are parameters characterising the state equation/constitutive model that have to be learned (elastic moduli or microstructural geometrical indicators, dielectric permittivity,...). After the training process, we obtain estimated values for these physical parameters $\boldsymbol{\lambda}$ relative to the state equation. As a particular case, the model may be perfectly defined:

$$\mathsf{H}(\boldsymbol{u}) = \boldsymbol{H}(\boldsymbol{u}), \tag{5.24}$$

so the network has no explanatory capacity, only predictive. In that case, the methodology presented may be seen as a pure model order reduction (MOR) technique or an offline calculator for posterior real time evaluations.

A simplified version of this problem is obtained when the constraints are applied directly to the input-output layers, as in the pure predictive problem with input-output constraints:

$$\begin{aligned} &\text{GET} \quad \boldsymbol{y} = \mathsf{Y}(\boldsymbol{x}), \\ &\text{s.t.} \quad \boldsymbol{R}(\boldsymbol{y}, \boldsymbol{x}; \boldsymbol{\lambda}) = \boldsymbol{0}. \end{aligned} \tag{5.25}$$

[15] When using tangent based solvers, it is necessary, additionally, to derive the derivatives of the output variables with respect to the input variables to evaluate the tangent operator, but this is easily done using the automatic differentiation capabilities of any ANN platform, such as TensorFlow.

This is the case, for instance, in the elastic problem when imposing the Navier equations and corresponding boundary conditions and compatibility constraints. This approach is the one explored by Raissi et al. (2019) in the so-called PINNs, but always in the latter form, which limits it to using only measurable variables.

(g) **Explanatory problem to identify the constitutive model**. In this problem, the idea is to test a set of potential constitutive models as in the previous case, getting the optimal parameters for each of them and then identifying the most likely of them as such with the lowest loss function. The mathematical formulation is, therefore:

$$\begin{aligned} &\text{GET} \quad \boldsymbol{y} = \Upsilon(\boldsymbol{x}); \quad \boldsymbol{v} = \boldsymbol{H}(\boldsymbol{u}; \boldsymbol{\lambda}), \quad \boldsymbol{H}, \in \mathcal{C} \\ &\text{s.t.} \quad \boldsymbol{\varepsilon} = \boldsymbol{E}(\boldsymbol{u}); \quad \boldsymbol{R}(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{\varepsilon}, \boldsymbol{v}) = \boldsymbol{0}. \end{aligned} \tag{5.26}$$

Here $\mathcal{C}$ denotes a *catalogue* of models. The one finally selected will be the one showing the best performance in terms of the loss function $\mathcal{L}$. As commented, a specific case is when $\boldsymbol{H}$ is totally specified, that is, $\boldsymbol{\lambda}$ are not learned from the data:

$$\begin{aligned} &\text{GET} \quad \boldsymbol{y} = \Upsilon(\boldsymbol{x}); \quad \boldsymbol{v} = \boldsymbol{H}(\boldsymbol{u}), \quad \boldsymbol{H} \in \mathcal{C}, \\ &\text{s.t.} \quad \boldsymbol{\varepsilon} = E(\boldsymbol{u}); \quad \boldsymbol{R}(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{\varepsilon}, \boldsymbol{v}) = \boldsymbol{0}. \end{aligned} \tag{5.27}$$

2. **Characterisation problems**: The goal here is to characterise the parameters of a pre-established constitutive model or state equation for different macroscopic materials. The material constitutive model or parameters are assumed to vary from one training data to another. Therefore, for the problem to make sense, enough physical information about the material response must be provided. The inputs are, consequently, the stimulus and the response of the material:

$$\boldsymbol{x} = (\boldsymbol{u}, \boldsymbol{f}, \boldsymbol{g}). \tag{5.28}$$

The output is constituted of variables related to the state model, that we will denote as $\boldsymbol{\theta}$:

$$\boldsymbol{y} = \boldsymbol{\theta}. \tag{5.29}$$

Two classes of descriptors may be provided as output variables.

- Any functional descriptor of the model characteristics (e.g. spatial homogeneity or time invariance, anisotropy or symmetries, linearity, memoryless, damage accumulation, ...). In general, this can be addressed, from a theoretical point of view, by using Lie Group theory and Noether's theorem. As an illustrative example, the spatial covariance of the elasticity tensor is an estimator of the material heterogeneity, and the autocorrelation function is an estimator of time invariance. In that sense $\boldsymbol{\theta} = \boldsymbol{\Theta}(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{f}, \boldsymbol{g})$, where $\boldsymbol{\Theta}$ is a prescribed function, obtained from the discretisation of a given functional.
- Prescribed model parameters. For instance, the Young modulus E and the Poisson ratio ν for the linear elastic problem, the diffusion coefficients (eigenvalues), D_I, D_{II}, D_{III} for three-dimensional problems, the particle

attraction a and the mole size exclusion parameter b for the Van Der Waals equation, or the medium permittivity and permeability coefficients $\boldsymbol{\varepsilon}$ and $\boldsymbol{\mu}$, among others. Following our previous notations, we state $\boldsymbol{\theta} = \boldsymbol{\lambda}$.

Therefore:

$$\begin{aligned} \boldsymbol{I}(\boldsymbol{u}, \boldsymbol{f}, \boldsymbol{g}) &= (\boldsymbol{u}, \boldsymbol{f}, \boldsymbol{g}), \\ \boldsymbol{O}(\boldsymbol{u}, \boldsymbol{f}, \boldsymbol{g}) &= \boldsymbol{\theta}. \end{aligned} \tag{5.30}$$

As in the previous case, many declinations are possible:

(a) **Pure characterisation problem**. The idea is now to establish a direct correspondence between the material stimulus and the response and model parameters (e.g. prediction of the elastic parameters from the external forces and displacements) without any constraint or explicit establishment of PILs. In this case, no constraint is imposed, recovering the black-box framework. Thus, the problem is formulated as:

$$\text{GET} \quad \boldsymbol{y} = \mathsf{Y}(\boldsymbol{x}). \tag{5.31}$$

(b) **Pure characterisation problem with input-output constraints**. As in the prediction problem, we assume some input-output relations, that are imposed via external constraints, without using the concept of internal state variables or PILs:

$$\begin{aligned} \text{GET} \quad & \boldsymbol{y} = \mathsf{Y}(\boldsymbol{x}) \\ \text{s.t.} \quad & \boldsymbol{R}(\boldsymbol{y}, \boldsymbol{x}) = \boldsymbol{0}. \end{aligned} \tag{5.32}$$

where $\boldsymbol{R}$ is a prescribed function that relates the input and output layers. For example, for a beam under axial force, at each element, $E - \frac{f_{i+1} - f_i}{u_{i+1} - u_i} \frac{L}{S} = 0$, $i = 1, \ldots, n$, with E (Young modulus) a parameter that needs to be characterised; f_i and u_i are the input variables (nodal forces and displacements) and L and S are known parameters (length and cross-section).

(c) **Pure predictive problem with internal derived variables**. We use again a particular PIL associated with the rates and gradients of the essential variables (e.g. strains, velocities, temperature change rate) $\boldsymbol{\varepsilon} = \boldsymbol{E}(\boldsymbol{u})$. The formulation of the problem is analogous to the one presented in the prediction problem, but adapted to the characterisation one:

$$\begin{aligned} \text{GET} \quad & \boldsymbol{y} = \mathsf{Y}(\boldsymbol{\varepsilon}, \boldsymbol{x}), \\ \text{s.t.} \quad & \boldsymbol{\varepsilon} = \boldsymbol{E}(\boldsymbol{u}). \end{aligned} \tag{5.33}$$

(d) **Characterisation problem with internal derived and hidden variables and physical constraints**. We combine the two previous cases, adding both PILs, thus helping the ANN to "know" that the system should fulfil the conservation laws and that the unknown constitutive model is composed of implicit relations between the internal hidden variables and the new observable ones

(e.g. relations between stresses and strains). The mathematical formulation is, therefore:

$$\begin{aligned} &\text{GET} \quad \boldsymbol{y} = \mathsf{Y}(\boldsymbol{x}); \quad \boldsymbol{v} = \mathsf{H}(\boldsymbol{u}) \\ &\text{s.t.} \quad \boldsymbol{\varepsilon} = \boldsymbol{E}(\boldsymbol{x}); \quad \boldsymbol{R}(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{\varepsilon}, \boldsymbol{v}) = \boldsymbol{0}, \end{aligned} \tag{5.34}$$

being Y and H the predictive and explanatory network, respectively. H tries to explain how the non-measurable variables are related to the measurable ones. This relation is physically interpretable thanks to the function $\boldsymbol{R}$, which establishes the physical constraints of the problem. As before, $\boldsymbol{E}$ is a known function defining the derived internal variables.

All these types of problems may be used both for non-transient problems or for transient ones, as described in Chapter 2, and following step by step a continuation approach. Of course, if some of the internal state variables depend on the history, this approach is always required even if the problem is time-independent (e.g. rate-independent plasticity) using a pseudo-time during the loading-unloading process.

To summarise, we recall that:

- If no constraint is applied and no PIL is defined, we recover the classical ANN framework.
- If the constraints are applied to the input or output layers we recover the formulation developed by other authors (Karpatne et al., 2017a; Raissi et al., 2019; Yadav et al., 2015). The prediction problem with the identification of some model parameters, in its simplified version, was also addressed in Raissi et al. (2019) for partial differential equations, without using the PIL concept.
- If the full information about the system is provided then we are bypassing the problem physics using a machine learning metamodel, also called a surrogate model (Cozad et al., 2014).

The rest of the cases that include PILs are original, up to the authors' knowledge.

5.2.3 Link to other methods

In recent years, many PIDS methods have been applied to solve problems where some part of the physics is known and the other has to be discovered. PGNNIV may be compared to the different classes of methods existing in the literature. A systematic revision of these methods is presented in Chapter 2. Here, we present only the seminal or important works and their relation with PGNNIV.

F. Chinesta and co-workers use Manifold Learning to establish, for computational mechanics (CM), the internal state equation $\boldsymbol{\varepsilon} \leftrightarrow \boldsymbol{\sigma}$ (Ibanez et al., 2018; Lopez et al., 2018). In their approach, the constitutive relationship is computed using ML techniques in the space $(\boldsymbol{\varepsilon}, \boldsymbol{\sigma})$ (note that $\boldsymbol{\sigma}$ is a non-measurable variable), represented as a low-dimensional manifold. This manifold is then used, instead of the constitutive equation, in the problem resolution. Our presented approach is similar in the sense that the model network $\boldsymbol{H}$ may be formulated using the Manifold Learning framework.

Indeed, methods such as kernel principal component analysis (kPCA), nonlinear kernel principal component analysis (NLPCA), locally linear embedding (LLE) and t-distributed stochastic Neighbour embedding (t-SNE) may be formulated in terms of appropriate weights, biases, activation functions and network connectivity.

A very recent idea uses the general equation for the non-equilibrium reversible-irreversible coupling (GENERIC) formalism (Grmela, 2010; Öttinger and Grmela, 1997) in time-dependent problems for model identification and evolution prediction (González et al., 2019; González et al., 2019). This may be seen as a particular PGNNIV, where $\boldsymbol{H}$ is defined using the Poisson and Dissipation operators, $\boldsymbol{L}$ and $\boldsymbol{D}$, together with the discrete version of other differential operators, if necessary. Constraints on many variables may be established, in order to ensure universal physics (the first and second laws of thermodynamics), by means of the degeneracy conditions. Combining this approach with the previous one leads to accurate solutions even while maintaining a reduced computational cost (Moya et al., 2019).

Two other approaches have been proposed with the same model-free philosophy to define the constitutive manifold using interpolation instead of regression. The first one is called the what you prescribe is what you get (WYPIWYG) strategy (Amores et al., 2019; Latorre and Montáns, 2014; Sussman and Bathe, 2009) and is based on spline interpolation. The second one is based on nearest-neighbour interpolation, which is totally model-free (Kirchdoerfer and Ortiz, 2016b). Both strategies have demonstrated good performance provided that we have the variables sampled at the space $(\boldsymbol{\varepsilon}, \boldsymbol{\sigma})$. They may be implemented using the PGNNIV framework by using the explicit spline equations or using radial basis function (RBF) for the closeness evaluation in the nearest-neighbour interpolation. However, these two approaches suffer from extrapolating capacity if the data-set provided has not a broad enough coverage, which is faced in the PGNNIV framework by making flexible the network associated with $\boldsymbol{H}$.

At last, and as mentioned before, when $\boldsymbol{H}$ is defined via a parametrisation of a classical model, $\boldsymbol{H}(\cdot) = \boldsymbol{H}(\cdot, \boldsymbol{\lambda})$, we recover the classical fitting framework (but using ANN tools and algorithms). If, in addition, $\boldsymbol{\lambda}$ is completely specified and the number of weights and biases is less than or equal to the number of parameters $\boldsymbol{\lambda}$, PGNNIV performs merely as a dimensionality reduction.

In a certain sense, the PGNNIV framework may be seen as a generalisation of all the former approaches. However, only the proposed approach is able to deal with non-measurable variables, albeit performing the data discovering in the state space, where both measurable and non-measurable variables are present. This is possible thanks to the network constraints $\boldsymbol{R}$, from which the state space is built and the internal model $\boldsymbol{H}$, which is unknown, is learned.

5.2.4 Formulation in continuum Physics

All the discussed in the previous sections has been formulated using the discrete language of ANNs. Even so, it has been remarked that the presented computational framework is able to cope with continuum problems, provided that they have been previously discretised in space and time by means of one of the many discretisation techniques available (Larsson and Thomee, 2009), such as finite differences (FD)

(Langtangen, 1999), finite elements (FE) (Zienkiewicz et al., 1977), or other spectral techniques (Boyd, 2001). Hence, the functional Eqs. (2.6) (usually a system of PDEs) defined in Chapter 2 may be expressed as Eqs. (5.1) after a discretisation. Nevertheless, there are important issues that deserve to be discussed in detail when the variables $\boldsymbol{u}$, $\boldsymbol{v}$, $\boldsymbol{f}$ and $\boldsymbol{g}$ and the operators $\boldsymbol{F}$, $\boldsymbol{G}$ and $\boldsymbol{H}$ have an intrinsic discretised value.

For instance, let us analyse the solid mechanics problem. We recast it using the notations and conventions of Eqs. (2.6) of Chapter 2:

$$\mathcal{F}(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{f}) = \boldsymbol{0}, \tag{5.35a}$$

$$\mathcal{G}(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{g}) = \boldsymbol{0}, \tag{5.35b}$$

$$\mathcal{H}(\boldsymbol{u}, \boldsymbol{v}) = \boldsymbol{0}. \tag{5.35c}$$

The functional $\mathcal{F}$ encodes mass, momentum and energy conservation equations at the domain and $\mathcal{G}$ at the boundaries, while $\mathcal{H}$ expresses the material-dependent constitutive relations written in terms of stresses and strains (or displacements). In this context, $\boldsymbol{u}$ is the displacement field and $\boldsymbol{v}$ represents the stresses and any other internal variable associated with the particular constitutive framework (plastic strains, hardening parameters, stress rates...). To fix ideas and lighten the volume of variables and equations, in the case of infinitesimal linear elasticity theory, we have:

$$\boldsymbol{\nabla} \cdot \boldsymbol{\sigma} = \rho \boldsymbol{b}, \tag{5.36a}$$

$$\boldsymbol{\varepsilon} = \frac{1}{2} \left(\boldsymbol{\nabla} \otimes \boldsymbol{U} + \boldsymbol{U} \otimes \boldsymbol{\nabla} \right), \tag{5.36b}$$

$$\boldsymbol{\sigma} = \boldsymbol{C} : \boldsymbol{\varepsilon}, \tag{5.36c}$$

with boundary conditions:

$$\boldsymbol{U} = \bar{\boldsymbol{U}}, \quad \text{in} \quad \Gamma_D, \tag{5.37a}$$

$$\boldsymbol{\sigma} \cdot \boldsymbol{n} = \bar{\boldsymbol{t}}, \quad \text{in} \quad \Gamma_N. \tag{5.37b}$$

Eqs. (5.36a) and (5.36b) are the equilibrium and kinematic relations respectively (the former is derived from universal laws: the variation of linear and angular *momenta*, while the latter is a mere definition of the strains in terms of the displacements). On the contrary, Eq. (5.36c) is the constitutive relation, which is postulated as a linear relationship between stress and strain. For this particular problem, $\boldsymbol{U}$ is the essential measurable field and $\boldsymbol{\sigma}$ is an internal field, represented by $\boldsymbol{v}$ in Eq. (5.35). Finally, $\boldsymbol{b}$ corresponds to the stimulus $\boldsymbol{f}$, and $\bar{\boldsymbol{U}}$ and $\bar{\boldsymbol{t}}$ are the known values of the respective Dirichlet and Neumann prescribed boundary conditions identified with $\boldsymbol{g}$ in Eq. (5.35).

Therefore, the functions $\mathcal{F}$, $\mathcal{G}$ and $\mathcal{H}$ are defined, following the recipe presented in Chapter 2 as:

- Involved variables: $\boldsymbol{u} = \boldsymbol{U}$, $\boldsymbol{v} = \boldsymbol{\sigma}$, $\boldsymbol{f} = \rho \boldsymbol{b}$ and $\boldsymbol{g} = (\bar{\boldsymbol{U}}, \bar{\boldsymbol{t}})$.

- Functional $\mathcal{F}$: Includes equilibrium and kinematics expressions:

$$\mathcal{F}(\boldsymbol{\varepsilon}, \boldsymbol{v}, \boldsymbol{f}) = \begin{pmatrix} \boldsymbol{\nabla} \cdot \boldsymbol{\sigma} - \rho \boldsymbol{b} \\ \boldsymbol{\varepsilon} - \frac{1}{2} \left(\boldsymbol{\nabla} \otimes \boldsymbol{u} + \boldsymbol{u} \otimes \boldsymbol{\nabla} \right) \end{pmatrix}. \tag{5.38}$$

- Functional $\mathcal{G}$: Includes boundary conditions:

$$\mathcal{G}(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{g}) = \begin{pmatrix} (\boldsymbol{u} - \bar{\boldsymbol{u}}) \, \mathbb{I}_{\Gamma_D} \\ (\boldsymbol{\sigma} \cdot \boldsymbol{n} - \bar{\boldsymbol{t}}) \, \mathbb{I}_{\Gamma_N} \end{pmatrix}, \tag{5.39}$$

  where $\mathbb{I}_A$ is the indicator function of the set A.

- Functional $\mathcal{H}$: Includes the constitutive equation:

$$\mathcal{H}(\boldsymbol{u}, \boldsymbol{v}) = \boldsymbol{\sigma} - \boldsymbol{C} : \boldsymbol{\varepsilon}. \tag{5.40}$$

We have used the auxiliar variable $\boldsymbol{\varepsilon} = \mathcal{E}(\boldsymbol{u})$ with

$$\mathcal{E}(\boldsymbol{u}) = \frac{1}{2} \left(\boldsymbol{\nabla} \otimes \boldsymbol{u} + \boldsymbol{u} \otimes \boldsymbol{\nabla} \right), \tag{5.41}$$

as explained in Section 5.2.2, for making clear the different physical relations and getting closer to the conventional approach, although we could have directly expressed $\mathcal{F}$ and $\mathcal{H}$ as:

$$\mathcal{F}(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{f}) = \boldsymbol{\nabla} \cdot \boldsymbol{\sigma} - \rho \boldsymbol{b}, \tag{5.42}$$

and

$$\mathcal{H}(\boldsymbol{u}, \boldsymbol{v}) = \boldsymbol{\sigma} - \boldsymbol{C} : \mathcal{E}(\boldsymbol{u}). \tag{5.43}$$

This example may be enriched when considering finite deformations, nonlinear/inelastic materials (Bonet et al., 2016), or more complex boundary conditions, keeping the structure of Eq. (5.35a) and boundary constraints given by (5.35b). For this particular problem, $\boldsymbol{U}$ is the essential measurable field, $\boldsymbol{\sigma}$ is an internal field, represented by $\boldsymbol{v}$ in Eq. (5.35) and $\boldsymbol{\varepsilon}$ is a derived variable to make explicit the pointwise state equation $\boldsymbol{\varepsilon} \to \boldsymbol{\sigma}$. Finally, $\rho \boldsymbol{b}$ corresponds to the stimulus $\boldsymbol{f}$, and $\bar{\boldsymbol{U}}$ and $\bar{\boldsymbol{t}}$ are the known values of the respective Dirichlet and Neumann prescribed boundary conditions identified with $\boldsymbol{g}$ in Eq. (5.35).

To solve numerically the physical problem (5.36) with boundary conditions (5.37), we first discretise it in space by means of one of the presented discretisation techniques. Once discretised, the problem may be written in the discrete form given by Eqs. (5.1). Now $\boldsymbol{u}$, $\boldsymbol{v}$ are unknown vectors of dimension n (number of degrees of freedom of the problem) containing all nodal values, $\boldsymbol{f}$ and $\boldsymbol{g}$ are known vectors of dimension n and n' (number of prescribed degrees of freedom at the boundary), respectively, and $\boldsymbol{F}$, $\boldsymbol{G}$ and $\boldsymbol{H}$ are corresponding array-valued functions. Note that in this discretised version of the problem, the continuous position label $\boldsymbol{x}$ is replaced by a discrete index i. Similarly, the fields $\boldsymbol{u}$, $\boldsymbol{v}$, $\boldsymbol{f}$ and $\boldsymbol{g}$ are replaced by their discrete counterparts, $\boldsymbol{u} = (u_1, \ldots, u_n)$, $\boldsymbol{v} = (u_1, \ldots, u_n)$, $\boldsymbol{f} = (f_1, \ldots, f_n)$ and $\boldsymbol{g} = (g_1, \ldots, g_{n'})$.

Depending on the selection of the input and output variables and the specific conditions of the problem, the different fields involved are either constant or variable fields. For instance, for the linear elastic problem presented we have at least two possibilities:

- The boundary conditions are fixed and we want to learn the displacement field $\boldsymbol{U}$ from the external forces $\rho\boldsymbol{b}$. In that case, $\bar{\boldsymbol{U}}$, $\boldsymbol{n}$, $\bar{\boldsymbol{t}}$ and $\boldsymbol{C}$ are data-independent tensor fields whereas $\boldsymbol{\varepsilon}$, $\boldsymbol{\sigma}$ and $\boldsymbol{U}$ are data-dependent fields.
- The boundary conditions (displacements and/or normal tractions at the boundaries) are taken as input and we want to learn the displacement field $\boldsymbol{u}$ when the external forces are known and fixed. In that case, $\rho\boldsymbol{b}$, $\boldsymbol{n}$ and $\boldsymbol{C}$ are data-independent and $\bar{\boldsymbol{U}}$, $\bar{\boldsymbol{t}}$, $\boldsymbol{\varepsilon}$, $\boldsymbol{\sigma}$ and $\boldsymbol{U}$ are data-dependent fields.

Using the particular expression of the functional $\mathcal{H}$ and of its discretised version $\boldsymbol{H}$, Eq. (5.35c) may be written as:

$$\boldsymbol{v} = \boldsymbol{H}(\boldsymbol{\varepsilon}), \tag{5.44}$$

that is much more adequate for our purposes, which are to **predict** new values of $\boldsymbol{u}$ but also to **explain** the nature of the constitutive relation $\boldsymbol{\varepsilon} \to \boldsymbol{v}$ (or $\boldsymbol{u} \to \boldsymbol{v}$ for more general conditions).

The next two sections describe an adequate computational representation for variables ($\boldsymbol{u}$, $\boldsymbol{v}$, $\boldsymbol{f}$ and $\boldsymbol{g}$) and operators ($\boldsymbol{F}$, $\boldsymbol{G}$ and $\boldsymbol{H}$) using any ANN platform, such as TensorFlow.

5.2.4.1 Variables description

In continuum physical problems, a time-dependent tensor field is a point-dependent magnitude $\boldsymbol{T} = \boldsymbol{T}(\boldsymbol{x}, t)$ indexed by the point coordinate $\boldsymbol{x}$ and time t. In many problems in continuum Physics, we deal with tensor fields that, once discretised, are represented by arrays of appropriate dimensions. For example, the time-dependent (l-covariant, m-contravariant) tensor field $\boldsymbol{T}$ is represented by the multi-indexed array $\tilde{\boldsymbol{T}}$, where $T^{i_1,\ldots,i_l}_{j_1,\ldots,j_m}[k_1, k_2, k_3, k_4] = T^{i_1,\ldots,i_l}_{j_1,\ldots,j_m}(x_{k_1}, y_{k_2}, z_{k_3}, t_{k_4})$.

The TensorFlow framework (Abadi et al., 2016) is particularly suitable for working with data associated with a physical discretised field. Indeed, a tensor field $\boldsymbol{T}$ is represented in TensorFlow notation by a multiarray tensor `T`. When that tensor field varies among samples of a given data-set, the tensor rank is expanded to take this into consideration. For instance, the value of a two dimensional discretised displacement field at a given time $t_k = k\Delta t$, $\boldsymbol{u}(\boldsymbol{x}, t_k)$, for a given sampled value i, is represented by $\mathtt{u}[i; \cdot, \cdot; k]$, that is a 2nd-rank tensor, where the symbol "$\cdot$" represents the two spatial indexes as a whole.

Field description. A continuum tensorial variable $\boldsymbol{T}$ is represented by a (TensorFlow) tensor (or multiarray) structure `T` of rank $1 + l + m + d + 1$ where l is the covariant index, m the contravariant index and d the dimension of the spatial manifold:

$$\mathtt{T}[\cdot, i_1, \ldots, i_l, j_1, \ldots, j_m; k_1, \ldots, k_d, l] = T^{i_1,\ldots,i_l}_{j_1,\ldots,j_m}(\boldsymbol{x}_{\mathsf{K}}, t_{\mathsf{l}}),$$

where K denotes a voxel indexation and l a time frame.

5.2.4.2 Operators description

The operator $\mathcal{F}$, on a vector field $\boldsymbol{u}$, acts, after discretisation, as a vectorial function $\boldsymbol{F}$. This is directly reframed in the TensorFlow language by defining a function relating two multiarray tensors. Briefly, a functional relationship $\boldsymbol{v} = \mathcal{F}(\boldsymbol{u})$ is first discretised into a n-variables function $\boldsymbol{v} = \boldsymbol{F}(\boldsymbol{u})$, which in turn is expressed in TensorFlow as a tensor relationship $\mathtt{v} = F(\mathtt{u})$. Nonetheless, there are two fundamental observations that have to be mentioned:

1. First, most of the operators acting in the formulation of the continuum physics are either (i) functions converting the value of a field function $\boldsymbol{u}$ at a certain point to the value of a different function $\boldsymbol{v}$ at the same point $\boldsymbol{v}(\boldsymbol{x}) = \boldsymbol{F}(\boldsymbol{u}(\boldsymbol{x}))$; (ii) Integro-Differential functional operators and, therefore, linear, changing a whole function into another $\mathcal{F}(\boldsymbol{u})(\boldsymbol{x}) = \boldsymbol{v}(\boldsymbol{x})$. TensorFlow is a framework that seems to have been developed for that purpose, as both cases of operators may be seen as prescribed network structures.

   The first case, that is an operator such that in the discrete case operates as $\boldsymbol{v}_{\mathsf{K}} = \boldsymbol{F}(\boldsymbol{u}_{\mathsf{K}})$, may be expanded, using convolution filters in the TensorFlow framework, into a multilayer perceptron. For example, if $\boldsymbol{v}$ is a 2D vector field, then, $v^{i_1}[k_1, k_2]$ is a rank 3 tensor (represented by the multiarray $\mathtt{v}$). Suppose that it is related to another 2D vector field $\boldsymbol{u}$, $v^{i'_1}[k'_1, k'_2]$ by the function $\boldsymbol{F}$, (represented by the multiarray $\mathtt{u}$). Therefore the relationship is represented by:

$$\underbrace{\mathtt{v_0}}_{[2,n_x,n_y]} \underbrace{\longrightarrow}_{[2,m_1,n_x,n_y]} \underbrace{\mathtt{v_1}}_{[m_1,n_x,n_y]} \underbrace{\longrightarrow}_{[m_1,m_2,n_x,n_y]} \cdots \underbrace{\longrightarrow}_{[m_{k-1},2,n_x,n_y]} \underbrace{\mathtt{v}_r}_{[2,n_x,n_y]} \tag{5.45}$$

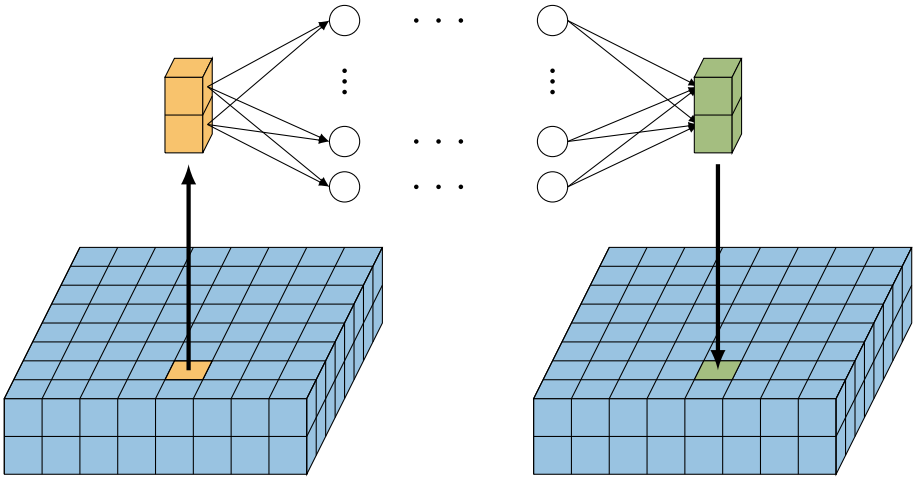
   where $\mathtt{v_0} = \mathtt{u}$ and $\mathtt{v_r} = \mathtt{v}$, being r the number of hidden layers and $m_1, \ldots, m_{r-1}$ the number of neurons at the corresponding hidden layer. The already mentioned universal approximation theorem guarantees that every sufficiently regular function $\boldsymbol{F}$ may be approximated by MLP so this approximation makes sense.

   The second case, that is, integro-differential linear operators, may be reframed in the TensorFlow framework using convolution filters of a given size. If $\boldsymbol{v} = \boldsymbol{F}(\boldsymbol{u})$ is a local linear operator relating a rank k tensor and a rank k' tensor, both representing 2D spatial fields:

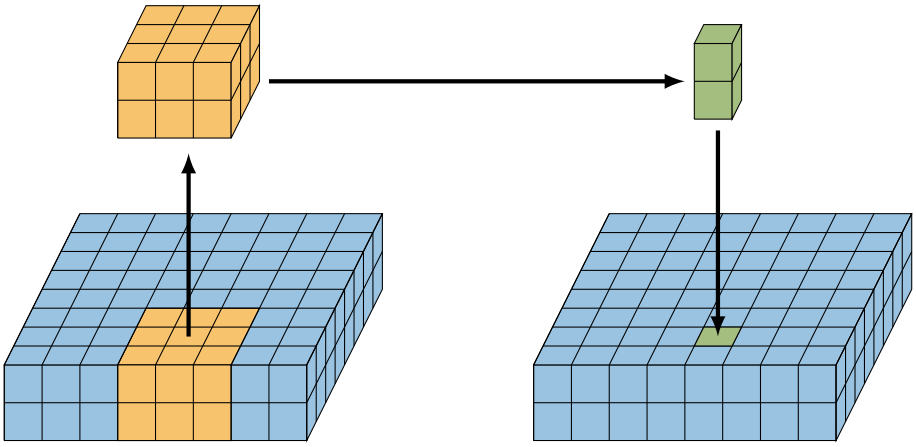
$$\underbrace{\overbrace{[2,\ldots,2}^{\mathtt{u}},n_x,n_y]}_{k} \underbrace{[2,\ldots,2}_{k}\overbrace{,2,\ldots,2}^{\longrightarrow},n_x,n_y]_{k'} \underbrace{\overbrace{[2,\ldots,2}^{\mathtt{v}},n'_x,n'_y]}_{k'} \tag{5.46}$$

   It is important to note that, if the considered operator is local or global, the spatial field $\boldsymbol{v}$ is undefined at some values close to the boundaries, so $n'_x \neq n_x$ and $n'_y \neq n_y$.

   Fig. 5.4 schematises the two presented families of operators of continuum physics: moving MLPs (nonlinear functions, Fig. 5.4a) and convolutional filters (linear functionals, Fig. 5.4b). Of course, other intermediate architectures are also possible. Hence, all common operators in physical problems may be framed in terms of ANN structures using MLP or CNN, including proper activation functions, in order to take into account possible nonlinearities.



(a) Nonlinear pointwise operator.



(b) Linear local operator.

Figure 5.4: Operator description based on MLP and CNN architecture. Local operators are moving filters with a prescribed window size and pointwise nonlinear are moving MLP. The procedure is illustrated for the relationship between two 2D vectorial fields.

2. Second, among the data-independent tensors, we distinguish between constant (non-trainable) and variable (trainable) tensors. Once we have fixed the physical problem and decided which is the input-output relation that has to be learned, the selection of the role of each operator tensor is natural: when a tensor is involved in a known operator, such as the ones related with $\boldsymbol{F}$ and $\boldsymbol{G}$ functions, it is a TensorFlow constant tensor and is denoted with a star. One particular example is the tensor associated with the derivation operator, D^*. If the tensor is associated with an unknown relationship, such as the predictive network Y or the explanatory network H, the tensor is a variable tensor. An example is the (possibly heterogeneous) elastic tensor $\boldsymbol{C}$, that, due to its intrinsic linear character, represents at once its nature (multiarray $\mathtt{C}$) and its action over the strain field (network C).

Functional description. A continuum tensorial operator $\mathcal{F}$ such that $\boldsymbol{v} = \mathcal{F}(\boldsymbol{u})$ is represented by a (TensorFlow) network architecture F:

$$\mathsf{F} : \mathcal{T}_u \to \mathcal{T}_v,$$

where $\mathcal{T}_u$ and $\mathcal{T}_v$ are the multiarray spaces associated with the fields $\boldsymbol{u}$ and $\boldsymbol{v}$, represented by the multiarrays $\mathtt{u}$ and $\mathtt{v}$ respectively. These operators may be constant or variable depending on the knowledge about the functional $\mathcal{F}$.

To illustrate the introduced concepts above, we particularise the general ideas above to a brief taxonomy of different operators found in continuum Physics problems. This discussion is not intended to be exhaustive and complete, but to show the suitability of the PGNNIV formulation to handle a very wide range of operators.

Common linear differential operators

With the presented framework, all differential operators can be casted as pre-defined filters acting on field tensors (Cai et al., 2012). A (discretised) differential operator is a function $\boldsymbol{D}$ transforming one multiarray tensor into another. Moreover, Dong et al. (2017) links the order of the differential operators to the order of sum rules. Differential operators may be encoded as known constant tensors $\mathtt{D}^*$. To fix ideas, let us consider the equilibrium equation in solid mechanics (infinitesimal theory):

$$\boldsymbol{\nabla} \cdot \boldsymbol{\sigma} = \rho \boldsymbol{b} \tag{5.47}$$

Eq. (5.47) involves the divergence operator, div, defined in this case for a two contravariant tensor. If $\boldsymbol{f} = \mathrm{div}(\boldsymbol{X})$, $f^i = X^{ij}_{|j}$, where $|$ represents the covariant derivative. In a coordinate representation, the covariant derivative is expressed for the considered tensor as $X^{ij}_{|j} = X^{ij}_{,j} + \Gamma^i_{jk} X^{kj} + \Gamma^j_{jk} X^{ik}$ where Γ^k_{ij} are the Christoffel symbols that, for the Levi-Civita connection, are defined in terms of the metric tensor $\mathfrak{g}$, and satisfy the following linear equation $\mathfrak{g}_{kl}\Gamma^k_{ij} = \frac{1}{2}\left(\mathfrak{g}_{jl,i} + \mathfrak{g}_{li,j} - \mathfrak{g}_{ij,l}\right)$. Now, as $\mathfrak{g}$ encodes the geometry of the problem, the only ingredient to reframe Eq. (5.47) to the multiarray framework is to select a discretisation of the common, one-dimensional, derivative operator ∂_i as a tensor operator. For instance, let us consider a two-dimensional problem. ∂_1, may be defined using first-order finite difference approximation as:

$$[\partial_1 F]^{\bullet}_{\bullet}[k,l] = \frac{1}{l} F^{\bullet}_{\bullet}[k+1,l] - \frac{1}{l} F^{\bullet}_{\bullet}[k,l], \tag{5.48}$$

so, as the operator is linear, the $\mathbb{R}^2 \to \mathbb{R}^2$ tensorial expression of ∂_1 at the 2D space is $[\partial_i][m,n,r,s] = -1/l$ if $r = m, s = n$, $[\partial_i][m,n,r,s] = 1/l$ if $r = m+1, s = n$ and $[\partial_i][m,n,r,s] = 0$ otherwise. Note that ∂_i is a convolutional filter, given in a planar 2D representation by the kernel stencil (moving from left to right and from bottom to top):

$$D_1 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & -1/l & 1/l \\ 0 & 0 & 0 \end{bmatrix}. \tag{5.49}$$

Analogously, the tensorial representation of ∂_2 is $[\partial_i]\,[m,n,r,s] = -1/l$ if $r = m, s = n$, $[\partial_i]\,[m,n,r,s] = 1/l$ if $r = m, s = n+1$ and $[\partial_i]\,[m,n,r,s] = 0$ otherwise, and its planar representation is:

$$D_2 = \begin{bmatrix} 0 & 1/l & 0 \\ 0 & -1/l & 0 \\ 0 & 0 & 0 \end{bmatrix}. \tag{5.50}$$

If we go back to the equilibrium equation, the tensor $\boldsymbol{\sigma}$ associated with the stress field is represented as $\sigma^{ij}[k,l]$, and the tensor $\boldsymbol{b}$ associated with the external forces field per unit mass is represented as $b^i[k,l]$, so the divergence operator may be expressed as a tensor $\mathtt{D}^*$ so that:

$$\rho b^i[r,s] = [D_j]^*[r,s,k,l]\sigma^{ij}[k,l]. \tag{5.51}$$

In particular, let us consider that there is no curvature, $\Gamma^k_{ij} = 0$. The non zero components of the tensor are $D^\bullet_1[1,1,1,1] = D^\bullet_2[1,1,1,1] = -1/l$ and $D^\bullet_1[1,1,1,2] = D^\bullet_2[1,1,1,2] = 1/l$ for a two-by-two grid.

Recall that all differential operators may be reframed as convolutional filters in the spatial slots. This has important consequences from a practical point of view:

- Tensor operators are sparse in the discretisation dimensions (that are those of greater dimensionality). This allows sparse-based algebra and storage, resulting in high performance computations and less demanding requirements.
- Differential operators may be easily built and used in standard ANN software codes and tools, such as TensorFlow, although some care must be taken in indexing.

In summary, all differential operators involved in the fundamental balance equations in continuum Physics may be encapsulated in this tensorial framework, provided we have established two main ingredients: the space geometry ($\mathfrak{g}$) and a given discretisation rule for differentiation.

State models

State models define the internal state (in general, non-measurable) variables of the problem in terms of the essential (measurable) ones. They can be written in a general case as:

$$\boldsymbol{v} = \mathcal{H}(\boldsymbol{u}), \tag{5.52}$$

where $\boldsymbol{v}$ is the set of internal variable fields and $\boldsymbol{u}$ is the set of essential variable fields and $\mathcal{H}$ must be interpreted as a functional (e.g. $\boldsymbol{\sigma} = \mathcal{H}(\boldsymbol{u}) = \frac{1}{2}\boldsymbol{C} : (\boldsymbol{\nabla} \otimes \boldsymbol{u} + \boldsymbol{u} \otimes \boldsymbol{\nabla})$ in linear elasticity, see Chapter 2 for other examples).

Once discretised, Eq. (5.52) is expressed as:

$$\boldsymbol{v} = \boldsymbol{H}(\boldsymbol{u}), \tag{5.53}$$

where now $\boldsymbol{v}$ and $\boldsymbol{u}$ are the tensor fields associated with the nodal field values and $\boldsymbol{H}$ is an (in general, nonlinear) mapping between tensors. Then, all symmetries and simplifications that may be assumed in the constitutive equation relating the two internal variable fields may be transcribed to the structure of the function $\boldsymbol{H}$:

- The linearity of the functional $\mathcal{H}$ is translated directly into the linearity of the function $\boldsymbol{H}$. Using the neural network language, this is equivalent to having no internal layers between neurons associated with the multiarrays $\mathtt{u}$ and $\mathtt{v}$, related to the fields considered. Different levels of complexity and non-linearity of $\mathcal{H}$ (and therefore of $\boldsymbol{H}$) may be handled with appropriate topologies of the ANN relating $\boldsymbol{v}$ and $\boldsymbol{u}$.
- The structure of the function $\boldsymbol{H}$ is further exploited using the ANN topology, involving different levels of sparsity (see Fig. 5.5) as described next:
  - Local constitutive laws imply that the operator $\mathcal{H}$ is local, that is, the value of $\boldsymbol{v}$ depends on the values of $\boldsymbol{u}(\boldsymbol{x})$ in a neighbourhood of $\boldsymbol{x}$. This means that $\mathcal{H}(\boldsymbol{u}) = \boldsymbol{H}(D^1[\boldsymbol{u}](\boldsymbol{x}), \dots, D^m[\boldsymbol{u}](\boldsymbol{x}))$, where D^k is a differential operator of order k. $m < \infty$ is called the order of the locality.

    When $m = 0$ we speak about order-zero local or pointwise constitutive laws. In that case $\boldsymbol{v}(\boldsymbol{x}) = \mathcal{H}(\boldsymbol{u}(\boldsymbol{x}))$. This entails block-diagonal tensorial operator structures, in the slots associated with the spatial discretisation: $\mathtt{v}[\cdot;\mathsf{I}] = \mathsf{H}(\mathtt{u}[\cdot;\mathsf{I}])$. In the deep learning (DL) framework, these kinds of relationships are associated with partitioned networks, as it is illustrated in Fig. 5.5a.

    When $m > 0$, the tensors are sparse but not necessarily block-diagonal. In the DL framework, these operators are associated with convolutional filters, as shown in Fig. 5.5b.
  - Non-local constitutive laws imply that $\mathcal{H}$ is not local, that is, the value of $\boldsymbol{v}$ depends on the values of $\boldsymbol{v}(\boldsymbol{x})$ on the whole spatial domain. In the language of differential operators, $\mathcal{H}(\boldsymbol{u}) = \boldsymbol{H}(D^1[\boldsymbol{u}](\boldsymbol{x}), \dots, D^m[\boldsymbol{u}](\boldsymbol{x}), \dots)$. There are many ways of defining non-local functionals (see, for instance, Ros-Oton (2016) and included references for a motivation and examples in elliptic operators). In that case, the tensor operators are dense and so it is the topology of the ANN associated, as illustrated in Fig. 5.5c.

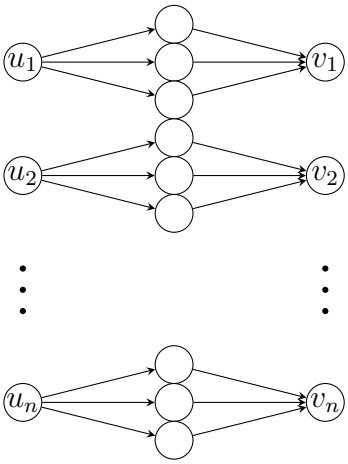
  These different situations may be modulated in several hierarchical levels in the network. For instance, in linear elasticity, $\boldsymbol{\sigma} = \frac{1}{2}\boldsymbol{C} : (\nabla \otimes \boldsymbol{u} + \boldsymbol{u} \otimes \boldsymbol{\nabla})$, so the material is local with respect to $\boldsymbol{u}$. Therefore, an accurate ANN for working with this model is obtained by combining the previous ideas, as shown in Fig. 5.5d.
- The difference between homogeneous and heterogeneous constitutive relations may be also handled. For local models (block-diagonal or sparse tensors), the different blocks or filters may be (or not) dependent on the spatial point considered. For the former, the indexes referring to the spatial part of the tensor are spurious and therefore may be omitted. For instance, in the linear elastic problem, in general, $\sigma^{ij}[\mathsf{I}] = C^{ijkl}[\mathsf{I},\mathsf{J}]\varepsilon_{kl}[\mathsf{J}]$ where I and J are voxel indexations. The fact that elasticity assumes a pointwise relationship $\boldsymbol{\sigma}(\boldsymbol{x}) = \boldsymbol{H}(\boldsymbol{\varepsilon}(\boldsymbol{x}))$ implies that $C^{ijkl}[\mathsf{I},\mathsf{J}] = C^{ijkl}[\mathsf{I}]\delta^{\mathsf{J}}_{\mathsf{I}}$, where $C^{ijkl}[\mathsf{I}]$ is the common elasticity tensor. For homogeneous materials, $\boldsymbol{C}[\mathsf{I}] = \boldsymbol{C}$.
- The tensor relations may be adapted for the exploitation of further symmetries in the constitutive equation. This includes:

  - Relations derived from the principle of objectivity, that is, reference frame independence.
  - Constraints related to the physical or geometrical foundations of the model (e.g. major and minor symmetries of the elastic tensor, associated with thermodynamics, angular momentum conservation, and compatibility constraints).
  - Additional constraints related to special symmetries of the constitutive model, that is, orthotropy, isotropy...

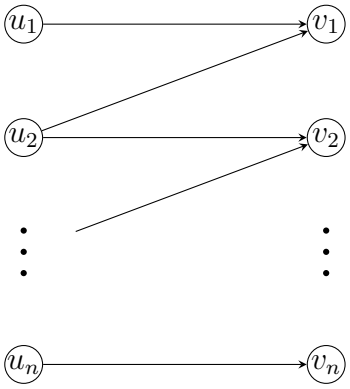
  All these symmetries may be enforced by adding constraints to the PGNNIV (implicit way) or by assuming a given topology for the ANN (explicit way). Indeed, if $\boldsymbol{v} = \boldsymbol{H}(\boldsymbol{u})$, the existence of a given symmetry is equivalent, in the Noether sense, to the action of a given group of transformations, so that $\boldsymbol{H}(\boldsymbol{u}) = \boldsymbol{H}(\boldsymbol{A}(\boldsymbol{u}))$, where $\boldsymbol{A} \in \mathcal{A}$ is a group of transformations. Therefore, we can look for a finite set of transformations $\boldsymbol{A}_k$ in a way such that $\boldsymbol{H}(\boldsymbol{u}) = \boldsymbol{H}(\boldsymbol{A}(\boldsymbol{u})), A \in \mathcal{A} \Leftrightarrow \boldsymbol{A}_k(\boldsymbol{H}(\boldsymbol{u})) = \boldsymbol{H}(\boldsymbol{u})$, or to *a priory* set up a topology for the deep neural network so that it becomes invariant under the action of all $\boldsymbol{A}$. This has been recently done in what has been coined as geometric deep learning (GDL) (Bronstein et al., 2017; Cao et al., 2020).

- Finally, the classical framework of parametric fitting is a very particular case, in which some of the internal layers are related to the others by means of a parametric explicit expression. In that case, $\boldsymbol{v} = \boldsymbol{H}(\boldsymbol{u}; \boldsymbol{\lambda})$ where the function $\boldsymbol{H}$ is explicitly imposed, and the functional relationship depends on the value of unknown parameters $\boldsymbol{\lambda}$, that are variable TensorFlow scalars obtained, in general, during the training process.

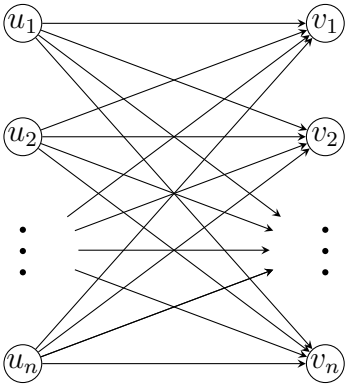
Sometimes it is useful to introduce other operators to enforce higher-order discretisations, or to adapt the problem to other numerical methods. This is the case of special filters for meshless approaches such as smoothed particle hydrodynamics (SPH) (Gingold and Monaghan, 1977), diffuse element method (DEM) (Nayroles et al., 1992) or natural element method (NEM) (Chinesta et al., 2013a; Sukumar et al., 1998) among many others. Also, it is easy to adapt this framework to integral formulations as in the finite element method (FEM). Indeed, FE integrals may be expressed in terms of the nodal values, being this relationship dependent on the shape function and the chosen numerical integrator, but otherwise fixed for a given degree of approximation. For instance, a moving averaging filter applied to the nodes recovers the framework of linear shape functions for a given element. Also, ANN methods have been adapted for dealing with pseudo-spectral (collocation) methods (Rudd and Ferrari, 2015). It is also possible to increase the order of the differential operators. This relies on the fact that a higher-order differential operator can be expressed as the subsequent application of lower-order ones, enriching the differentiation scheme. For instance, if Δ_h^+ is the forward difference operator ($\Delta_h^+ f^i = \frac{1}{h}(f^{i+1} - f^i)$), of order h, $\frac{1}{h}(\Delta_h^+ - \frac{1}{2}(\Delta_h^+)^2)$ is a forward difference operator of order h^2. Finally, another useful possibility are stabilisation filters in time-dependent problems to ensure the fulfilment of well-known stability criteria (Fischer and Mullen, 2001), or the filters that are designed for obtaining high fidelity time integrations, such as Crank-Nicolson integration or Runge-Kutta integrators (Raissi et al., 2019).



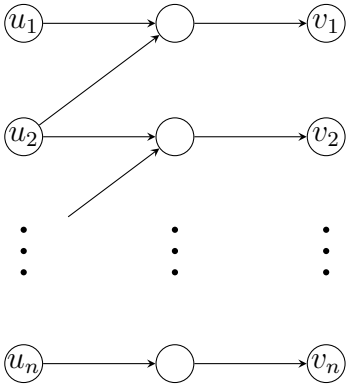
(a) Pointwise constitutive relation.



(b) Local constitutive relation.



(c) Global constitutive relation.



(d) Hierarchical combination.

Figure 5.5: Illustration of the different structures associated with constitutive equations. The different schemes are illustrative for one-dimensional problems. Note that when convolutional layers are applied (local operators), values at the boundaries may not be conveniently described.

5.2.4.3 Probes and quantities of interest

The last straightforward application of the TensorFlow framework to continuum problems is probes and QoI. Probes provide values of the continuum fields at certain prescribed points, and QoI are computed from the fields by a known functional $\mathcal{Q}$. The most common probe is the value of a tensor field evaluated at a point or a region. Other common probes are tractions on a plane, strains along a direction and fluxes over a certain surface at a certain time frame. These values may be expressed in terms of tensor quantities. For probes, given a point $\boldsymbol{x}$ and a time frame t, $P = \Pi\left(\boldsymbol{T}(\boldsymbol{x}, t)\right)$, where Π is a linear map and, in many cases, a projection operator. For instance, the normal stress to a plane defined by its outward normal vector $\boldsymbol{n}$ at a point $\boldsymbol{x}$ is $\sigma_n = (\boldsymbol{\sigma}(\boldsymbol{x}) \cdot \boldsymbol{n}) \cdot \boldsymbol{n}$, the pressure at a point $\boldsymbol{x}$ and a time t is $p = \mathrm{Tr}\left(\boldsymbol{\sigma}(\boldsymbol{x}, t)\right)$, and the y-displacement at a point $\boldsymbol{x}$ and time $\boldsymbol{t}$ is $u_y = \boldsymbol{u}(\boldsymbol{x}, t) \cdot \boldsymbol{e}_y$. The more general case of a QoI is analogous but considering functionals instead of linear maps, that is $Q = \mathcal{Q}(\boldsymbol{T})$. For instance, the mass flow over a surface $\mathcal{S}$ is $Q_m = \int_{\mathcal{S}} \rho \boldsymbol{v} \cdot \boldsymbol{n} dS$. Therefore, probes and QoIs are treated at the TensorFlow framework as networks of

variable complexity, Q. As, for that case, the operator is always known, it will be denoted by Q^*.

Probes and QoIs may be evaluated for prediction purposes, for the model interpretability, and also may be included in the loss function $\mathcal{L}$ when a measurement feedback is possible.

Probes and QoI description. A probe or a QoI, Q, is represented by a (TensorFlow) network architecture Q^*

$$\mathsf{Q}^* : \mathcal{T}_u \times \mathcal{T}_v \to \mathbb{R},$$

where $\mathcal{T}_u$ and $\mathcal{T}_v$ are the multiarray spaces associated with the fields $\boldsymbol{u}$ and $\boldsymbol{v}$, represented by the multiarrays `u` and `v` respectively.

5.3 Applications

We now apply the presented methodology to two SBES problems. The first one is a simple although very illustrative problem coming from hydraulics. We will dwell on many details and many declinations of the problem in order to illustrate the methodology, trying to use it as a conductor example. The second one is an illustration of how the method can be used for continuum problems formulated in terms of PDEs.

5.3.1 Discrete problem

We consider several examples to illustrate the methodology and the different types of applications and we present the two approaches described: prediction and characterisation.

5.3.1.1 Problem statement

Let us consider a pipe segment of length L_1, with a circular cross-section of diameter D_1, and a sudden change in its circular cross-section to another segment of length L_2, with the same shape but with a bigger diameter D_2 (Fig. 5.6). The initial objective is to compute the head pressure loss, Δp, along the length of the whole pipe in the steady-state regime, assuming fluid incompressibility. Using the Bernoulli equation, along a streamline, the hydraulic head is defined as:

$$h = \frac{v^2}{2g} + z + \frac{p}{\gamma}, \tag{5.54}$$

with $\gamma = \rho g$, being ρ the density of the fluid, g the gravity acceleration, z the elevation, so that Δh is the hydraulic head loss due to eddy dissipation and wall friction that corresponds to the state equation of the problem, so it has to be characterised by means of:

- Additional assumptions, usually with poor accuracy.

- Experimental tests with the corresponding fitted phenomenological equations.
- Simulations with complex fluid flow models.

Here, we shall assume that the hydraulic head loss is associated with two physical phenomena:

1. Viscous dissipation distributed along the pipe.
2. Localised dissipation at the pipe expansion.

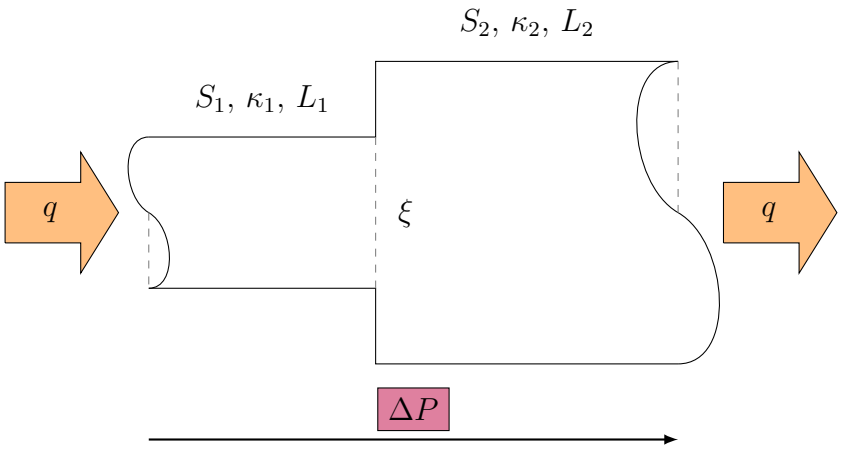


Figure 5.6: Scheme of the illustrative example. Hydraulic head loss along a piecewise cross-section pipe.

For the first physical phenomenon, we have distributed losses along a streamline, so assuming the flow as one-dimensional and considering a horizontal pipe, we can write:

$$\frac{d}{dx}\left(\frac{v^2}{2g}+\frac{p}{\gamma}\right)=\frac{dh}{dx}=-i, \tag{5.55}$$

with i the hydraulic head slope.

It is common to express the hydraulic head slope in terms of the fluid velocity by using the Darcy-Weisbach expression (Weisbach, 1845):

$$i=f_D\frac{1}{2g}\frac{v^2}{\Phi}, \tag{5.56}$$

where Φ is the hydraulic diameter of the pipe and f_D the Darcy friction factor, an empirical coefficient that, again, is determined by additional hypotheses or semi-empirical expressions.

Another common empirical model is the Hazen-Williams expression for the hydraulic head slope, that considers the fluid viscosity and the pipe roughness simultaneously (Williams and Hazen, 1908):

$$i=\lambda\left(\frac{q}{\kappa}\right)^{\alpha}\Phi^{\beta}, \tag{5.57}$$

with $q = Sv$ the flow rate (S is the cross-section area), $\lambda = 10.67$, $\alpha = 1.8520$, $\beta = -4.8704$, κ the roughness parameter of the pipe wall and Φ its hydraulic diameter.

If the pipe has the same properties along a certain distance L, the previous expression can be immediately integrated, getting:

$$\Delta h = \frac{dh}{dx} L = -iL. \tag{5.58}$$

If we now move to the second term of the energy loss, due to the sudden pipe expansion, a typical approach is the so-called Borda-Carnot equation (Batchelor, 2000):

$$\Delta h = \xi \frac{1}{2g} \left(1 - \frac{S_1}{S_2}\right)^2 v_1^2, \tag{5.59}$$

where S_1 and v_1 are the cross-section area and flow velocity before the expansion and S_2 is the cross-section area after expansion. ξ is, again, an empirical coefficient accounting for the magnitude of the viscous eddy dissipation[16].

Combining these two hydraulic head loss phenomena, we can finally write the expressions for the head pressure losses, depending on the empirical model used:

- **Darcy-Weisbach model**:

$$(\Delta p)_1 = \frac{1}{2} \rho f_{D_1} \frac{v_1^2}{\Phi_1} L_1, \tag{5.60a}$$

$$(\Delta p)_e = \frac{1}{2} \rho q^2 \left[\left(\frac{1}{S_2^2} - \frac{1}{S_1^2} \right) + \xi \left(\frac{1}{S_1} - \frac{1}{S_2} \right)^2 \right], \tag{5.60b}$$

$$(\Delta p)_2 = \frac{1}{2} \rho f_{D_2} \frac{v_2^2}{\Phi_2} L_2, \tag{5.60c}$$

  where we have to consider that f_{D_i} is the Darcy friction factor that may depend, in general, on the Reynolds number, Re, and on the pipe relative roughness ε/Φ, where ε is the pipe's effective roughness height. Some models have been formulated for this dependence, being Colebrook-White equation (Colebrook and White, 1937; Colebrook et al., 1939) the most common one.

- **Hazen-Williams model**:

$$(\Delta p)_1 = \lambda \gamma \left(\frac{q}{\kappa_1} \right)^\alpha \Phi_1^\beta L_1, \tag{5.61a}$$

$$(\Delta p)_e = \frac{1}{2} \rho q^2 \left[\left(\frac{1}{S_2^2} - \frac{1}{S_1^2} \right) + \xi \left(\frac{1}{S_1} - \frac{1}{S_2} \right)^2 \right], \tag{5.61b}$$

$$(\Delta p)_2 = \lambda \gamma \left(\frac{q}{\kappa_2} \right)^\alpha \Phi_2^\beta L_2, \tag{5.61c}$$

[16] Assuming an almost uniform flow velocity profile, which is the case for fully developed flows, it is possible to derive, from the mass and momentum conservation equations, that $\xi \simeq 1$ (Batchelor, 2000).

Eq. (5.60) or (5.61) describes the whole physics of the model when Darcy-Weisbach and Borda-Carnot loss models or Hazen-Williams and Borda-Carnot models are assumed, respectively. In addition to the empirical equation relating the energy losses with the velocity, there is another underlying universal physics inherent to the problem, although part of it has been already used even if it was masked: the momentum conservation equation has been used in the derivation of the Bernoulli equation while the energy conservation has been applied to get Eq (5.58). Another universal equation is mass conservation (constant flow equation), which was used only for obtaining Eq. (5.60b) or Eq. (5.61b) but will be used again later.

Let us now define the two different approaches, prediction or characterisation, to solve this simple problem.

- **Prediction problem:**

  The aim here is to predict a pressure drop Δp from a given flow q through the pipe. This problem is illustrative in the sense that we know a conserved quantity of the problem (the mass) and we want to know the functional dependence between another physical variable (pressure drop) and this conserved quantity (equivalent to velocity). We can write Eq. (5.60) or Eq. (5.61) as:

  $$(\Delta p)_i = f_i(q), \tag{5.62}$$

  where f_i is a function to be learned.

  This model may be seen as a physically-based mathematical relation, relating one input, $x = q$ to an output, $y = \Delta p$ variable, by a function $y = Y(x; \boldsymbol{\lambda})$, that includes many formal parameters $\boldsymbol{\lambda}$ obtained for each empirical model from the problem data:

  - The cross-section areas of the pipe: S_i, $i = 1, 2$.
  - The lengths of the different sections of the pipe: L_i, $i = 1, 2$.
  - Some physical parameters: density of the water ρ and gravitational acceleration g.

  In addition to the parameters related to the geometry and physical properties, for the Darcy-Weisbach model, where $f_{D_i} = f_{D_i}(\mathrm{Re}, \varepsilon/\Phi)$, we have to include:

  - The different hydraulic diameters: Φ_i, $i = 1, 2$.
  - The flow kinematic viscosity: ν.
  - The effective roughness height of each pipe: ε_i, $i = 1, 2$.

  Analogously, for the Hazen-Williams model, we have to include:

  - The different hydraulic diameters: Φ_i, $i = 1, 2$.
  - The roughness of the pipe wall: κ_i, $i = 1, 2$.

  Finally, for the Borda-Carnot model, we have to include the parameter ξ. Taking into account that for circular-based cylindrical pipes $\Phi_i = \sqrt{\frac{4S_i}{\pi}}$, we can forget about these parameters for all the models presented.

  Hence, depending on the selected approach:

1. We try to learn the relationship $(\Delta p)_i = \mathsf{Y}(q)$, $i = 1, 2, 3$.
2. We learn the pressure drop from the velocities, $(\Delta p)_i = \mathsf{Y}(v_0, v_1, v_2)$, where $v_i = \frac{q}{S_i}$ represents the velocity at the inlet ($i = 0$), before the transition ($i = 1$) and at the outlet ($i = 2$). Note that for this problem $v_1 = v_0$. This is a simple but illustrative example of defining a new state variable from the input variable q.
3. We learn $(\Delta p)_i$ from the flow q but by means of local pressure gradients, $(\Delta p)_i = w_i L_i$, where $w_i = \frac{dp}{dx}\big|_i$ is the local pressure drop gradient along the segment i, and $(\Delta p)_e$. Besides, we have to postulate the relation $(w_1, w_2, (\Delta p)_e) = \mathsf{H}(q)$.
4. A combination of the two previous ones. We try to learn $(\Delta p)_i$ from the flow q but by means of local pressure gradients as before. Besides, we postulate $(w_1, w_2, (\Delta p)_e) = \mathsf{H}(v_0, v_1, v_2)$, where we have defined a new set of internal variables, v_i, that must satisfy mass conservation equation, $v_i S_i = q$.
5. We try to learn $(\Delta p)_i$ from the flow q as in the previous example. The only difference is that, now, we define $(w_1, w_2, (\Delta p)_e) = \boldsymbol{H}(v_0, v_1, v_2; \boldsymbol{\lambda})$.
   - For the Darcy-Weisbach model we have to consider Eqs. (5.60). Assuming that the kinematic viscosity ν is known, the unknown parameters are $\lambda_1 = \xi$, $\lambda_2 = \varepsilon_1$ and $\lambda_3 = \varepsilon_2$. In the particular case of laminar regime, we have $f_{D_i}(v_i; \nu, \varepsilon_i) = 64\nu/(v_i \Phi_i)$ so it is possible to define $\lambda_2 = \nu/\Phi_1$ and $\lambda_3 = \nu/\Phi_2$ and ignore the effective roughness heights. We will proceed this way, since there is no loss of generality and we do not have to use the complex Colebrook-White dependencies.
   - Analogously, for the Hazen-Williams model we have to consider Eqs. (5.61). Now, we can establish as unknown parameters $\lambda_1 = \xi$, $\lambda_2 = \frac{\lambda \Phi_1^\beta}{\kappa_1^\alpha}$ and $\lambda_3 = \frac{\lambda \Phi_2^\beta}{\kappa_2^\alpha}$.

   The two presented alternatives postulating that $\mathsf{H}(v_0, v_1, v_2) = \boldsymbol{H}(v_0, v_1, v_2; \boldsymbol{\lambda})$ are quite common, as they represent the fact that we do not know the roughness of each pipe segment but we do know the model and the geometry. The rest of the parameters act as multiplicative constants.
6. The same problem as before, except for the fact that, now, we have several possible models and we want to select the best. For example, in our case, we have to select between the Darcy-Weisbach model and the Hazen-Williams' one.

- **Characterisation problem:**

  Now, the aim is to characterise some of the parameters of the pipe segments for a given set of values (q, p_0, p_1, p_2). That is, the parameters of the constitutive equation vary from one sample to another, and the final goal is to predict those parameters defining the intrinsic behaviour of the system, assuming a given state model structure. Let us suppose, for instance, that we want to characterise the roughness of the pipe in terms of the two parameters κ_1 and κ_2 using the Hazen-Williams model. For the sake of simplicity, let us fix a constant area $S_1 = S_2$, equivalent to assuming $\xi = 0$ and $\rho = 0$ at Eqs. (5.60) and (5.61). Note that this example is very illustrative in the sense that it characterises a

spatially variable property of a given material. This may be extrapolated to obtain the profile of a material parameter $\kappa = \kappa(\boldsymbol{x})$ for heterogeneous materials, when monitoring its behaviour under certain actions.

As we are in the heterogeneous case, for discovering the roughness parameters, it is necessary to measure the pressure drop at the two segments, otherwise, the problem would be undetermined. For the present problem, the relation to be learned is $(q, p_0, p_1, p_2) \rightarrow (\kappa_1, \kappa_2)$. Considering, for instance, the Hazen-Williams model, the parameters κ_1 and κ_2 are related to the above variables by:

$$\kappa_i = \left(\gamma \lambda \Phi_i^{\beta} \delta_i\right)^{1/\alpha} q \left(p_{i-1} - p_i\right)^{-1/\alpha}, \tag{5.63}$$

that is $\kappa_i = \lambda_{1,i} q \left(p_{i-1} - p_i\right)^{-1/\alpha}$, where $\lambda_{1,i} = \left(\gamma \lambda \Phi_i^{\beta} \delta_i\right)^{1/\alpha}$. Note that the parameter dependence is $\kappa_1 = Y(p_0, p_1)$ and $\kappa_2 = Y(p_2, p_1)$. This is not the general case, but could be exploited in the design of the state equation model $\boldsymbol{\kappa} = \mathsf{H}(p_0, p_1, p_2)$, and will be discussed later. Here, a conventional MLP, H, will be used to model the state equation.

In an analogous manner to the prediction problem, depending on the selected approach, we would like:

1. To learn the relationship $(\kappa_1, \kappa_2) = \mathsf{Y}(q, p_0, p_1, p_2)$.
2. To learn the variables κ_1 and κ_2 from the velocities and pressures, $(\kappa_1, \kappa_2) = \mathsf{Y}(v_1, v_2, p_0, p_1, p_2)$, with velocities satisfying the conservation equation, $q = S_i v_i$.
3. To learn κ_1 and κ_2 from the pressures and flow velocities, $(\kappa_1, \kappa_2) = \mathsf{Y}(v_1, v_2, w_1, w_2)$, where $(w_1, w_2) = \mathsf{H}(p_0, p_1, p_2)$, with w_1, w_2 the head pressure drops (related to viscous forces). $R(w_1, w_2, p_0, p_1, p_2) = (\delta_1 w_1 - (p_1 - p_0), \delta_2 w_2 - (p_2 - p_1)$ comes from the momentum conservation.

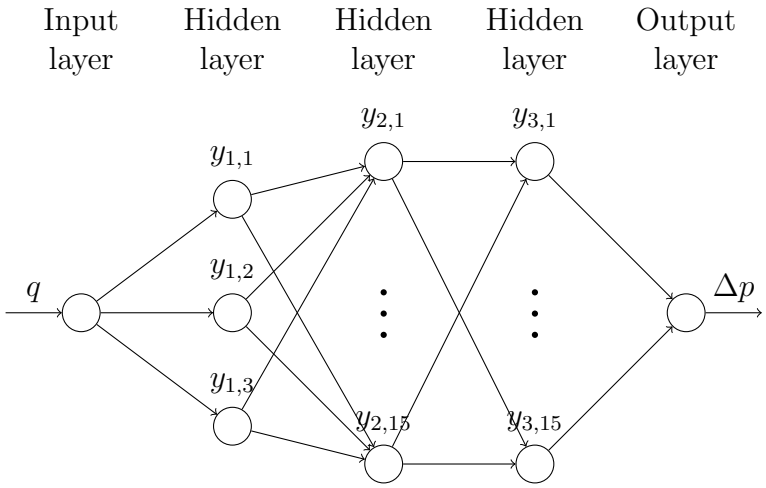
5.3.1.2 The prediction problem

Let us first consider the problem of predicting directly the nonlinear relationship $q \rightarrow \Delta p$ without any additional constraint, which is a standard ANN approach. An ANN is established to solve the single input - single output problem proposed. We choose an ANN with only three hidden layers of $n_1 = 3$, $n_2 = 15$ and $n_3 = 15$ neurons, respectively. The network is illustrated in Fig. 5.7a.

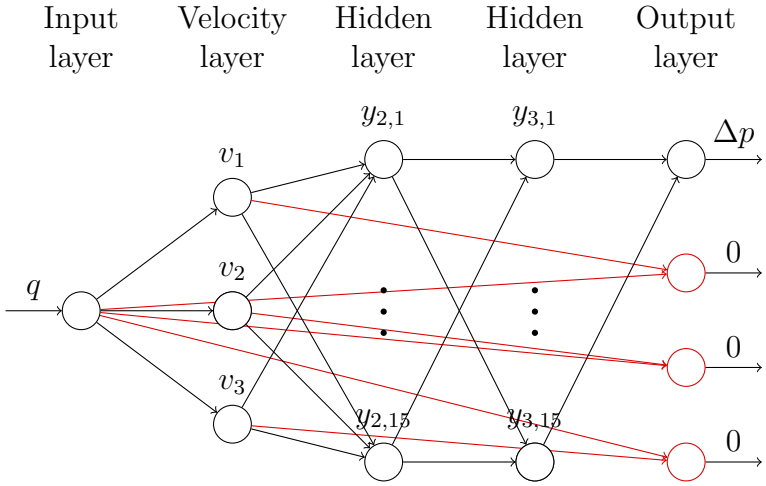
The neural layers are mathematically defined as follows. Given $\boldsymbol{x} = \boldsymbol{y}_0 = q$, $\boldsymbol{y} = \boldsymbol{y}_4 = \Delta p$ and:

$$\boldsymbol{y}_1 = \phi(\boldsymbol{x}\boldsymbol{W}_1 + \boldsymbol{b}_1), \tag{5.64a}$$
$$\boldsymbol{y}_2 = \phi(\boldsymbol{y}_1\boldsymbol{W}_2 + \boldsymbol{b}_2), \tag{5.64b}$$
$$\boldsymbol{y}_3 = \phi(\boldsymbol{y}_2\boldsymbol{W}_3 + \boldsymbol{b}_3), \tag{5.64c}$$
$$\boldsymbol{y}_4 = \boldsymbol{y}_3\boldsymbol{W}_4 + \boldsymbol{b}_4, \tag{5.64d}$$

with ϕ a ReLU function.



(a) Unconstrained ANN.



(b) Physically-Guided ANN.

Figure 5.7: Comparison of unconstrained and constrained neural networks. The constrained neural network (PGNNIV) is illustrated by its physically augmented network, where constraints have been replaced by extra outputs.

Now we define an analogous ANN in which we impose mass conservation, that is, constant flow in the three reference points of the pipe:

$$v_i Si = q, \quad i = 0, 1, 2. \tag{5.65}$$

The imposition of Eqs. (5.65) gives the hidden variables $\boldsymbol{y}_1$ a clear physical interpretation: the flow velocities v_i. Eq. (5.65) are imposed in the neural network system via constraints between the values of the corresponding neurons following a penalty approach. Fig. 5.7b illustrates the interpretation of this PGNNIV. The network topology is identical to the one of the unconstrained network, but the objective function includes now additional terms, accounting for the constraints associated with the physics.

Solving the prediction problem

For the training process, the data input was randomly generated with a uniform distribution, using the state model presented in Eq. (5.61) for $q \in [1.0; 5.0]\,(\mathrm{m}^3/\mathrm{s})$,

that is, the Hazen-Williams model. The physical parameters used for the data generation are shown in Table 5.1.

Parameter	S_1	S_1	ρ	ξ	g	κ_1	κ_2	L_1	L_2
Value	$1.0\,\mathrm{m}^2$	$2.0\,\mathrm{m}^2$	$1.0\,\mathrm{kg/m}^3$	1.0	$9.81\,\mathrm{m/s}^2$	140	140	10 m	10 m

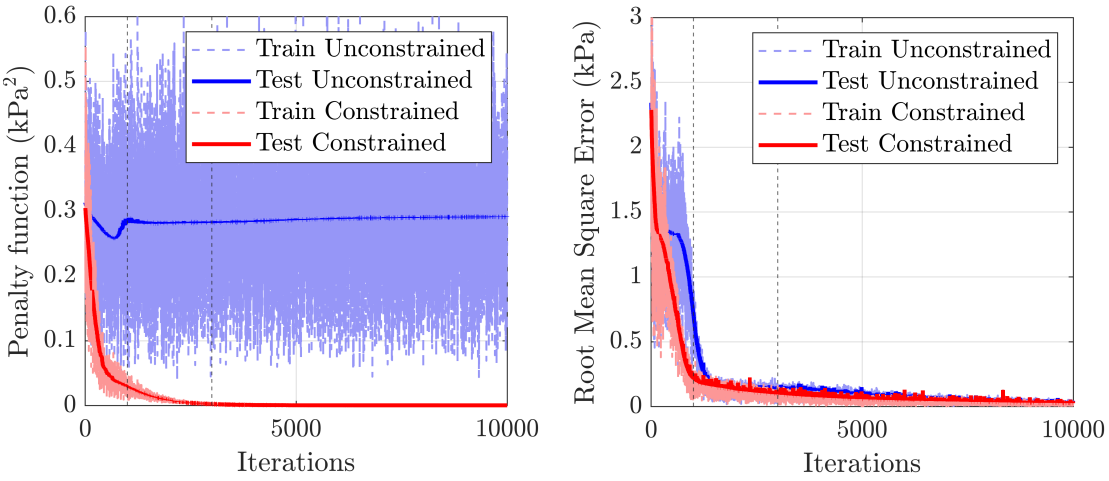
Table 5.1: Physical parameters for the problem with fixed geometry. These parameters are arbitrary but are reported for reproducibility purposes.

As the learning algorithm, a gradient descent optimiser was selected with learning rate parameter $\beta = 0.001$. At each training step, $n = 4$ data points were selected, enough for our purposes. For the constrained network, that is the PGNNIV, we chose a penalty parameter $p = 0.01\,\mathrm{Pa}^2\mathrm{s}^2/\mathrm{m}^6$. $N_{\text{test}} = 1000$ samples were randomly generated for the testing procedure.

Fig. 5.8 shows the value of the root mean square error (RMSE) and the penalty (PEN) functions along with the training iterations. The effect of including the penalty term (related to the physics of the problem) is clearly illustrated in Fig 5.8, which shows that the RMSE has a faster decay in the early learning steps. Indeed, the error converges to the same value when the number of iterations increases. Although there is not a general recipe for the model improvement and each benchmark problem requires its own strategy, as discussed in the broad bibliography on ANN (Nielsen, 2015), the fundamental conclusion here is that the constrained network does not necessarily improve the accuracy of the model if they both have the same network topology and metaparameters, but the introduction of physical constraints does speed-up the network convergence. This speed-up is also explained by the evolution of the penalty term value PEN (Fig. 5.8b) for both neural networks: for the unconstrained one, this term is not included in the penalty function and therefore it is non necessarily decreasing. Of course, since the unconstrained network does not force its fulfilment and the error RMSE also goes to zero, it is clear that, with this network topology, there is not a global minimum solution but many local minima. However, the convergence to one of these minimal solutions is accelerated in the constrained case and for, let's say, $N = 600$ iterations, the behaviour of the unconstrained network is suboptimal. Unless the whole underlying parametric model is assumed as known, $H = H(\cdot, \boldsymbol{\lambda})$, what would place us in a case analogous to classical parametric fitting via optimisation procedures, there is no simple way of defining a reduced enough network to guarantee both the required abstraction capability (generalisation) and global minimal requirements (specificity), reducing the computational cost.

Fig. 5.9 shows the accuracy of constrained and unconstrained neural networks after $N = 1000, 3000, 10000$ iterations (marked with a dashed bar in Fig. 5.8) when compared to the analytical solution. As explained before, the performance of both networks, if we assume convergence, is similar and only the learning rate, not the accuracy, is improved by the constrained network. The error included in Fig. 5.9 was computed as:

$$E_{\mathcal{L}_2} = \left(\int_0^{10} (\hat{\Upsilon}(q) - \Upsilon(q))^2 \, dq \right)^{\frac{1}{2}}, \tag{5.66}$$



(a) RMSE (RMSE $= \sqrt{\text{MSE}}$). **(b)** Penalty function (PEN).

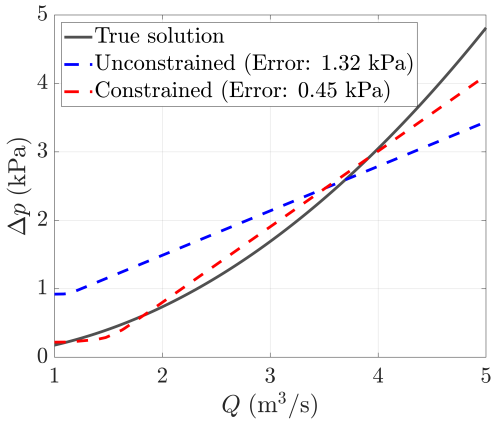
Figure 5.8: RMSE and penalty function. Note that, as the topology and parameters of the network between the v and Δp layers are the same, the asymptotic trend of both RMSE functions is the same. Only a modification in the network topology associated with the model, that is, in $\boldsymbol{H}$, would improve the accuracy of the method. The main difference between the networks is the evolution of the value of the penalty function: while for the unconstrained network no penalty term associated with the fundamental physics is added to the loss function, for the constrained network, the penalty term in the loss term (OF) ensures the fulfilment of the constraint, providing physical meaning to the neurons of some internal layers.

where $\hat{\Upsilon}(q)$ is the network-predicted pressure drop and $Y(q)$ is the true solution using the Hazen-Williams loss model. In addition to the convergence speed-up, the method computes the true value of the internal variables v_1, v_2 and v_3. We will come back to this in more detail later.

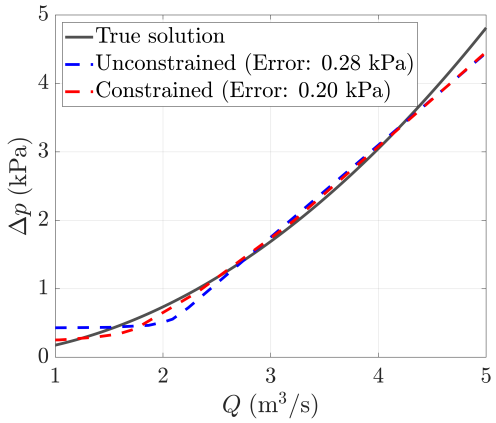
Variable geometry

The problem may be enriched by taking into account some geometrical aspects. For example, we can consider the two pipe lengths, L_1 and L_2 as extra input variables. This adds a double benefit: it allows us to consider variable geometries and, if some problem parameters are known, the neural network may be simply adapted and simplified to include more physical knowledge. Indeed, for $\xi = 0$, we know that all pressure drop is associated with the distributed head loss along the two stretches of lengths L_1 and L_2 so that the hidden layer may be replaced by a layer with two neurons, whose relationship with the output neuron will be $\Delta p = L_1 y_{3,1} + L_2 y_{3,2}$. With these considerations, the third hidden layer also acquires physical meaning (the local pressure drop per unit length at stretches 1 and 2).

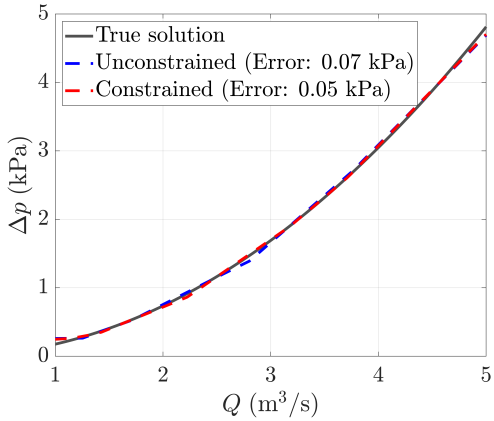
It is important to note that the constraint in the first hidden layer is now crucial because the input variables have different dimensions and the normalisation and the ReLU activation function ϕ acting between the first and second layer are complemented by the constraint indicating that the lengths L_1 and L_2 do not influence the flow velocity.



(a) $N = 1000$.



(b) $N = 3000$.



(c) $N = 10000$.

Figure 5.9: Comparison of the network output with the true model. Both models present a similar error. The effect of the constraints is not to improve the accuracy, but to speed-up the convergence, besides giving physical interpretation to some of the internal layers.

Neural layers are mathematically defined as follows. Given $\boldsymbol{x} = (q, L_1, L_2)$:

$$\boldsymbol{y}_1 = \phi(\boldsymbol{x}\boldsymbol{W}_1 + \boldsymbol{b}_1), \tag{5.67a}$$
$$\boldsymbol{y}_2 = \phi(\boldsymbol{y}_1\boldsymbol{W}_2 + \boldsymbol{b}_2), \tag{5.67b}$$
$$\boldsymbol{y}_3 = \phi(\boldsymbol{y}_2\boldsymbol{W}_3 + \boldsymbol{b}_3), \tag{5.67c}$$
$$\boldsymbol{y}_4 = y_{3,1}L_1 + y_{3,2}L_2 = \Delta p. \tag{5.67d}$$

The constrained network includes the same constraint as before, relating flow and velocities. The physically-guided representation of this new constrained ANN is illustrated in Fig. 5.10.

All physical and geometrical parameters are the same as in the preceding example, except that $\xi = 0$, $S_1 = S_2 = 1$ m^2, since the effect of the pipe expansion is not the relevant phenomenon here, $\kappa_1 = 140$ and $\kappa_2 = 100$ in order to differentiate between the two stretches. L_1 and L_2 were uniformly generated between 0 and 10. As before, a gradient descent optimiser was selected with a learning rate parameter $\beta = 0.003$.

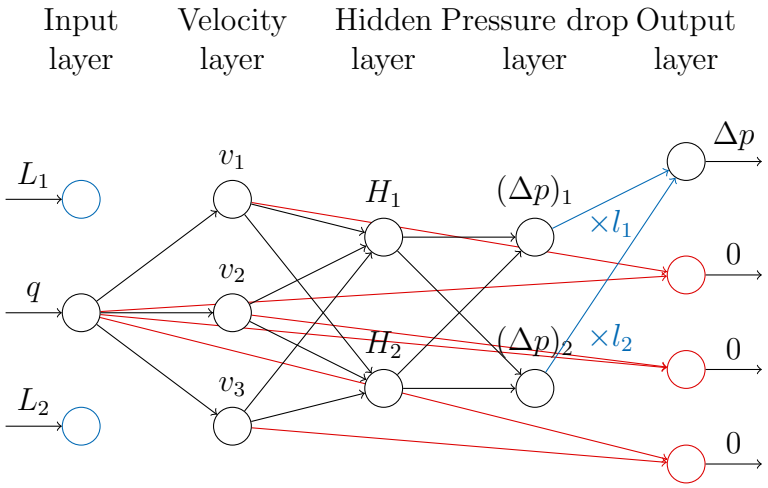


Figure 5.10: Physically augmented neural network for the geometry-dependent problem. The red lines illustrate the velocity definition in terms of flow, while the blue lines represent the geometry inclusion by means of the momentum conservation equation.

At each training step, $n = 100$ data points were selected. The same value for the penalty parameter ($p = 0.01\,\mathrm{Pa^2 s^2/m^6}$) was selected and $N_{\text{test}} = 10000$ samples were randomly generated for the testing procedure.

As in the previous case, Fig. 5.11 shows the convergence curves for the RMSE and penalty functions demonstrating good convergence and no overfitting. Fig. 5.12 shows the accuracy of the constrained and unconstrained neural networks after $N = 20000$ iterations. In that case, in addition to the speed-up of the convergence, the PGNNIV shows a better accuracy, as expected, because the topology of the network was thought in a physical sense, with the last hidden layer having a physical interpretation (the internal variable w_i, $i = 1, 2$).

For comparison purposes, Table 5.2 shows the statistics of the absolute value of the relative error, $|\varepsilon_r|$, with

$$\varepsilon_r = \frac{\hat{\mathrm{Y}}(q, L_1, L_2) - Y(q, L_1, L_2)}{Y(q, L_1, L_2)}, \tag{5.68}$$

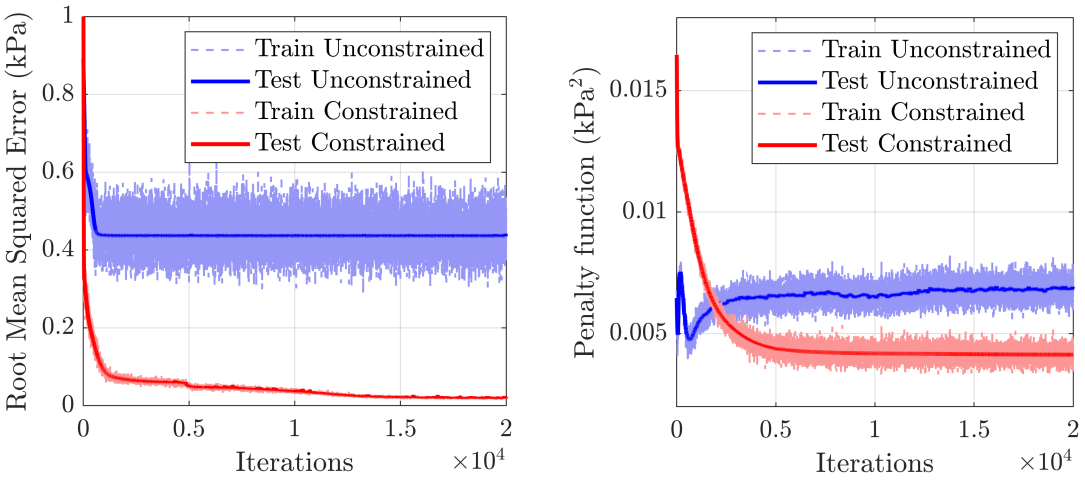
obtained for both the unconstrained and constrained networks, when $100 \times 100 \times 100$ values of q, L_1 and L_2 were sampled in $[1; 5] \times [0; 10] \times [0; 10]$. It is clear that the effect of the constraints is to reduce the relative error together with its variability.

Network	Minimum	Q_1	Q_2	Q_3	Maximum	(Mean $\pm$ Std. error)
Unconstrained	2.0×10^{-6}	2.3×10^{-1}	5.0×10^{-1}	8.2×10^{-1}	2.8×10^{2}	$(8.28 \pm 0.03) \times 10^{-1}$
Constrained	4.9×10^{-8}	1.2×10^{-2}	2.1×10^{-2}	4.8×10^{-2}	5.8×10^{-1}	$(4.433 \pm 0.009) \times 10^{-2}$

Table 5.2: Statistics of the absolute relative error $|\varepsilon_r|$ for the two networks. The PGNNIV improves ostensibly the prediction statistics.

Prediction of the internal variables and model identification

Besides the constraints associated with physical principles, we evaluate the effect of adding model constraints to the network. We shall also discuss the explanatory



(a) RMSE (RMSE $= \sqrt{\text{MSE}}$). **(b)** Penalty function (PEN).

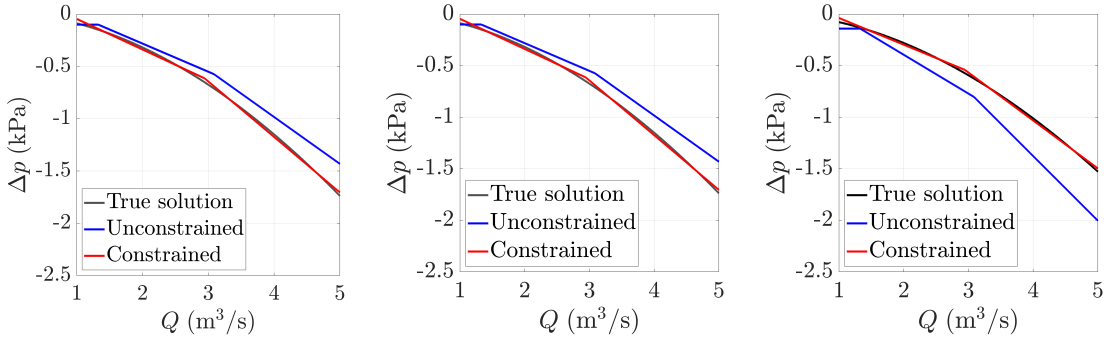
Figure 5.11: RMSE and penalty function for the network including geometry. The physical constraints give the hidden layers the correct physical interpretation, since the integration constraint, $\Delta p = w_1 L_1 + w_2 L_2$ is correctly formulated. This is achieved thanks to the effect of the penalty term, which gives the PILs their correct interpretation. As in Fig. 5.8, the difference between the constrained and the unconstrained networks is the fulfilment of the constraints after training.

capacity of the presented method in learning the internal physical variables and, if it is the case, the model parameters. As previously explained, there are two types of equations used in the formulation of the hydraulic head loss in a pipe:

- **Fundamental principles**: Mass conservation ($S_1 v_1 = S_2 v_2$), linear momentum conservation ($\Delta p = w_1 L_1 + (\Delta p)_e + w_2 L_2$), and energy (hydraulic head) balance ($\Delta h = (\Delta h)_1 + (\Delta h)_e + (\Delta h)_2$) (the subscripts indicate the corresponding segment of the pipe).
- **Constitutive equation**: These equations relate the hydraulic head loss (which is directly related to pressure drop by means of Bernoulli equation) to the fluid velocity along the streamline. E.g. the Darcy-Weisbach or Hazen-Williams equation for losses associated with the pipe roughness and Borda-Carnot's for the eddy energy dissipation due to the pipe expansion.

It is clear that there is no general need for including constraints related to the constitutive equation, following the approach of the two previous examples. This is, indeed, contraindicated if there is no knowledge about the underlying behaviour of the fluid (physical nature, regime...). However, it might be interesting in at least two circumstances:

1. **Model selection**: We want to select among many candidate models able to capture, from a macroscopic point of view, the fluid behaviour. For instance, in the present example, we shall choose between the Darcy-Weisbach and the Hazen-Williams models for the hydraulic losses. Indirectly, this may give us



(a) $l_1 = 5\,\text{m}$, $l_2 = 5\,\text{m}$. **(b)** $l_1 = 2\,\text{m}$, $l_2 = 8\,\text{m}$. **(c)** $l_1 = 7\,\text{m}$, $l_2 = 3\,\text{m}$.

Figure 5.12: Exact and predicted solution for different lengths of the segments. The specified network topology and the introduction of the physical constraints are responsible for the network convergence. Results are given for different segment lengths.

information on the fluid regime, since the relationship between the Darcy factor f_D and the fluid velocity is fixed ($f_D = \frac{64\nu}{v\Phi}$, for instance, for laminar regime).

2. **Structure physical discovering**: Usually, the model parameters are related to some physical properties that give us insight into the nature, structure, or geometry of the problem. In the present example, for the Hazen-Williams model, κ_1 and κ_2 are related to the roughness of the pipe segments and in the Borda-Carnot model, ξ is related to the gradualness of the expansion.

As the aim is to predict the state model, it is clear that now the output for each data point must be a triplet of values $((\Delta p)_1, (\Delta p)_e, (\Delta p)_2)$ corresponding to the pressure drop at segment 1, expansion and segment 2, respectively. Without this multiple-output consideration, it would be impossible to distinguish between effects in the whole pressure drop. In what follows, three neural networks (with and without constraints) are compared:

1. **Model-free approach**: PGNNIV where the physics (fundamental laws) are imposed via appropriate constraints in certain layers. This occurs when we add the constraints by means of functions $v_i = E_i(q) = \frac{q}{S_i}$, $i = 1, 2$.
2. **Model-based approach**: PGNNIV where both, physical and empirical (constitutive/state equations) laws are imposed.

   (a) Darcy-Weisbach model: This corresponds to the constraints given by Eqs. (5.60). Note that in that case, for the laminar regime, $f_{D_i} = \frac{64\nu}{v_i\Phi_i}$ and f_{D_i} are constant, while they depend on the pipe roughness in the rough turbulent regime.

   (b) Hazen-Williams model: This corresponds to the constraints given by Eqs. (5.61).

In the model-free network, the network topology is prescribed as:

$$\boldsymbol{y}_1 = \boldsymbol{y}_0 \boldsymbol{W}_1 + \boldsymbol{b}_1, \tag{5.69a}$$
$$\boldsymbol{y}_2 = \phi(\boldsymbol{y}_1 \boldsymbol{W}_2 + \boldsymbol{b}_2), \tag{5.69b}$$
$$\boldsymbol{y}_3 = \phi(\boldsymbol{y}_2 \boldsymbol{W}_3 + \boldsymbol{b}_3), \tag{5.69c}$$
$$\boldsymbol{y}_4 = \boldsymbol{y}_3 \boldsymbol{W}_4 + \boldsymbol{b}_4, \tag{5.69d}$$

with $\boldsymbol{y}_0 = \boldsymbol{x} = q$ and $\boldsymbol{y}_4 = \boldsymbol{y} = ((\Delta p)_1, (\Delta p)_e, (\Delta p)_2)$ and ϕ again a ReLU activation function. As before, a PIL is prescribed for the variables $\boldsymbol{y}_1$, that will be identified with v_1 and v_2 while the mass conservation is imposed via the constraint $v_i - \frac{q}{S_i} = 0$. Layers 2 and 3 are composed of $n_1 = n_2 = 15$ neurons.

Similarly, in the model-based network, we propose the following topology:

$$\boldsymbol{y}_1 = \boldsymbol{y}_0 \boldsymbol{W}_1 + \boldsymbol{b}_1, \tag{5.70a}$$
$$\boldsymbol{y}_2 = \boldsymbol{H}(\boldsymbol{y}_1; \boldsymbol{\lambda}), \tag{5.70b}$$

where $\boldsymbol{H}$ is the model equation, formulated in terms of the model parameters $\boldsymbol{\lambda}$, which are defined as $\lambda_1 = \xi$, $\lambda_2 = \Phi_1^\beta / \kappa_1$ and $\lambda_3 = \Phi_2^\beta / \kappa_2$ for the Hazen-Williams model and $\lambda_1 = \xi$, $\lambda_2 = \nu / \Phi_1$ and $\lambda_3 = \nu / \Phi_2$ for the Darcy-Weisbach model (in the laminar regime). Both neural networks are illustrated in Fig. 5.13.

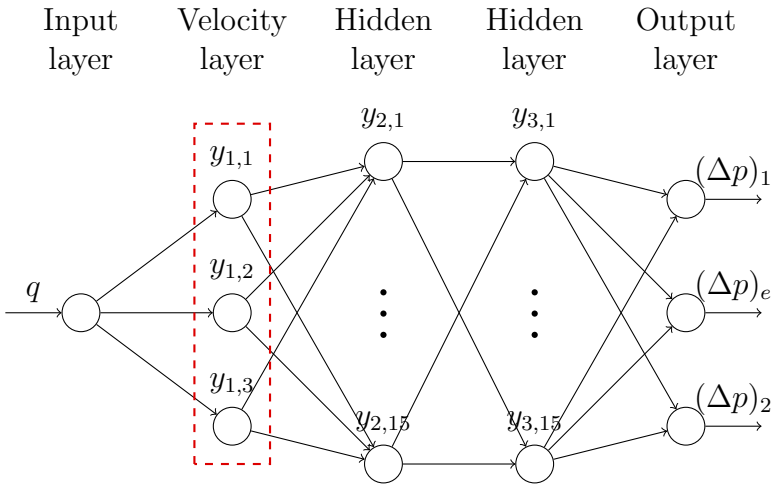
For the training process, the data input was randomly generated with a uniform distribution using the state model presented in equation (5.61) for $q \in [1.0; 5.0]\,(\mathrm{m}^3/\mathrm{s})$, that is, the Hazen-Williams model. The physical parameters used for the data generation are shown in Table 5.1. As a learning algorithm, a gradient descent optimiser was selected with a learning rate parameter $\beta = 0.0001$. At each training step, $n = 4$ data points are selected. For the constrained network, we chose a penalty parameter of $p = 0.01\,\mathrm{Pa}^2\mathrm{s}^2/\mathrm{m}^6$. $N_{\text{test}} = 1000$ samples were randomly generated for the testing procedure.

To evaluate the performance of all neural networks, we illustrate in Table 5.3 the statistics of the relative error of the predicted value (when compared to the analytical one) for the different variables involved in the problem:

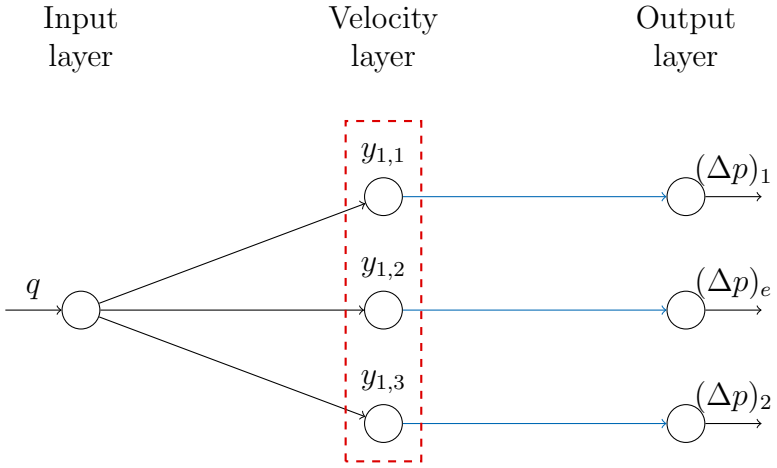
- Measurable variables (output variables), that is, the pressure drops $(\Delta p)_1$, $(\Delta p)_e$ and $(\Delta p)_2$.
- Non-measurable variables (internal variables), that is, the flow velocity at each segment, v_1 and v_2.

Figs. 5.14 and 5.15 illustrate the predictive capacity of the different networks in estimating the internal and measurable variables respectively for different values of q. Once the model-based network has converged, it is possible to extract the model parameters, whose relative error is illustrated in Table 5.4 .

The conclusion drawn is clear and natural. If we want predictive capacity, a model-free neural network is always preferred except if the underlying constitutive model is perfectly known (what is, in general, a strong assumption). A wrong model



(a) Model-free neural network.



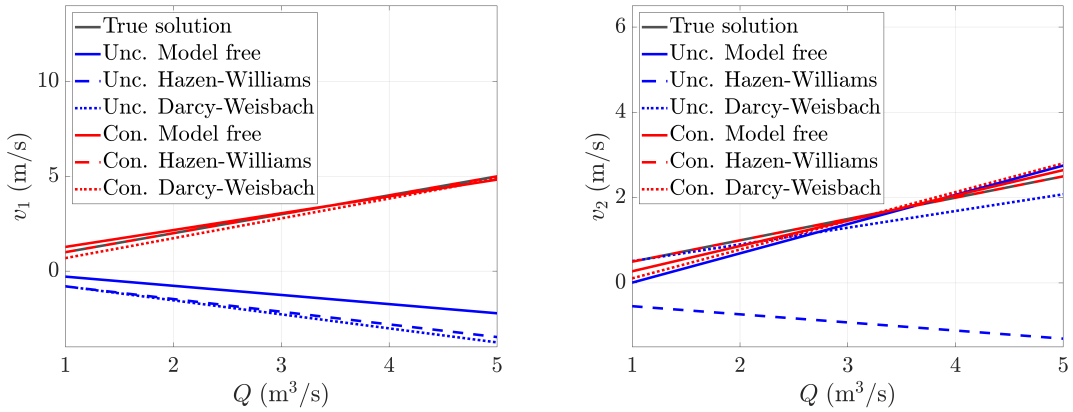
(b) Model-based neural network.

Figure 5.13: Model-free and model-based PGNNIV. The universal constraints are illustrated using the red dashed boxes. Model-based constraints are illustrated using blue lines.

assumption worsens the network accuracy with respect to a model-free one. Therefore, model-based networks can help in model identification and can shed light on the system's physical and geometrical structure. The specification of an incorrect underlying model affects both correctly specified variables ($(\Delta p)_e$) and those that are not ($(\Delta p)_i$, $i = 1, 2$) as the error is distributed in all the predicted variables, although the latter to a greater extent.

Performance evaluation

The first important property of the presented methodology, beyond its explanatory capacity, is the improvement of the performance with respect to other ANN methods. As PGNNIVs have internal constraints between layers (or, equivalently, a higher number of outputs with zero value) it is obvious that the search space for the optimal solution will be smaller (see Fig. 5.3). This observation leads to important consequences that are quantified next. We present the main features



(a) v_1. **(b)** v_2.

Figure 5.14: Predictive capacity of each ANN in estimating the internal variables. The constrained network is the only one able to predict accurately the internal variables. Model specification improves the accuracy only when the model assumed is the correct one.

		Measurable variables						Internal variables			
		$(\Delta p)_1$		$(\Delta p)_e$		$(\Delta p)_2$		v_1		v_2	
MF	Unc.	−0.031	(±0.002)	0.030	(±0.002)	−0.070	(±0.002)	1.405	(±0.001)	0.211	(±0.007)
	Con.	−0.049	(±0.003)	0.023	(±0.002)	−0.03	(±0.02)	0.059	(±0.002)	0.096	(±0.003)
MB (HW)	Unc.	−1.000	(±0.000)	0.008	(±0.000)	−1.000	(±0.000)	1.723	(±0.000)	1.666	(±0.004)
	Con.	0.000	(±0.000)	0.000	(±0.000)	0.000	(±0.000)	0.000	(±0.000)	0.000	(±0.000)
MB (DW)	Unc.	−1.000	(±0.000)	0.003	(±0.000)	−0.56	(±0.02)	1.764	(±0.000)	0.120	(±0.001)
	Con..	−0.37	(±0.01)	0.021	(±0.001)	−0.165	(±0.004)	0.097	(±0.002)	0.165	(±0.006)

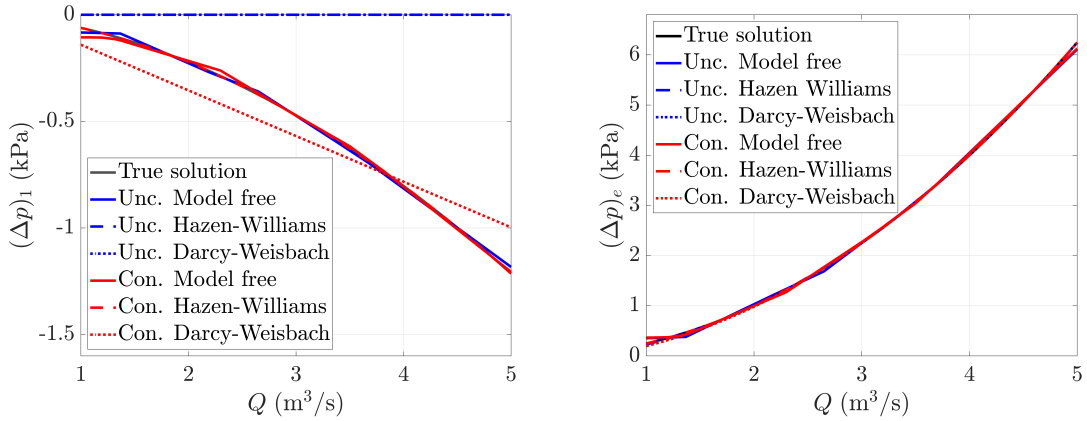
Table 5.3: Statistics of the relative error ε_r for the different ANN analysed. Data are presented as mean value (± Std. error). MF: Model-Free. MB (HW): Hazen-Williams model-based. MB (DW): Darcy-Weisbach model-based. Errors below 1‰ are not reported.

		Parameter		
		λ_1	λ_2	λ_3
MB (HW)	Unc.	1.68×10^{0}	1.00×10^{0}	9.97×10^{-1}
	Con.	5.00×10^{-6}	1.95×10^{-6}	1.43×10^{-6}
MB (DW)	Unc.	1.75×10^{0}	1.00×10^{0}	5.90×10^{-1}
	Con.	4.20×10^{-1}	2.80×10^{-1}	6.50×10^{-1}

Table 5.4: Relative error ε_r of the model parameters for the different model-based networks. MB (HW): Hazen-Williams Model-Based. MB (DW): Darcy-Weisbach Model-Based.

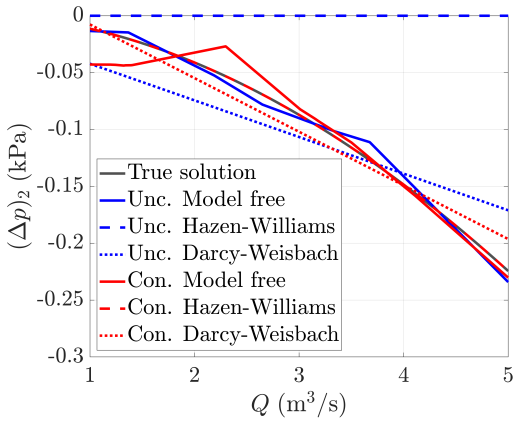
with some important figures showing our claims. For a more comprehensive analysis showing some additional figures about the different variable prediction, the reader is encouraged to consult the published version of this chapter in Ayensa-Jiménez et al. (2021a).

- **Convergence speed-up**



(a) $(\Delta p)_1$.

(b) $(\Delta p)_e$.



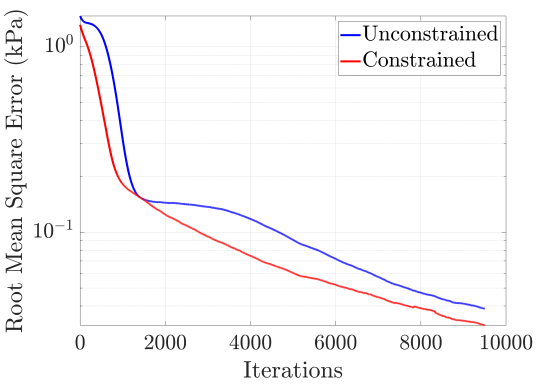
(c) $(\Delta p)_2$.

Figure 5.15: Predictive capacity of each neural network in estimating the measurable variables. The unconstrained and constrained networks have a similar capacity in estimating the measurable variables. Model specification improves the accuracy only when the model assumed is the correct one.

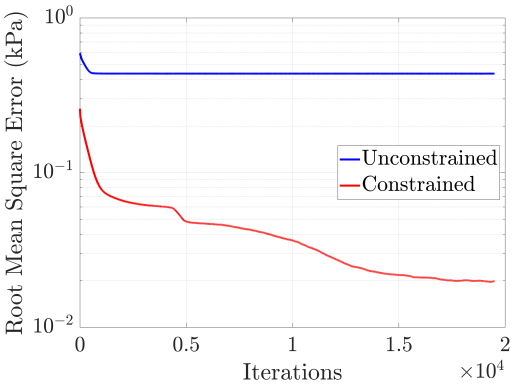
Fig 5.16 shows the effect of the constraints in the network convergence for the different problems: the fundamental prediction problem (Fig. 5.16a), the geometry inclusion (Fig. 5.16b) and the model inclusion (Fig. 5.16c). They show the training process for both ANNs (constrained and unconstrained) in terms of the RMSE.

- **Data need decrease**

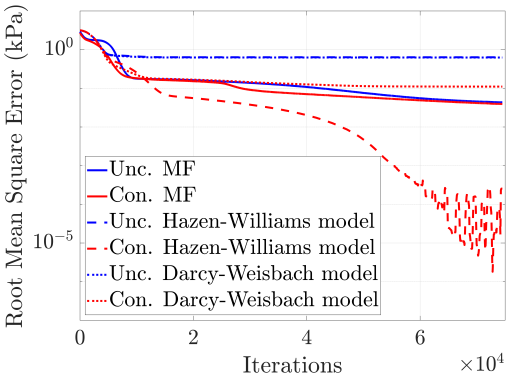
  Another way of seeing the speed-up capability of PGNNIV is to focus on data need. In many engineering problems, especially in those related to material sciences (solid and fluid mechanics, electromagnetism, optics, etc.), there is a lack of experimental data due to technical or economic reasons (Small Data framework), so, reducing the amount of training data required in the process is essential. The effect of the constraints will be evaluated in terms of the amount of data required, using the pipe flow prediction problem. For this purpose, the learning curve is evaluated for a varying size of the data-set, $M = 2, 10, 50$. The number of iterations was set as $N = 3000$ and the batch size is fixed to $n = M$



(a) Prediction problem.



(b) Problem with variable geometry.



(c) Problem with model inclusion.

Figure 5.16: Convergence comparison of constrained and unconstrained ANN. Convergence curves are smoothed with a moving average filter (window $W = 500$) for easier comparison. For all cases, the constrained neural network shows a convergence speed-up since the search space is smaller. The accuracy is not necessarily improved but the network convergence is accelerated. Adding constraints (see **(c)**) always speeds-up the convergence, regardless of whether the accuracy is improved or not.

(that is, the whole data-set is evaluated in each training step). The RMSE convergence curves are depicted in Fig. 5.17.

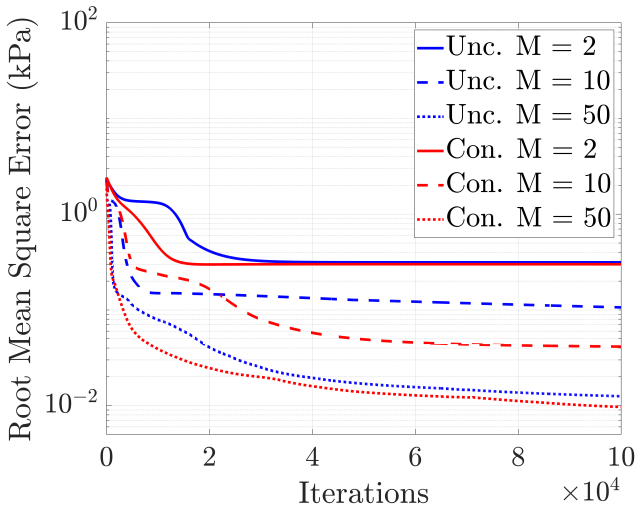


Figure 5.17: Learning performance for the two networks and different data-set sizes. Convergence curves are smoothed with a moving average filter (window $W = 500$) for an easier comparison. For small data-sets, PGNNIV has an impact not only on the convergence speed-up, but also on the network accuracy.

From Fig. 5.17, it is possible to make some particular conclusions:

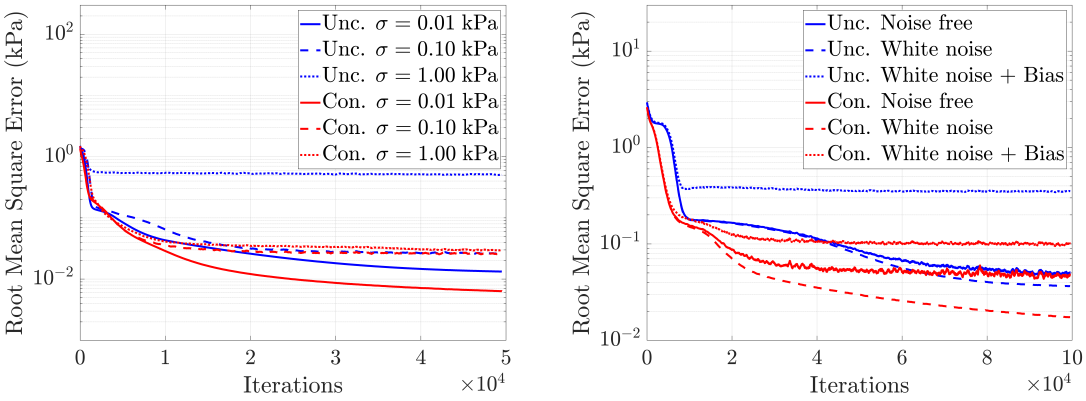
- If both networks have the same final accuracy, given a fixed number of iterations, the physically constrained network performance is better (lower error) than the unconstrained one.
- The impact of the data-set size is more gradual in the constrained network. Indeed, the curves associated with unconstrained networks are more step-like (particularly for small data-sets), when compared to the constrained ones. This is very important in order to detect network convergence stabilisation.

In brief, constrained neural networks accelerate the learning process in such a way that they are able to discover important new features with less data (small data problems), which is extremely important in a practical engineering context.

- **Filtering capacity improvement**

  Here, the noise filtering capacity of the constrained network is explored. In the prediction pipe flow problem, the data-set has been considered noise-free. That is, a data-set was generated directly from the Eq. (5.61). Here, we compare the performance of both neural networks when working with noisy data, a more realistic situation in experimental problems and data acquired from sensors. To show the noise impact in the learning process, let us assume a data-set with added Gaussian noise, i.e. $x = \bar{x} + Z$ with $Z \sim \mathcal{N}(0, \sigma)$ for $x = q, \Delta p$ and $\sigma = 0.01, 0.10, 1.00\,\text{kPa}$. The RMSE convergence curves are illustrated in Fig. 5.18a. The effect is even stronger if the constraint acts on the output layer. For instance, let us consider the network with output $((\Delta p)_1, (\Delta p)_e, (\Delta p)_2)$ and let us add the constraint $(\Delta p)_1 + (\Delta p)_e + (\Delta p)_1 = \Delta p$, where Δp is another measured

variable (the total pressure drop). In addition to a noise of $\sigma = 0.1\,\text{kPa}$, we consider also the possibility of adding a systematic bias of $-0.2\,\text{kPa}$ to all the measured variables. The results are shown in Fig. 5.18b.



(a) Effect of the noise level.

(b) Effect of white noise and bias.

Figure 5.18: Effect on the learning performance of the data error. Convergence curves are smoothed with a moving average filter (window $W = 500$) for easier comparison. The noise has a much lower impact on the network convergence, both on speed-up and accuracy. Bias is partially corrected and noise filtered in constrained networks. Some level of noise may improve the network accuracy, as reported in the literature (Grandvalet et al., 1997; Holmstrom and Koistinen, 1992; Skurichina et al., 2000).

As main results:

- The performance and learning capacity of both networks decrease with noise. This is in agreement with other works (Jim et al., 1996; Kalapanidas et al., 2003; Rolnick et al., 2017).
- The impact of the noise is lower in the network convergence, as the RMSE curves are closer for the PGNNIV networks.
- The noise has an impact not only on the network convergence rate, but also on the network accuracy, as the curves associated with constrained networks are strictly under the curves associated with the unconstrained ones for $\sigma = 0.01\,\text{kPa}$ and $\sigma = 1.00\,\text{kPa}$. In other words, the physical constraints are able to partially filter the noise.
- Other systematic errors as bias may be partially corrected by the addition of the constraints to the network. That is, PGNNIV presents bias correction capability.

- **Extrapolation capability**

  Let us consider the problem with the constitutive model for head loss estimations. We evaluate the network performance in predicting values of the pressure drops out of the learning data-set, that is, for $q \geq 5$. Fig 5.19 shows the relative errors statistics (mean and standard error bar) of the different variables (internal and measurable variables) when extrapolating to the new values of q.

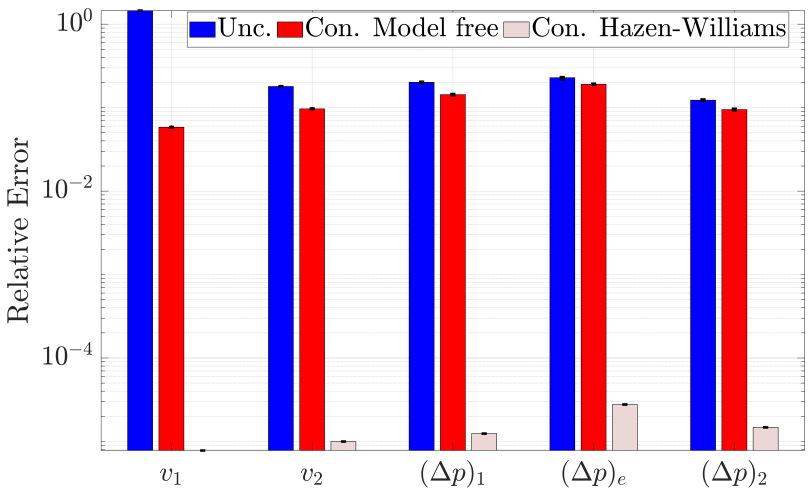


Figure 5.19: Evaluation of the extrapolation capacity. PGNNIV always improve the extrapolation capacity, when compared to standard ANNs. This improvement is significant even for the measurable variables, although the internal variables are of course better estimated. For model-based networks, the extrapolation capacity is highly increased. The relative error is plotted as mean ± std. deviation.

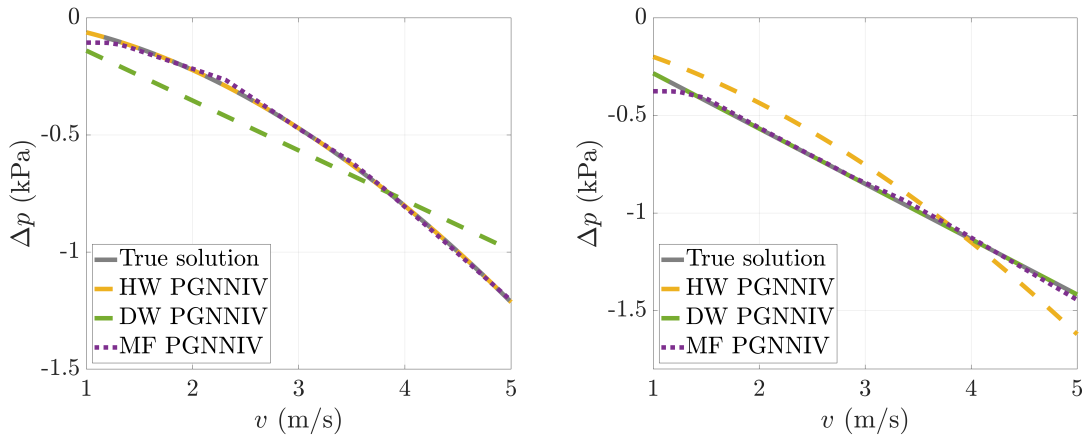
We conclude:

- Even if PGNNIV are designed, among other purposes, for a good estimation of the internal variables, their physically-based nature enables the estimation of the measurable variables out of the learning data-set, which increases its generalisation capacity.
- Model-based PGNNIV networks are, in a certain sense, similar to standard parameter fitting algorithms so, if the model is assumed to be known, the converged network has no error in predicting the values following the model assumptions, even for data out of the training range. However, if we compare PGNNIV to classical fitting procedures, the former has the advantage of all specific hardware and software relative to ANN technology, as explained before.

Internal state discovering

Despite the characteristics that make PGNNIV faster, less data-demanding and more robust than common ANN, these are not the main causes that justify its use in scientific and engineering problems. Indeed, it is due to their physical explanatory capacity. This can be exploited in two ways: i) accurate prediction of non-measurable internal variables, such as velocities, fluxes or viscous losses and ii) discovery of the hidden physics in an internal state equation such as the Hazen-Williams'.

For example, in the prediction problem, we have defined three PGNNIVs: a model-free (MF) PGNNIV, a Hazen-Williams model-based (MB) PGNNIV and a Darcy-Weisbach model-based (MB) PGNNIV. The presented methodology allows to discover the internal state equation, and to select the best model among several ones. In Fig. 5.20, we illustrate the relationship $\Delta p = H(v)$, exported from the network after reaching convergence, for the three PGNNIVs and two data-sets, with data derived from Hazen-Williams and Darcy-Weisbach models respectively. It is clear that MF-PGNNIV gives good results for both models, although the two MB-PGNNIV are better suited for the two specific cases. MF-PGNNIV have, therefore, more explanatory capacity, while MB-PGNNIV have more predictive capacity for the specific considered cases. This is another illustration of the trade-off between explanatory and predictive capacity.



(a) Hazen-Williams model. **(b)** Darcy-Weisbach model.

Figure 5.20: Model explanatory capacity for the prediction problem. MF-PGNNIV has a better explanatory capacity as it is able to provide good results for data-sets following different models and to discriminate between them. MB-PGNNIV give perfect fit when the model is truly the expected one, but gives us worse results if not.

5.3.1.3 The characterisation problem

Solving the characterisation problem

Recall that the aim is now to characterise some of the parameters of the pipe segments for a given set of values (q, p_0, p_1, p_2), that is, the relationship to be learned now is $(q, p_0, p_1, p_2) \rightarrow (\kappa_1, \kappa_2)$.

As Eq. (5.63) is complex and highly nonlinear, it is expected that the number of hidden layers required will be large. Thus, we refer to this part of the network as a DL box with its own internal number of layers, neurons and connectivity.

For the problem presented, the PGNNIV topology is shown in Fig. 5.21. The DL Box is a MLP with 5 dense layers of 20, 40, 80, 40 and 20 neurons respectively, with activation functions of ReLU type. The network performance is compared to the same network in which the physical constraint has not been included. The difference between the constrained and the unconstrained networks is that the penalty parameter is set to zero for the unconstrained network.

The training data-set was created using the analytical model, with κ_1 and κ_2 randomly generated between $\kappa = 80$ and $\kappa = 140$ (that are standard values of the roughness parameter) and a flow q varying from $1\,\mathrm{m}^3/\mathrm{s}$ to $5\,\mathrm{m}^3/\mathrm{s}$. For the training process, we used batches of $n = 300$, a penalty parameter of $p = 0.001$ and a learning rate parameter of $\beta = 1 \times 10^{-5}$ for the gradient descent optimiser. The input x_i and output y_i values are normalised between their maximal and minimal value as:

$$\hat{x}_i = \frac{x_i - x_{\min}}{x_{\max} - x_{\min}}, \tag{5.71a}$$

$$\hat{y}_i = \frac{y_i - y_{\min}}{y_{\max} - y_{\min}}, \tag{5.71b}$$

where $x_{\max}$ and $x_{\min}$ are the maximal and minimal values for the input and $y_{\max}$ and $y_{\min}$ the maximal and minimal values for the output, respectively. $N_{\text{test}} = 100$ test values are used to evaluate the performance.

As for the prediction problem, Fig. 5.22 shows the performance of both neural networks for the characterisation one. As in the previous case, the constraints accelerate the convergence of the network.

The accuracy is shown in Fig. 5.23 where the predicted values of κ_1 and κ_2 are compared with the theoretical ones for $N_{\text{test}} = 100$ test values. The figure shows a good performance of the neural network although it decays close to the boundaries, which is natural, since the neural network has been trained with data (roughness coefficient) $\kappa \in [80; 140]$.

Performance evaluation

As for the prediction problem, the constrained neural network improves the unconstrained one, having a faster convergence and fewer data requirements, a better filtering capacity and a higher extrapolation capacity. With the intention of not being repetitive, we illustrate only the first feature in Fig. 5.24.

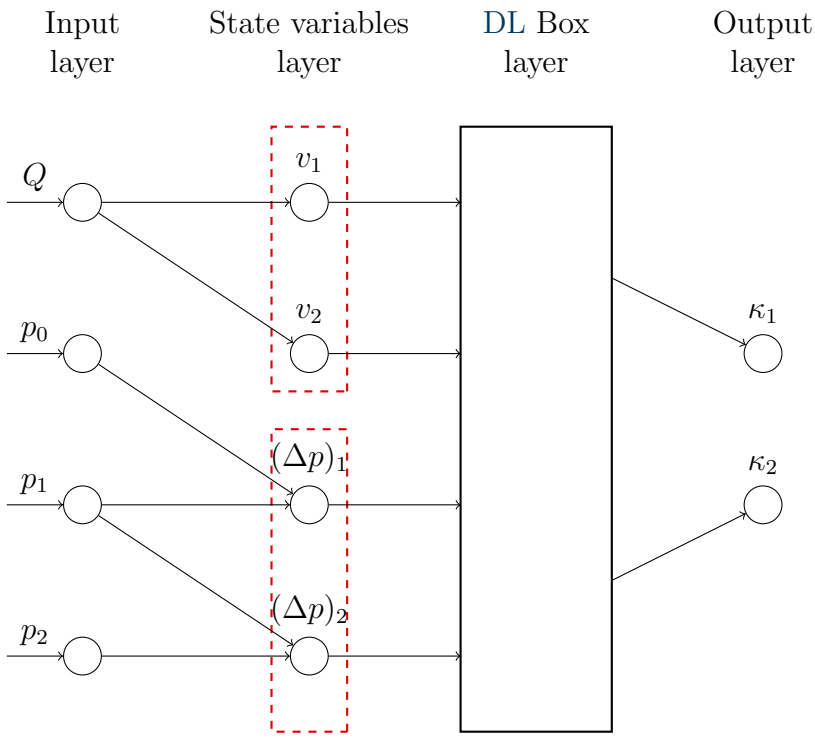
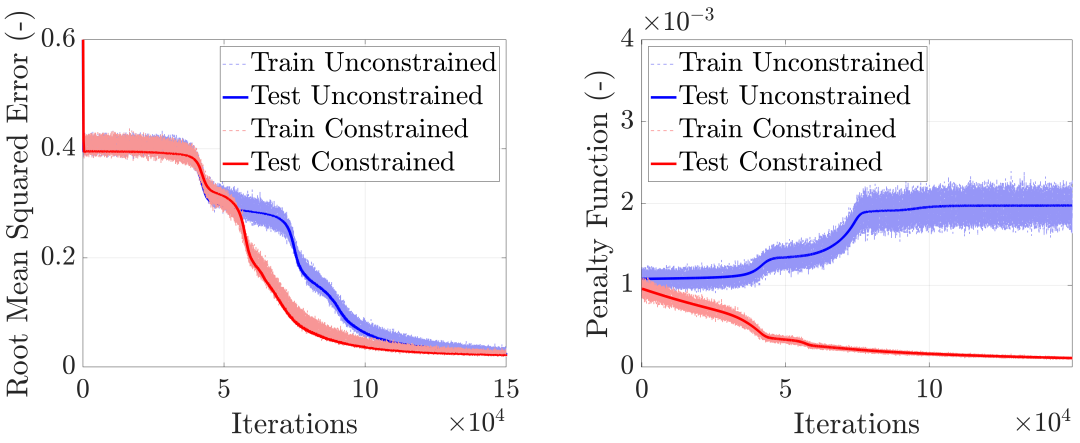


Figure 5.21: PGNNIV for the characterization problem. Red dashed rectangles represent physical constraints on neurons. The relationship between flow and velocities is imposed in v_1 and v_2, $v_i = \frac{q}{\Sigma_i}$ and the definition of the incremental pressure drop is imposed in $(\Delta p)_1$ and $(\Delta p)_2$, $(\Delta p)_i = p_i - p_{i-1}$.

Internal state discovering

The network explanatory capability may be further explored in system characterisation. First, under convergence assumptions, the PGNNIV may be used for predicting the quantitative relation between the different internal variables. But also, the network may provide major features of the structure of the empirical model, for instance, structural dependence and separability. More in detail, the explanatory capability may be also explored for accurately predicting the $(q, p_1, p_2, p_3) \rightarrow (\kappa_1, \kappa_2)$ relation and also for learning about the model separability. Fig. 5.25 shows both the real and predicted values for κ_1 and κ_2 for $q = 3$ m^3/s and different values of $(\Delta p)_1$ and $(\Delta p)_2$. As it may be seen in Fig. 5.25, the predicted values are close to the real ones, but a more important fact is that the PGNNIV, thanks to its topology, is able to separate the dependency between variables, that is $\kappa_i = H_i((\Delta p)_i)$ for $i = 1, 2$ instead of the general case $\boldsymbol{\kappa} = \boldsymbol{H}(\boldsymbol{\Delta} p)$. Thus, some features of the model become explainable, in this case, the fact that the pipe roughness is related only with the pressure loss at this stretch.



(a) RMSE. **(b)** Penalty function (PEN).

Figure 5.22: RMSE and penalty function for the characterization problem. Even if, in all cases, the Deep Learning Box has not enough power to capture the complex nonlinear model perfectly (observe that RMSE does not converge to 0) the effect of the penalty is to speed-up the network convergence.

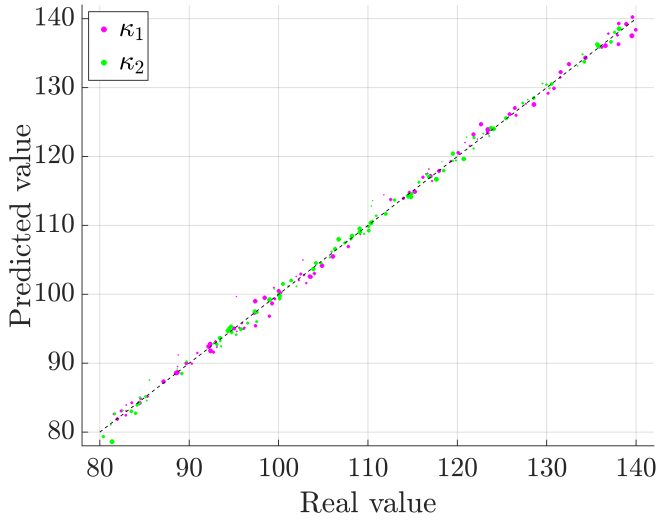


Figure 5.23: Results predicted by the PGNNIV. Both predicted values κ_1 and κ_2 are compared to the theoretical ones. Line $y = x$ identifies a perfect estimation. The size of the dot is proportional to the value of the input flow: the model tends to give worse results for smaller flows.

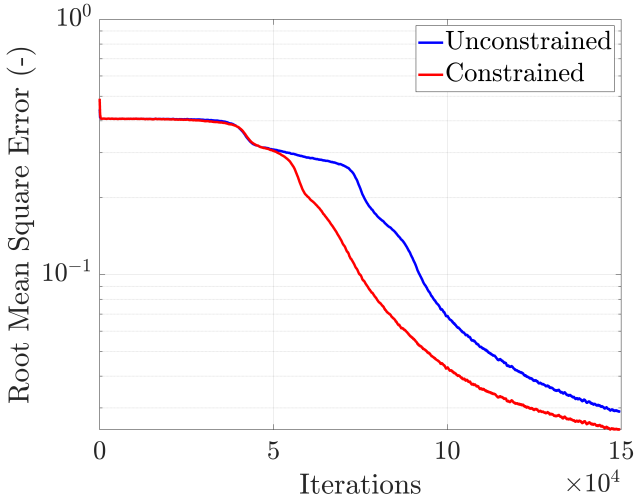
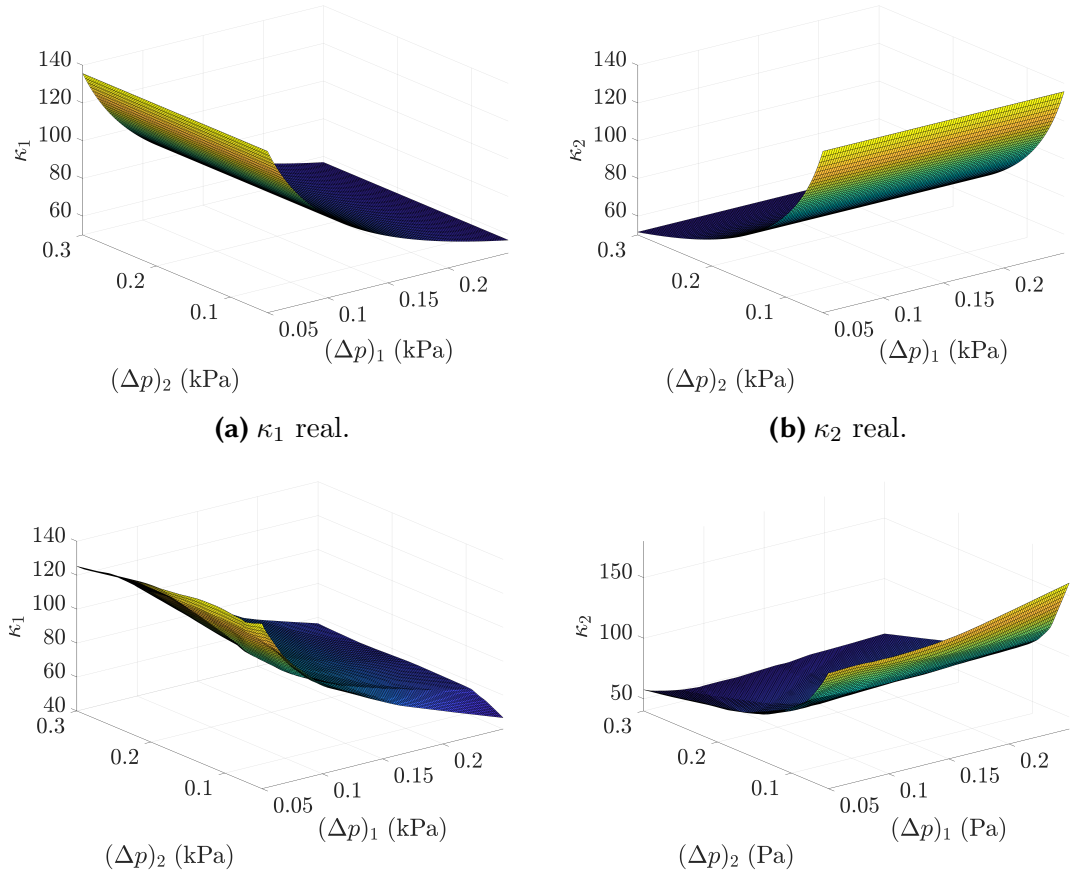


Figure 5.24: Convergence comparison of constrained and unconstrained ANN for the characterisation problem. Convergence curves are smoothed with a moving average filter (window $W = 500$) for easier comparison. For all cases, the constrained neural network shows a convergence speed-up since the search space is smaller. Again, the accuracy is not necessarily improved but the network convergence is accelerated.



(a) κ_1 real.

(b) κ_2 real.

(c) κ_1 predicted.

(d) κ_1 predicted.

Figure 5.25: Model explanatory capacity for the characterization problem. Values of κ_1 and κ_2 predicted compared to the real ones for different values of $(\Delta p)_1$ and $(\Delta p)_2$ and $q = 3\ \mathrm{m}^3/\mathrm{s}$.

5.3.2 Stationary heat/diffusion equation

To illustrate the convenience of the presented methodology for continuum physics problem, we deal with a simple although very illustrative example: the diffusion equation.

5.3.2.1 Problem statement

The next example illustrates the use of PGNNIV in continuum physical problems after discretisation. Let us suppose the following PDE corresponding to a diffusion problem:

$$\boldsymbol{\nabla} \cdot (\boldsymbol{K} \boldsymbol{\nabla} u) = f \tag{5.72}$$

where u is the solution field and f is the source term. This problem is ubiquitous in Physics and Engineering. For example, it is used, for instance, in stationary heat transfer conduction problems with u the temperature, also in steady-state, water seepage in soil mechanics, or in electrostatics, among others. Eq. (5.72) is the combination of two different laws:

- **A fundamental functional principle** as it is energy conservation (heat transfer), mass conservation (diffusion) or Gauss law (electrostatics), that states that $\boldsymbol{\nabla} \cdot \boldsymbol{q} = f$, where $\boldsymbol{q}$ is the flow vector (heat flow, mass flow or electric displacement field) and f is the source term (heat source, mass source or electric charge density).
- **A constitutive functional equation** as it is the Fourier law (heat transfer), Fick's law (diffusion) or dielectric behaviour (electrostatics), relating the flux variable $\boldsymbol{q}$ that plays the role of internal state field (non-measurable if no additional assumption is made, e.g. uniform distribution of the transported magnitude through a certain area), with the essential field u. Commonly, this relationship is formulated in tensor form as $\boldsymbol{q} = -\boldsymbol{K} \boldsymbol{\nabla} u$, where $\boldsymbol{K}$ is the thermal conductivity tensor, the diffusion tensor or the dielectric permittivity tensor, respectively, depending on the particular physical problem considered. For general nonlinear problems, the tensor $\boldsymbol{K}$ may be dependent (in a functional sense) on the field u as well as on the point $\boldsymbol{x}$ as any other field. It is common, however, to particularise this equation for linear (u-independent), homogeneous ($\boldsymbol{x}$-independent) and isotropic simplifications.

With these assumptions, Eq. (5.72) may be split in:

$$\boldsymbol{\nabla} \cdot \boldsymbol{q} = f \qquad \text{(fundamental principle)}, \tag{5.73a}$$

$$\boldsymbol{q} = -\boldsymbol{K} \boldsymbol{\nabla} u \quad \text{(constitutive equation)}, \tag{5.73b}$$

together with the appropriate boundary conditions.

Using the framework described above, Eq. (5.73a) is the universal law of the problem, and Eq. (5.73b) is the internal state equation. Using any discretisation technique, the values of u, $\boldsymbol{q}$ and $\boldsymbol{K}$ are replaced by the corresponding approximation functions, which are expressed in terms of the interpolating (nodal) values or by the approximation parameters, depending on the particular approach.

For instance, for one-dimensional problems, using forward first-order finite differences, the discretised version of Eq. (5.73) is:

$$\frac{q_{i+1} - q_i}{L_i} = f_{i+1}, \qquad i = 0, \dots, n, \tag{5.74a}$$

$$q_i = -k_i \frac{u_{i+1} - u_i}{L_i}, \quad i = 0, \dots, n, \tag{5.74b}$$

where L_i is an appropriate mesh size, u_i are the field variables, q_i are the internal state variables and the functional relationship $q = \mathcal{H}(u)$ is now expressed in the form of an algebraic equation $\boldsymbol{q} = \boldsymbol{H}(\boldsymbol{u})$, with $\boldsymbol{q} = (q_0, \dots, q_n)$, $\boldsymbol{u} = (u_0, \dots, u_n)$ and $\boldsymbol{f} = (f_1, \dots, f_n)$. Of course, when solving Eq. (5.73), or its corresponding discrete version Eq. (5.74), proper boundary conditions must be supplied. If the problem is now formulated within the PGNNIV framework, and for most cases, these boundary values, together with $\boldsymbol{f}$, are the natural inputs of the problem, being $\boldsymbol{u}$ the output. We have named this problem the prediction problem. In other cases, however, we are interested in characterising a given material from its response to different stimuli. In that case $\boldsymbol{g}$, $\boldsymbol{f}$ and $\boldsymbol{u}$ become the input variables while $\boldsymbol{K}$ is the output one in what we named the characterisation problem.

We focus now on predicting the fields $u(\boldsymbol{x})$, $q(\boldsymbol{x})$ and $k(\boldsymbol{x})$, for given values of the boundary conditions $\boldsymbol{g}$ and of the stimuli $\boldsymbol{f}$. This will be possible by using the approach stated in Section 5.2, such that the nodal values of $\boldsymbol{u}$ will be learned from a sufficiently big and varied data-set of input-output values, but constrained by the two following equations:

$$\frac{q_{i+1} - q_i}{L} = f_{i+1}, \qquad i = 0, \dots, n, \tag{5.75a}$$

$$q_i = -k_i(u_1, \dots, u_n) \frac{u_{i+1} - u_i}{L}, \quad i = 0, \dots, n, \tag{5.75b}$$

$$u_0 = g_1, \quad u_n = g_2. \tag{5.75c}$$

Eqs. (5.75) are formulated so that the internal state variables verify the fundamental principle of flow conservation, Eq. (5.75a), and the specific boundary conditions, Eq. (5.75c). At this level, the main problem relies on the form of the function $k_i = k_i(u_1, \dots, u_n)$, which is a multiple input-multiple output relationship that will be learned using DL regression techniques. Further assumptions can be made on the functional form of this relationship, which may be translated to the structure of the deep subnetwork associated with the constitutive equation. For example:

- It is possible to assume a local relationship of order m between q and u, that is, $k_i = F\left((\Delta^+)^m(u_i), (\Delta^+)^{m-1}(u_i), \dots, (\Delta^+)^0(u_i)\right)$, where Δ^+ is the forward difference operator and $(\Delta^+)^0(u_i) = u_i$. Furthermore, it is possible to extend the methodology for non-local operators (Ciaurri et al., 2018), with the inconvenience of additional numerical and computational complexity. In particular, a nonlinear relationship may be reduced to a separable form involving the field u, the gradient of the field u, and higher-order derivatives, or even non-local operators. Thus, $q_i = \Pi_{j=1}^k F_j\left((\Delta^+)^{r_j}(u_i)^{s_j}\right)$ where now F_j are the functions to be learned. This functional form is ubiquitous in mathematical (Osserman, 2013), physical (Frank,

2005; Ishizuka et al., 2008), engineering (Barenblatt et al., 1989; Caffarelli and Vazquez, 2010) and financial (Ankudinova and Ehrhardt, 2008; Barles and Soner, 1998) problems. As a very particular but common case, a pointwise relationship is expressed as $k_i = F_0\left((\Delta^+)^0(u_i)^1\right)$.

- It is also possible to assume a linear (possibly heterogeneous) relationship between the gradient of u and the flow q. In that case, $q_i = k_i\Delta^+ u_i$, that is, k_i are constants. The homogeneous case is a particular one, provided that $k_i = k, i = 1, \ldots, n$.

Finally, the function $k_i = k_i(u_1, \ldots, u_n)$ may be parametrised using model parameters $\boldsymbol{\lambda}$ with physical meaning. This approach recovers the classical parametric fitting when a given constitutive model structure is assumed, and is either useful for prediction problems and for model selection or validation.

For illustrative purposes let us consider the problem:

$$\frac{d}{dx}\left(k\frac{du}{dx}\right) = 0, \tag{5.76}$$

with boundary conditions:

$$u(x=0) = g_1, \quad u(x=1) = g_2. \tag{5.77}$$

Together with g_1 and g_2, $q_1 = q(x=0)$ and $q_2 = q(x=1)$ were considered as input variables to ensure that the PGNNIV is associated with a well-posed problem: we need at least one value of the flow, since the is flow constant and k undefined, meaning that the conservation equation is fulfilled for any flow distribution up to an additive constant value. With these considerations, (g_1, g_2, q_1, q_2) are the input variables and $\boldsymbol{u}$ is the output variable. The error term is $e^2 = \pi_0^2 = \int_0^1 (u(x) - \hat{u}(x))^2 dx$, that is, the L_2^2 norm error between the predicted and the measured fields, u and $\hat{u}$ respectively. The penalty functions are associated with flow conservation, essential boundary conditions and natural boundary conditions, respectively:

$$\pi_1^2 = \int_0^1 \left[\frac{d}{dx}\left(k\frac{du}{dx}\right)\right]^2 dx, \tag{5.78a}$$

$$\pi_2^2 = (u(x=0) - g_1)^2 + (u(x=1) - g_2)^2, \tag{5.78b}$$

$$\pi_3^2 = (q(x=0) - q_1)^2 + (q(x=1) - q_2)^2. \tag{5.78c}$$

To facilitate the discussion, we shall analyse separately the effect of including heterogeneity and nonlinearities, since each problem has its own particularities, even if both problems may be simultaneously studied in one stroke.

5.3.2.2 Homogeneous vs heterogeneous problem

In this case, we seek for solutions to the problem in Eq. (5.76) and boundary conditions in Eq. (5.77) when considering $k = k(x)$. Let us consider for such purposes two PGNNIVs whose topology and hyperparameters are presented next.

Predictive network

Let us denote the tensor input as $\mathtt{X}$ (shape $[N, 4]$), such that $\mathtt{X}[j,1] = g_1^j$, $\mathtt{X}[j,2] = g_2^j$, $\mathtt{X}[j,3] = q_1^j$ and $\mathtt{X}[j,4] = q_2^j$, while the tensor output is denoted as $\mathtt{u}$ (shape $[N, n]$) such that $\mathtt{u}[i,j] = u_i^j$. To predict the values u_i we use standard ANN regression techniques. In this work, we consider MLPs as the fundamental tool for building predictive networks, although more sophisticated ANNs may be suitable for other problems. In particular, we used a 4-layer network with two hidden layers, each with 15 neurons, such that $\mathtt{Y} = \mathsf{Y}[\mathtt{X}]$, with Y the nonlinear operator that identifies the input-output relation in the neural network. Up to this point, there is no novelty compared to a traditional ANN approach. With all these notations, the prediction error is:

$$\mathtt{e} = \mathtt{Y} - \mathtt{u}. \tag{5.79}$$

Continuum explanatory network

We establish now a tensor operator $\mathtt{D}^*$ associated with the first-order forward differential operator Δ^+ and we define $\mathtt{dY} = \mathtt{D}^*[\mathtt{Y}]$ (shape $[N, n-1]$). Then, we set a variable tensor, $\mathtt{K}$, of shape $[n-1, n-1]$. The structure of the mathematical equation implies that the tensor $\mathtt{K}$ is diagonal. Here we try two possibilities, resulting in two different PGNNIV: a scalar-tensor (N1) and a general diagonal tensor (N2). The first one is associated with a generic homogeneous problem and the second one with a heterogeneous one. Now we define $\mathtt{q} = -\mathtt{dY} \cdot \mathtt{K}$ (shape $[N, n-1]$) and we concatenate it with the two boundary flow values obtaining $\tilde{\mathtt{q}}$ (shape $[N, n+1]$). Finally, we define $\mathtt{f} = \mathtt{D}^*[\tilde{\mathtt{q}}]$, $\bar{\mathtt{u}} = (\mathtt{Y}[\cdot, 0] - g_1, \mathtt{Y}[\cdot, n] - g_2)$ and $\bar{\mathtt{q}} = (\mathtt{q}[\cdot, 0] - q_1, \mathtt{q}[\cdot, n-1] - q_2)$. Consequently, the penalty terms involved in the problem are:

$$\pi_1 = \mathtt{f}, \tag{5.80a}$$

$$\pi_2 = \bar{\mathtt{u}}, \tag{5.80b}$$

$$\pi_3 = \bar{\mathtt{q}}. \tag{5.80c}$$

Cost function, learning algorithm and optimization parameters

Besides the cost function and a learning algorithm and its associated hyperparameters as in the standard case, in the constrained ANN we also have to specify the weights associated with the PILs related to the constraints or, equivalently, the penalty coefficients that become, therefore, new hyperparameters. In this work, we consider the MSE as the cost function, both for the error and penalty terms, and we selected the Adam optimiser (Kingma and Ba, 2014). The resulting cost function (CF) (or loss function) of the optimisation procedure is, therefore:

$$\mathrm{CF} = c_0 \mathrm{MSE}(\mathtt{e}) + c_1 \mathrm{MSE}(\pi_1) + c_2 \mathrm{MSE}(\pi_2) + c_3 \mathrm{MSE}(\pi_3). \tag{5.81}$$

The optimisation parameters of the PGNNIV are summarised in Table 5.5.

To evaluate the network performance, we will consider the two cases $k(x) = 1$ (homogeneous problem, P1) and $k(x) = x + 1$ (heterogeneous problem, P2). The analytical solutions of these two problems are summarised in Table 5.6.

Parameter		Value
Learning rate		0.0003
Error coefficients		10^7
Penalty coefficients	Flow conservation	10^2
	Essential boundary conditions	10^3
	Natural boundary conditions	10^3

Table 5.5: PGNNIV optimisation parameters. We distinguish between the parameters corresponding to any ANN approach and those inherent to PGNNIV (the penalty coefficients).

Fields	$u(x)$	$q(x)$	$k(x)$
P1	$(g_2 - g_1)x + g_1$	$g_2 - g_1$	1
P2	$u(x) = \frac{g_2 - g_1}{\ln 2} \ln(x+1) + g_1$	$\frac{g_2 - g_1}{\ln 2}$	$x + 1$

Table 5.6: Analytical solutions associated with the two considered problems. The solution u, flow q and diffusivity k fields are shown.

N profiles of $u(x)$ were synthetically generated for different values of the boundary conditions, independently and uniformly sampled on $[0;1]^2$, that is $g_1, g_2 \sim \mathcal{U}[0;1]$ and independent. The values of the field u were particularised at $n = 10$ points, $x_i = (i-1)/(n-1), i = 1, \dots, n$ so the output variables correspond therefore to the nodal values $u_i = u(x_i), i = 1, \dots, 10$.

Network convergence

To evaluate the performance of the continuum-based PGNNIVs we generated $N = 10^4$ samples of input-output values for both P1 and P2 problems. We used 80% of the generated values as training data and 20% as test data. At each iteration throughout the optimisation process, the PGNNIV was fed with the whole training data-set. The process was stopped after 10^5 iterations. The convergence of both neural networks is shown in Fig. 5.26, where we plot the value of the cost function (including the error and penalty terms), demonstrating that a low value of the cost function was obtained in both cases for the homogeneous problem. Nonetheless, for this problem, the homogeneous network showed a faster convergence as it includes fewer learning parameters. However, only the heterogeneous network was able to reach similar low values for the heterogeneous problem.

The inability of the homogeneous network to reach low error predictions in the heterogeneous problem is explained by the impossibility of obtaining arbitrarily small values for the penalties associated with the flow conservation constraint for such a limited model. This is well illustrated in Fig. 5.27, which shows the value of the different penalty terms during the optimisation process. The penalty terms associated with the flow are unable to reach sufficiently low values because they are incompatible with the assumption of homogeneous material.

Finally, it is interesting to compare the convergence of the network in terms of the model network parameters. This is illustrated in Fig. 5.28. As for the homogeneous problem, all model network parameters converge to the theoretical value ($k = 1$),

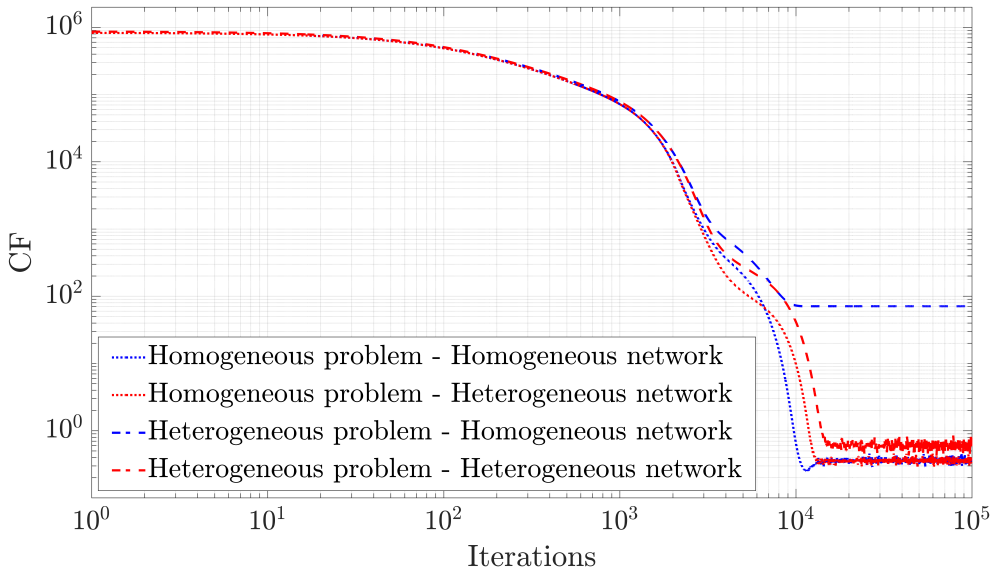


Figure 5.26: Learning curve for the different networks and data-sets. CF values are smoothed using a constant filter of bandwidth $w = 1000$ to avoid sharp oscillations in the logarithmic scale.

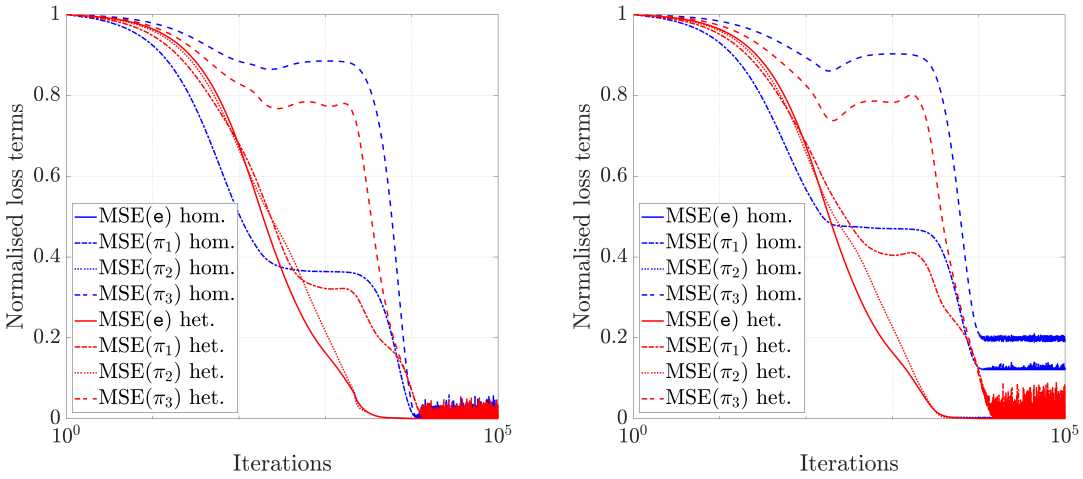
both for the homogeneous and heterogeneous network, albeit the homogeneous one converges faster. However, for the heterogeneous problem, each model parameter of the heterogeneous network model converges to a value satisfying the nodal constitutive relationship (that is, the nodal value k_i), while the homogeneous network does whatever is possible to reduce the CF, that is, the only model parameter converges to an intermediate value of the diffusivity k, so it never achieves the same predictive power as the heterogeneous network.

Predictive capacity

Once the network has converged, we can predict the values of the field u by simply interpolating the obtained values for the nodes, u_i, predicted by the network. Note that this prediction for the whole space has a minimal cost (the one of a single evaluation), as in any other neural network once trained, since it does not require the inversion of any system of equations or iterative procedure. Moreover, the values of the fields q and k are obtained as a byproduct of the network without any post-process beyond the nodal interpolation. As a simple illustration, Fig. 5.29 shows the neural network prediction of the essential and derivative fields for the heterogeneous problem for one particular set of boundary conditions ($g_1 = 0.31$, and $g_2 = 0.79$).

Even if we show the field associated with one single value of the inputs, the performance is general for the whole coverage of the input values. Indeed, the statistics of the normalised $\mathcal{L}_2$ errors corresponding to the prediction of the different fields are shown in Tables 5.7 and 5.8. This error is computed by using the estimate:

$$E_r^2[f] = \frac{\int_0^1 (\hat{f}(x) - f(x))^2 \, dx}{\int_0^1 f(x)^2 \, dx}, \tag{5.82}$$



(a) Homogeneous problem. **(b)** Heterogeneous problem.

Figure 5.27: Evolution of the loss term along the optimisation process. The different loss terms are normalised by their maximal value for a simpler and more consistent comparison. Curves associated with e and π_2 overlap as they are both related with the values of the solution field u.

where $\hat{f}$ is the predicted value of the field and f is the true value given by the analytical solution (see Table 5.6).

	$E_r^2[u]$					$E_r^2[q]$				
	min	Q_1	Q_2	Q_3	max	min	Q_1	Q_2	Q_3	max
N1	1×10^{-4}	1×10^{-4}	1×10^{-4}	2×10^{-4}	4×10^{-1}	2×10^{-4}	6×10^{-4}	1×10^{-3}	2×10^{-3}	1
N2	5×10^{-5}	1×10^{-4}	1×10^{-4}	1×10^{-4}	4×10^{-1}	5×10^{-4}	6×10^{-4}	7×10^{-4}	1×10^{-3}	1

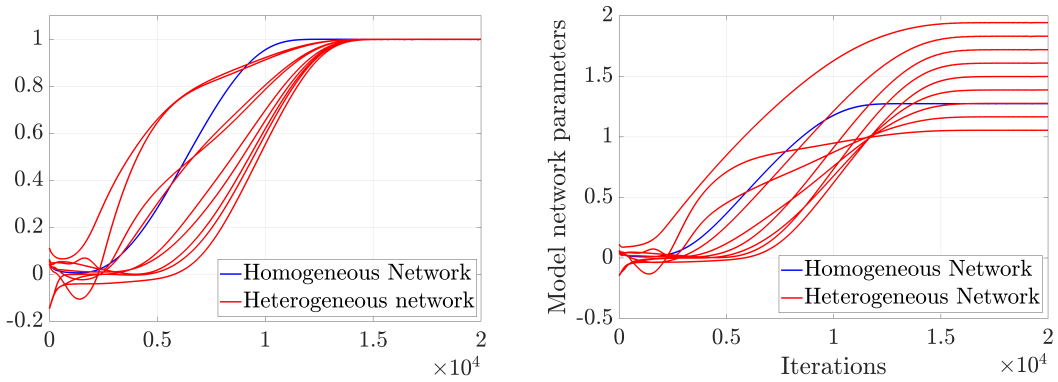
Table 5.7: Statistics of the error when predicting the output fields for the homogeneous problem. The two networks have an error of the same order of magnitude when solving the homogeneous problem.

	$E_r^2[u]$					$E_r^2[q]$				
	min	Q_1	Q_2	Q_3	max	min	Q_1	Q_2	Q_3	max
N1	7×10^{-6}	7×10^{-4}	1×10^{-3}	1×10^{-3}	4×10^{-1}	2×10^{-1}	2×10^{-1}	2×10^{-1}	2×10^{-1}	1
N2	1×10^{-4}	3×10^{-4}	6×10^{-4}	1×10^{-3}	4×10^{-1}	1×10^{-3}	2×10^{-3}	2×10^{-3}	3×10^{-3}	1

Table 5.8: Statistics of the error when predicting the output fields for the heterogeneous problem. The heterogeneous network outperforms the homogeneous network, as expected.

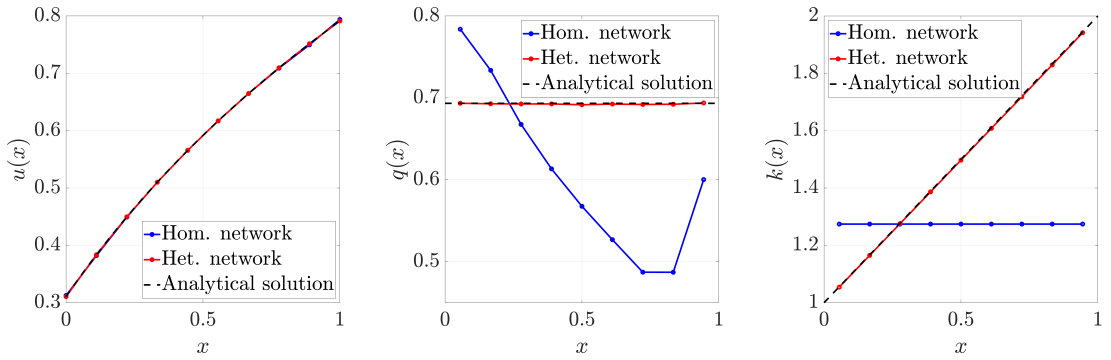
As the output field $k(x)$ does not depend on the value of the boundary conditions, all quantile indicators collapse to a single error value, as shown in Table 5.9.

Note that, except for some very particular predictions, the error remains small if the PGNNIV is able to learn the constitutive relation (less than 1% error for more than 75% of the predictions). Only the homogeneous network fails when estimating the values of the field q, which is the one associated with the constitutive model. Consequently, the PGNNIV is not capable either of learning accurately the value of k (error of the order 20%).



(a) Homogeneous problem. **(b)** Heterogeneous problem.

Figure 5.28: Network model parameter convergence. The nodal values of k are plotted until $M = 2 \times 10^4$ iterations, where the model network has converged.



(a) Field u. **(b)** Field q. **(c)** Field k.

Figure 5.29: PGNNIV prediction of the output fields. Both neural networks are able to predict the output field u for the heterogeneous problem but the heterogeneous network outperforms the homogeneous one when predicting the fields q and k.

In Figs. 5.30 and 5.31 the error obtained is depicted as a function of the boundary conditions for both neural networks and problems. As it may be seen, the error remains always small, independently of the values of g_1 and g_2, when estimating the field u for both problems and neural networks. The heterogeneous PGNNIV is additionally able to accurately estimate the field q, except for values close to the line $g_1 = g_2$, when of course $q(x) = 0$ and therefore $E_r^2[q] \to \infty$. Apart from this singular case, the error is generally higher when getting closer to the boundaries of the data-set coverage ($g_1 = 0, 1$, and/or $g_2 = 0, 1$), which is expected due to the self-learning nature of the method presented.

5.3.2.3 Linears vs nonlinear problem

With the aim of getting now a nonlinear version of the problem (5.76) and boundary conditions (5.77), we state $k = k(u)$. Let us consider now three PGNNIVs described next.

To evaluate the network performance, we shall consider three cases $k(u) = 1$ (constant diffusivity, P1), $k(u) = u$ (linear diffusivity, P2) and $k(u) = \exp(u)$ (exponential

		$E_r^2[k]$
Homogeneous problem	N1	7×10^{-6}
	N2	5×10^{-4}
Heterogeneous problem	N1	2×10^{-1}
	N2	7×10^{-3}

Table 5.9: Statistics of the error when predicting the diffusivity field for both problems. The heterogeneous network is the only one able to accurately predict the diffusivity field.

diffusivity, P3). The analytical solutions of these three problems are summarised again in Table 5.10.

Fields	$u(x)$	$q(x)$	$k(x)$
P1	$(g_2 - g_1)x + g_1$	$g_2 - g_1$	1
P3	$u(x) = \sqrt{(g_2^2 - g_1^2)x + g_1^2}$	$\frac{g_2^2 - g_1^2}{2}$	$\sqrt{(g_2^2 - g_1^2)x + g_1^2}$
P2	$u(x) = \ln\left((\exp(g_2) - \exp(g_1))x + \exp(g_1)\right)$	$\exp(g_2) - \exp(g_1)$	$(\exp(g_2) - \exp(g_1))x + \exp(g_2)$

Table 5.10: Analytical solutions associated with the problems considered. The solution u, flow q and diffusivity field k are shown.

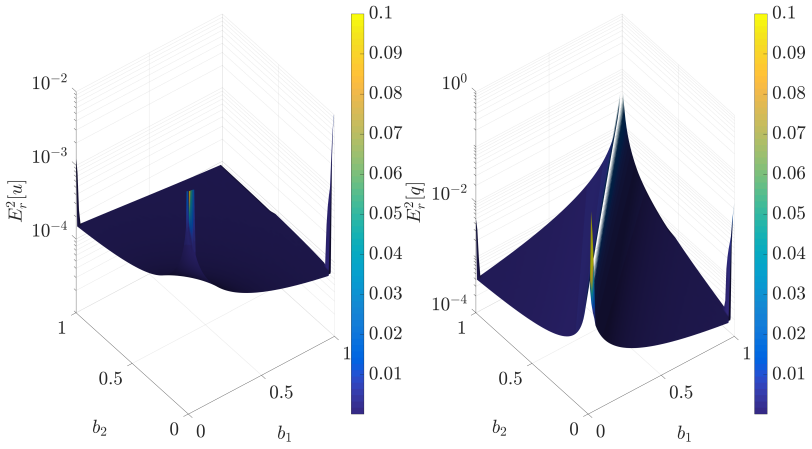
The input and output values that feed the neural network are generated analogously to the previous example.

Predictive network

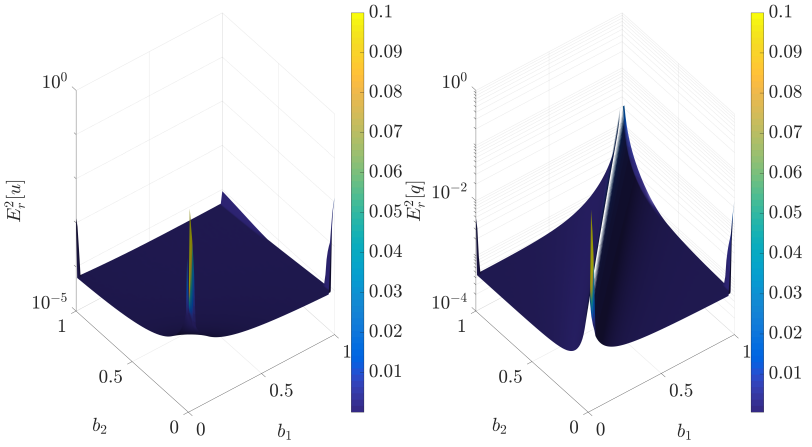
One of the advantages of this methodology is that the predictive neural network only depends on the nature of the input and output variables. Therefore, the predictive neural network for this nonlinear problem is exactly the same as in the previous example and so it is for the prediction error. The nature of the hidden state equation only affects the physical constraints associated with the PILs layers.

Continuum explanatory network

It is at this level where a supplementary effort has to be made. From the tensor $\mathtt{y}$, we obtain the derivative field using the tensorial operator $\mathtt{D}^*$, $\mathtt{dY} = \mathtt{D}^*[\mathtt{Y}]$ (shape $[N, n-1]$). However, we define additionally a new tensor $\mathtt{k}$ representing the diffusivity associated with each element. Note that this field has a shape of $[N, n-1]$, as it is defined in the elements rather than in the nodes, so we define an element field $\mathtt{um} = \mathtt{M}^*[\mathtt{u}]$ (shape $[N, n-1]$) obtained by averaging the nodal values of the field associated with the considered element. This operator is formulated in terms of a tensor operator, so a convolutional one-dimensional filter of size 2 with constant kernel $1/2$ was used. The next step is to define a neural network model relating the tensors $\mathtt{um}$ and $\mathtt{k}$. The point-wise character of this (unknown) relationship, that is, the fact that $\mathtt{k}[N, j] = \mathtt{F}(\mathtt{um}[N, j])$, is easily formulated in the deep learning TensorFlow framework by defining a CNN that expands in higher dimensional spaces the content of each neuron associated with each element.

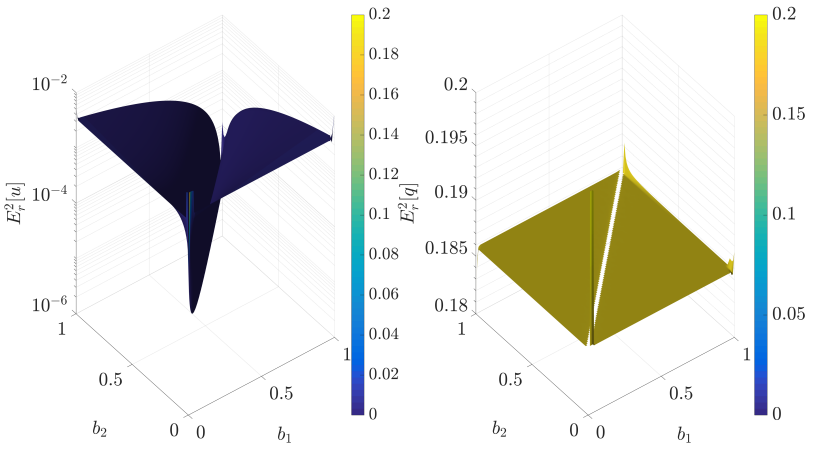


(a) Homogeneous network.

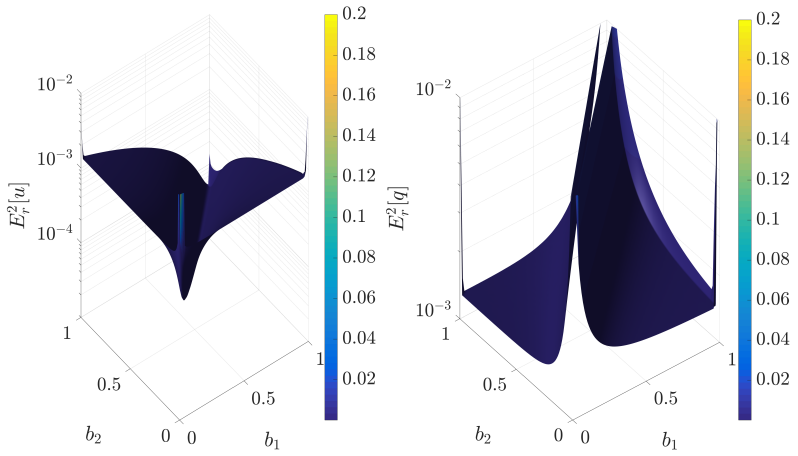


(b) Heterogeneous network.

Figure 5.30: Errors in the homogeneous problem for both neural networks. Both neural networks are able to predict accurately the fields u and q (less than 1% error) except for the case $g_1 = g_2$ when $q(x) = 0$.



(a) Homogeneous network.



(b) Heterogeneous network.

Figure 5.31: Errors in the heterogeneous problem for both neural networks. u is accurately predicted by the two neural networks, while only the heterogeneous neural network is able to predict accurately the field q (less than 1% error, except for the case $g_1 = g_2$ when $q(x) = 0$).

For illustrative purposes we tried two possibilities, resulting in two different PGNNIVs: a 2-layer CNN (no hidden layers) and a 3-layer CNN. For the first, no activation function was used so the CNN is able to reproduce only linear relationships. For the second one, we used sigmoid activation functions in the hidden layer. In all layers, we considered bias terms before and after applying the activation function. Another possibility, which was explored alternatively, was to prescribe a parametric relationship between the two tensors `um` and `k`. Here we illustrate this possibility by prescribing $k(u) = \alpha + \beta u^{\gamma}$ where α, β and γ are model parameters.

As a summary, the tensorial flow at the model network is, for the three proposed approaches:

$$\underbrace{\texttt{um}}_{[N,n-1]}\ \underbrace{\rightarrow}_{[1,n-1,n-1]}\ \underbrace{\texttt{k}}_{[N,n-1]}$$

$$\underbrace{\texttt{um}}_{[N,n-1]}\ \underbrace{\rightarrow}_{[1,m,n-1,n-1]}\ \underbrace{\texttt{h}}_{[N,m,n-1]}\ \underbrace{\rightarrow}_{[m,1,n-1,n-1]}\ \underbrace{\texttt{k}}_{[N,n-1]}$$

$$\underbrace{\texttt{um}}_{[N,n-1]}\ \underbrace{\rightarrow}_{[1,n-1,n-1]}\ \underbrace{\texttt{k}}_{[N,n-1]}$$

where we indicate under each tensor or operation, the shape of the tensor or filter. The penalty terms related to the PGNNIV are the same used in the previous example. Note that with respect to the previous approach, there is an extra hyperparameter related to the network topology, which is the size of the convolutional filter in the second network, that is, m. For the following results obtained, we set a value of $m = 5$. The impact of this parameter is analysed in Section 5.3.2.4.

Cost function, learning algorithm and hyperparameters

We used the same cost function, learning parameters and hyperparameters than in the preceding example (see Table 5.5).

Network convergence

Convergence in terms of the CF and the different penalties presents the same trend discussed before when analysing Fig. 5.26. However, it is interesting here to show the evolution of the model parameters during the optimisation problem, which is informative about the nature of the model $k = k(u)$ (Fig. 5.32).

First, it is easy to observe that the parameters of both the 2L-CNN model and the parametric CNN are easily interpreted in physical terms. Indeed, for the problem $k(u) = 1$, the weight of the 2L-CNN converges to 0 and the bias converges to 1 because $k(u) = 0 \cdot u + 1$. This also happens in the parametric network, where $\alpha \to 1$ and $\beta \to 0$ although there is a third spurious parameter, γ, that remains undetermined and therefore its convergence is not guaranteed. The parameters associated with the 3L-CNN, although not so easily interpretable, do converge, which is indicative of the fact that the CNN network is able to find an optimal solution, regardless of whether this solution is accurate or not.

For the linear diffusivity case, the interpretation is similar: the weight of the 2L-CNN converges to 1 and the bias converges to 0, because $k(u) = 1 \cdot u + 0$, while for

the parametric network, $\alpha \to 0.06$, $\beta \to 0.97$ and $\gamma \to 1.2$. Note that even if the underlying model is learned well enough, the numerical error intrinsic to the network induces another error in the parameter estimation, which darkens the linear $k-u$ relationship. This may be dramatic when extrapolating, being this a well-known drawback when using complex parametric models without paying attention to overfitting. The 3L-CNN did not totally converge after $M = 10^5$ iterations even if, as we will see later, yields good enough results. This is possibly due to an excess of network parameters (excess of neurons) and/or to an insufficiently good model learning approach that could be improved with longer runs or using different optimisation algorithms (from the mathematical point of view, the problem is not bounded or the search algorithm has not reached a local minimum).

Finally, in the exponential diffusivity problem, the 2L-CNN network and the parametric network reach convergence close to $M = 3 \times 10^4$ iterations, while for the parametric network we get $\alpha = 1.06$, $\beta = 1.60$ and $\gamma = 1.47$. This may be interpreted as the optimal least-squares solution for the parametric problem, when using the CF considered. This solution may be also obtained by using another optimisation approach different from backpropagation (for instance, the Levenberg-Marquardt algorithm (Levenberg, 1944) commonly used in parametric fitting). This latter family of algorithms is usually resource-intensive and complex to use in large-scale problems. The behaviour of the 3L-CNN performance is, in that case, similar to the one for the linear diffusivity problem.

Predictive capacity

As in the previous example (homogeneous vs heterogeneous), the performance of the presented methodology in predicting the value of the different fields is now evaluated. There is, however, a difference with respect to that example that is inherent to nonlinear problems: the value of the field $k(x)$ depends now on the boundary conditions, as $k = k(u(x))$ and u depends on the boundary conditions. Fig. 5.33 shows the neural network prediction of all fields involved in the problem in the hardest case analysed, that is, exponential diffusivity, for one particular set of boundary conditions ($g_1 = 0.31$ and $g_2 = 0.79$).

More important than a particular prediction for one set of boundary conditions is the statistics of the errors for the boundary conditions varying in the whole learning space. This statistics is shown in Tables 5.11, 5.12 and 5.13 for the fields u and q and the three problems, respectively, while the one for the diffusion field k is presented in Table 5.14.

	$E_r^2[u]$					$E_r^2[q]$				
	min	Q_1	Q_2	Q_3	max	min	Q_1	Q_2	Q_3	max
2L-CNN	4×10^{-7}	1×10^{-6}	2×10^{-6}	4×10^{-6}	1×10^{-1}	3×10^{-5}	3×10^{-5}	5×10^{-5}	7×10^{-5}	3×10^{-1}
3L-CNN	9×10^{-7}	5×10^{-6}	9×10^{-6}	2×10^{-5}	2×10^{-1}	4×10^{-5}	8×10^{-5}	1×10^{-4}	2×10^{-4}	4×10^{-1}
Parametric	3×10^{-5}	3×10^{-5}	4×10^{-5}	4×10^{-5}	1×10^{-1}	1×10^{-4}	2×10^{-4}	4×10^{-4}	8×10^{-4}	4×10^{-1}

Table 5.11: Statistics of the error when predicting the output fields for the problem with constant diffusivity. We obtain similar statistical results for all the neural networks.

	$E_r^2[u]$					$E_r^2[q]$				
	min	Q_1	Q_2	Q_3	max	min	Q_1	Q_2	Q_3	max
2L-CNN	3×10^{-4}	3×10^{-3}	5×10^{-3}	8×10^{-3}	3×10^{-1}	5×10^{-3}	2×10^{-2}	3×10^{-2}	4×10^{-2}	2×10^{1}
3L-CNN	2×10^{-4}	3×10^{-3}	5×10^{-3}	8×10^{-3}	3×10^{-1}	4×10^{-3}	2×10^{-3}	3×10^{-3}	5×10^{-2}	2×10^{1}
Parametric	3×10^{-4}	2×10^{-3}	4×10^{-3}	6×10^{-3}	9×10^{-2}	2×10^{-3}	1×10^{-2}	2×10^{-2}	3×10^{-2}	2×10^{1}

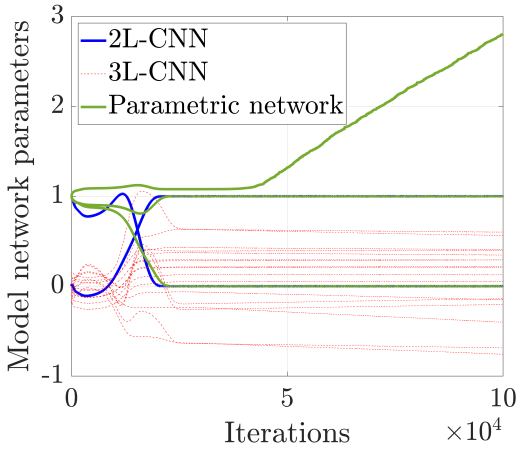
Table 5.12: Statistics of the error when predicting the output fields for the problem with linear diffusivity. We obtain similar statistical results for all the neural networks.

	$E_r^2[u]$					$E_r^2[q]$				
	min	Q_1	Q_2	Q_3	max	min	Q_1	Q_2	Q_3	max
2L-CNN	1×10^{-4}	2×10^{-3}	3×10^{-3}	5×10^{-3}	1×10^{-1}	3×10^{-3}	2×10^{-2}	3×10^{-2}	4×10^{-2}	7×10^{-1}
3L-CNN	1×10^{-4}	2×10^{-3}	3×10^{-3}	5×10^{-3}	2×10^{-1}	4×10^{-3}	2×10^{-2}	3×10^{-2}	5×10^{-2}	7×10^{-1}
Parametric	1×10^{-4}	2×10^{-3}	4×10^{-3}	5×10^{-3}	2×10^{-1}	2×10^{-3}	9×10^{-3}	2×10^{-2}	3×10^{-2}	8×10^{-1}

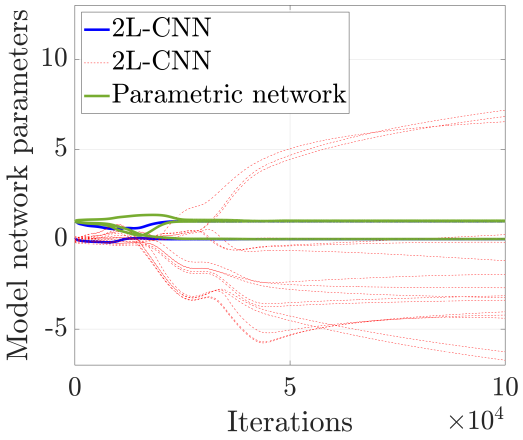
Table 5.13: Statistics of the error when predicting the output fields for the problem with exponential diffusivity. We obtain similar statistical results for all the neural networks.

		$E_r^2[k]$				
		min	Q_1	Q_2	Q_3	max
Constant diffusivity	2L-CNN	3×10^{-4}	3×10^{-4}	3×10^{-4}	4×10^{-4}	4×10^{-4}
	3L-CNN	7×10^{-7}	4×10^{-5}	5×10^{-5}	5×10^{-5}	2×10^{-3}
	Parametric	2×10^{-7}	1×10^{-5}	2×10^{-5}	2×10^{-5}	2×10^{-5}
Linear diffusivity	2L-CNN	4×10^{-3}	7×10^{-3}	1×10^{-2}	2×10^{-2}	∞
	3L-CNN	1×10^{-3}	9×10^{-3}	1×10^{-2}	2×10^{-2}	∞
	Parametric	1×10^{-3}	6×10^{-3}	1×10^{-2}	2×10^{-2}	∞
Exponential diffusivity	2L-CNN	6×10^{-4}	2×10^{-2}	3×10^{-2}	3×10^{-2}	8×10^{-2}
	3L-CNN	6×10^{-4}	1×10^{-2}	2×10^{-2}	5×10^{-2}	2×10^{-1}
	Parametric	6×10^{-4}	4×10^{-3}	7×10^{-3}	1×10^{-2}	7×10^{-2}

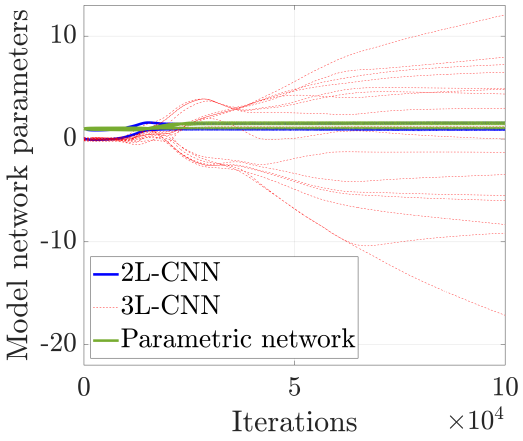
Table 5.14: Statistics of the error when predicting the diffusivity field for the three problems. We obtain similar statistical results for all the neural networks.



(a) Constant diffusivity.



(b) Linear diffusivity.

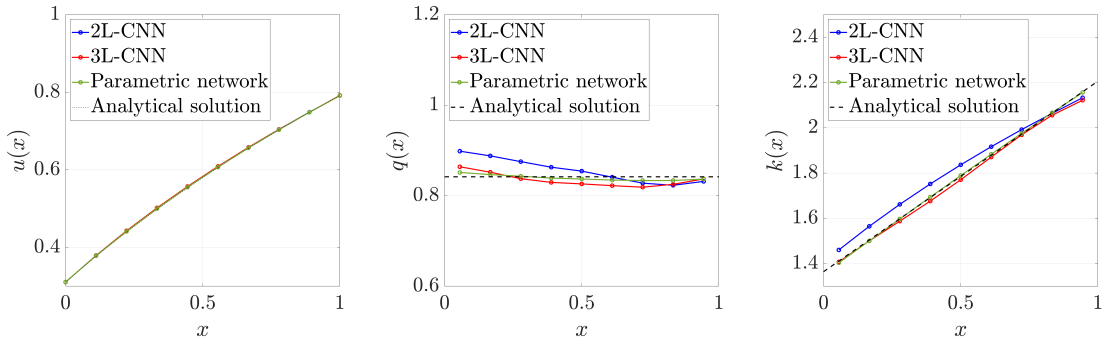


(c) Exponential diffusivity.

Figure 5.32: Network model parameter convergence. The values of all the network parameters are displayed, but only those represented in continuous line have a direct interpretation in terms of parameters in the relation $k = k(u)$.

In addition to the error associated with the fields u and q, it is possible to evaluate the error of the field k for the different boundary conditions. Fig. 5.34 shows the relative $\mathcal{L}_2$ error for the three tested networks and the three data-sets.

From the figures and tables presented, we can draw several important observations. For the first problem, the three PGNNIVs have a good accuracy, though the 2L-CNN, whose learning power is specific for linear models, provides the best results in terms of errors. Besides, as the problem is linear, the error has a linear shape when visualised in terms of the boundary conditions. The 3L-CNN and the parametric PGNNIV are nonlinear models and so it is the error. For the second problem, the three PGNNIVs estimate accurately the value of the field k except for some values close to the boundaries of the sampling space. Therefore, the three networks are useful as model learners. Finally, for the third problem, we observe that, although the error is low, in general, for the three models, the 3L-CNN and the parametric network achieve smaller values. That is because the first network has less predictive power than the second for a general class of functions, while, for that case, a parametric model



(a) Field u. **(b)** Field q. **(c)** Field k.

Figure 5.33: PGNNIV prediction of the output fields. The three neural networks are able to predict the output field u for the exponential diffusivity problem but the predictive capacity varies from one network to another for q and k.

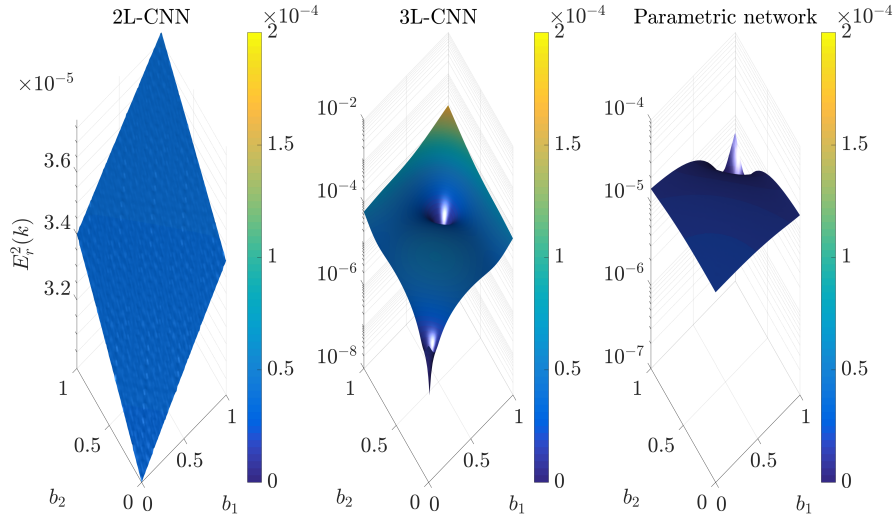
of the form $k(u) = \alpha + \beta u^{\gamma}$ is able to describe the underlying physics accurately enough. Note that this, however, is particular to the problem in hands, that is, $k(u) = \exp(u)$. For all cases, the highest errors always appear close to the boundaries of the learning domain. If we compare only the second and third PGNNIV, the third one reaches better predictions close to the boundaries. This is a consequence of the multi-parametric nature of neural networks when used in regression problems: certain overfitting is unavoidable to some extent and is glimpsed especially close to the boundary values.

Explanatory capacity

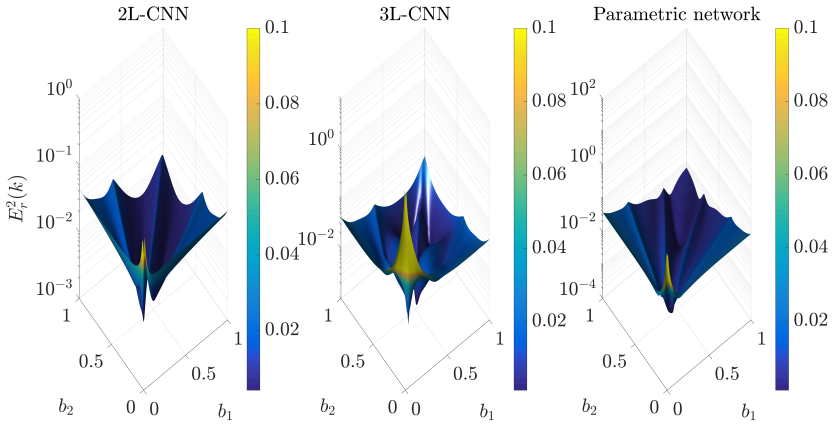
As explained in Section 5.2, PGNNIVs have both predictive and explanatory capacity, in the sense that they can unravel the hidden internal state equations. This has been explored in the precedent example by reproducing the field $k = k(x)$. This field is a direct output of the problem, but when expressed in terms of the variable x may be seen as an explanation (identification) of the heterogeneous constitutive model. For nonlinear problems, however, the problem becomes richer. The interest here is to learn the model $k = k(u)$. One way is to sample the two output fields $k(x)$ and $u(x)$ for each x and all boundary conditions. This leads however to a point cloud due to the noise and discretisation errors. But there is one more elegant alternative that consists in exporting the network related to the model. Its convolutional nature makes it independent of the considered point x (that is, element independent). Fig. 5.35 shows the model predictions for the three data-sets and the three tested PGNNIV. It is important to emphasise once again that the parametric network has a good learning capacity since $1.06 + 1.60u^{1.47}$ is a good enough approximation of $\exp(u)$, although this characteristic is specific to the problem in hands.

5.3.2.4 Numerical experiments

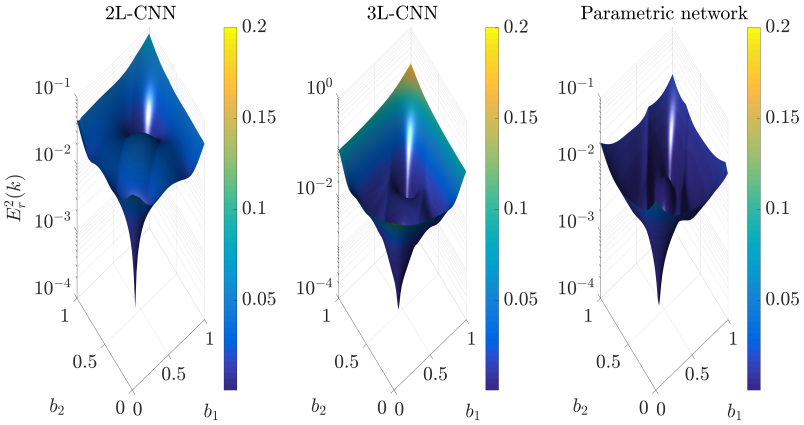
Next, we evaluate the performance of the methodology presented according to different parameters, inherent to the presented methodology: the data-set size, the error in the training data-set and the size of the hidden layers. The following discussion corresponds to the more complex exponential diffusivity problem using the 3L-CNN network.



(a) Constant diffusivity.

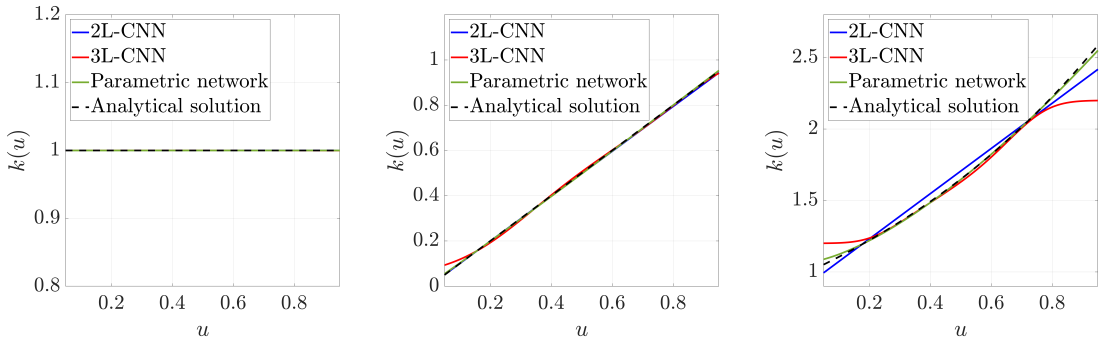


(b) Linear diffusivity.



(c) Exponential diffusivity.

Figure 5.34: PGNNIV error in predicting the field k. All networks are able to estimate the value of the field k for the constant and linear diffusivity problems. Although the 2L-CNN network yields small enough errors for the exponential case, the 3L-CNN and the parametric networks provide more accurate results.



(a) Constant diffusivity. **(b)** Linear diffusivity. **(c)** Exponential diffusivity.

Figure 5.35: Unravelling capacity of the PGNNIVs. The three presented PGNNIVs try to explain the constitutive model $k = k(u)$. Data-set coverage leads to sampled values of `um` in $[0.13, 0.87]$ which explains the differences obtained beyond those values.

Data-set size

Fig. 5.36 shows the E_r^2 errors when varying the data-set size, both considering the predictive and unravelling capacity of the network. The main conclusion is that the data-set size has an important impact on accuracy and precision, but not much on the model learning capacity. This result is expected since for each sample of the data-set, the model learning is performed at the nodal level, so the learning capacity is amplified as a consequence of the discretisation. This is even more evident when analysing the spatial error defined for a spatial field f as:

$$\varepsilon_r[f](x) = \left| \frac{\hat{f}(x) - f(x)}{f(x)} \right| \tag{5.83}$$

This is illustrated in Fig. 5.37: even if for all fields, a larger data-set implies better estimations, the field k demands fewer data to be learned.

Noise impact

To analyse the impact of the data noise on the results we added a white noise to the training data. The noise level is evaluated by introducing a white noise, proportional to the standard deviation of spatial profile. That is, $u_i^n \sim \mathcal{N}(u_i, s)$ where u_i is the noise-free nodal values, u_i^n is their noisy counterpart, and $s = p\sigma$, being σ the standard deviation of $\{u_i\}_{i=1,\dots,n}$. First, we analyse the noise impact on the network convergence. Fig. 5.38 shows the CF evolution as well as the different penalty terms during the optimisation process. As the noise affects directly the output field u, the differences in the convergence of the CF function are associated mainly with the prediction error term associated with u, already in the 1000^{th} iteration. However, once the output field noise has been filtered by the PGNNIV, there is another error source associated with the numerical discretisation, which plays an important role in the last stage of convergence, around $M = 2 \times 10^4$.

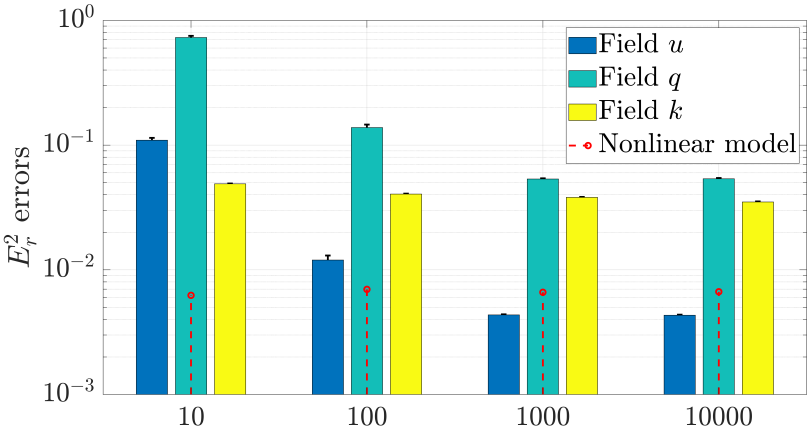


Figure 5.36: Impact of the data-set size on the predictive and unravelling capacity of the network. The relative E_r^2 is shown for different data-set sizes. As the prediction of the fields depends on the boundary conditions, the error is shown with its associated standard error bar.

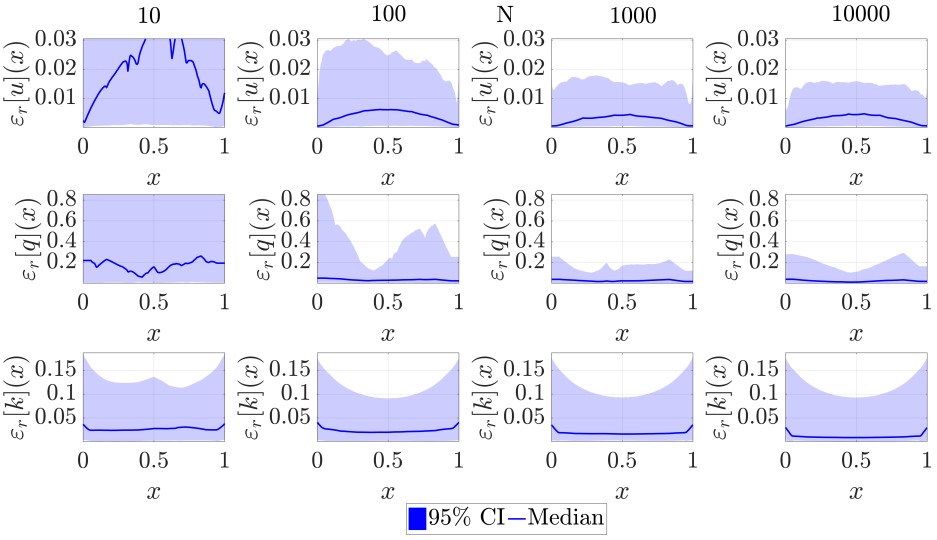


Figure 5.37: Impact of the data-set size on the spatial field prediction. As the different fields involved depend on the boundary conditions, the median of the relative error is shown together with a 95% confidence interval.

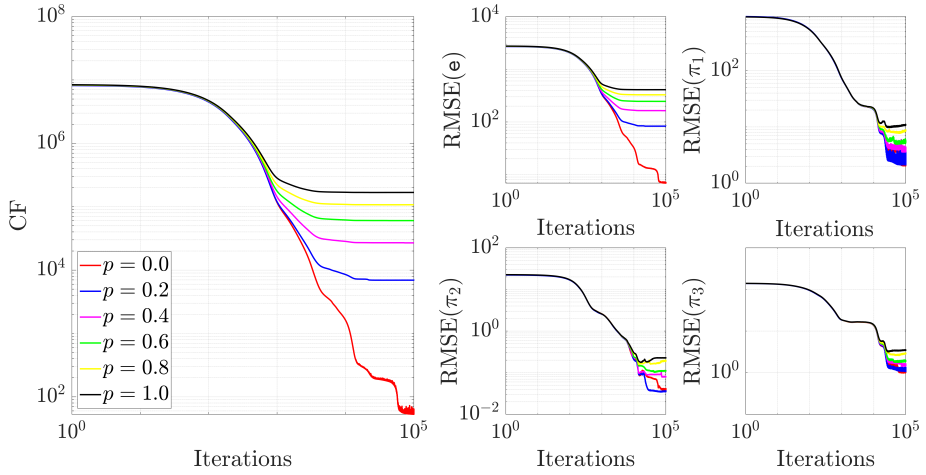


Figure 5.38: Impact of the noise level on the network convergence. The CF plotted at the left panel is the squared sum of the terms shown at the right panel.

We analyse now the impact of the noise in the E_r^2 errors, both for predictive and unravelling capacity. The results are shown in Fig. 5.39, getting a similar interpretation to the one for the data-set impact. However, there is one subtlety: the fields k and q are associated with the derivatives of the field u so, even if the filtering capacity of the network is remarkable for the prediction of all fields, as it is common when using PGNNIV methodology, the error in the prediction is higher for the fields involving a derivative of the discretised function, as the error is amplified by numerical discretisation. Fig. 5.37 shows the spatial errors, illustrating this fact more clearly.

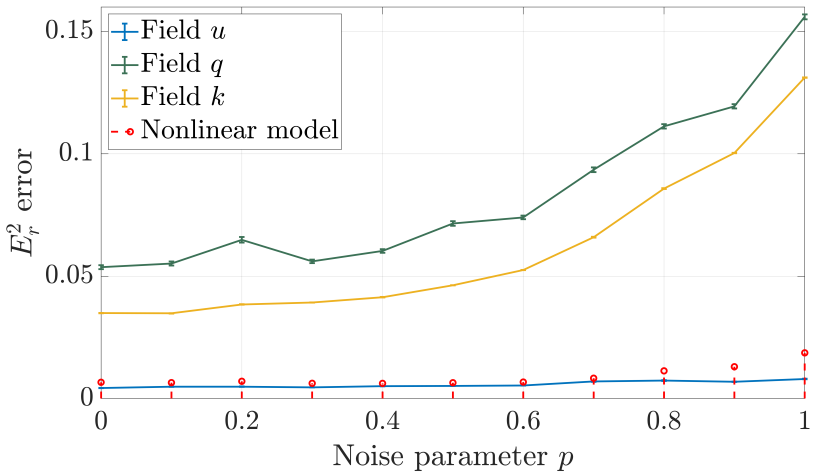


Figure 5.39: Impact of the noise level on the predictive and unravelling capacity of the network. The relative E_r^2 is shown for different data-set sizes. As the prediction of the fields depends on the boundary conditions, the error is shown with its associated standard error bar.

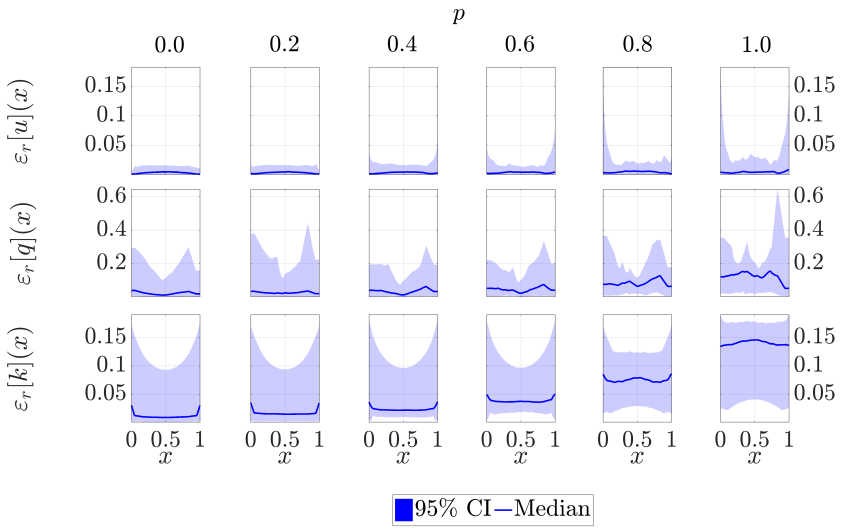


Figure 5.40: Impact of the noise level on the spatial field prediction. As the different fields involved depend on boundary conditions, the median of the relative error together with a 95% confidence interval is shown.

Effect of discretisation

Now, we explore the effect of increasing the discretisation (that is, to augment the number of nodes or to reduce the mesh size). This is implemented by increasing the number of output values u_i and, consequently, the number of fluxes q_i. the results are shown in Fig. 5.41. As it can be seen, a rough increase of the discretisation does not necessarily improve the prediction or explanatory capacity. This is a fundamental difference between this methodology and the usual simulation approach (that is merely predictive). The convergence with the mesh is a very critical aspect. Here we point out the need of new research results in this line. In a certain sense, we still suffer from the lack of mathematical results, playing the role that, for instance, convergence theorems play in FE analysis.

Learning space size

Finally, we evaluate the effect of varying the learning space associated with the model. The first comment corresponds to the convergence of the network, which is shown in Fig. 5.42: the larger the learning space (i.e. the higher the number of neurons in the network), the higher the converge cost.

The predictive and unravelling capacity of the network is shown in Fig. 5.43. As it is common in the neural network framework, the learning power of the model increases with the model parameters (number of neurons) until it reaches a stagnation point, beyond which the accuracy does not improve. It is important, however, to remark that an augmentation of the learning space is always related with an average accuracy improvement, but not necessarily with a precision improvement for all specific problems. Indeed, if the learning space is large, the model risks being overfitted, resulting in poor predictions for some special cases. This may be seen when comparing the error bar for the different predictions. To illustrate this fact, Fig. 5.44 plots the spatial distributions of the errors of fields q and k (the ones involved in

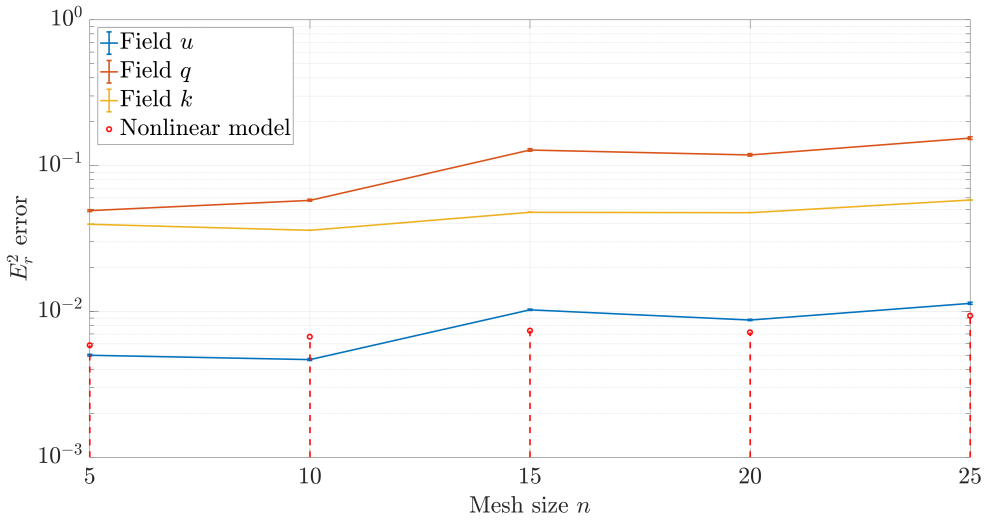


Figure 5.41: Effect of the discretisation. Note that a mesh refinement is not necessarily accompanied by an improvement in the results.

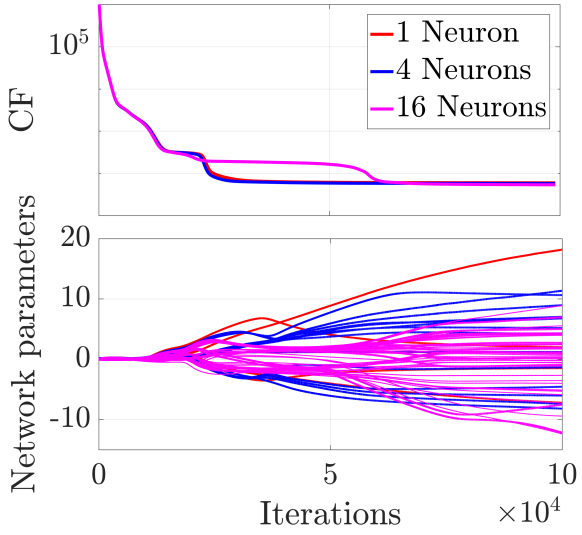
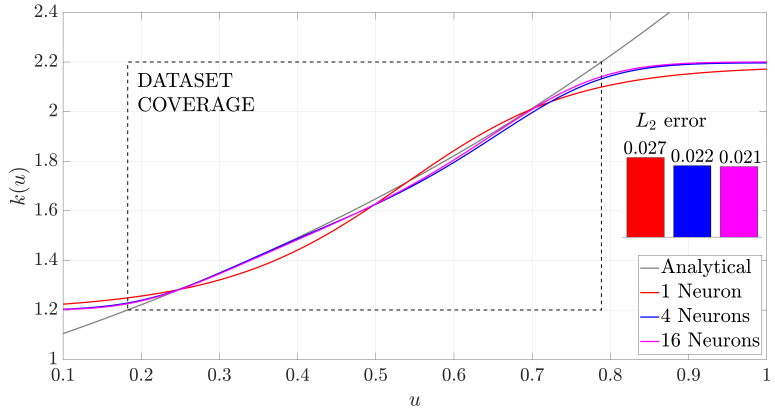
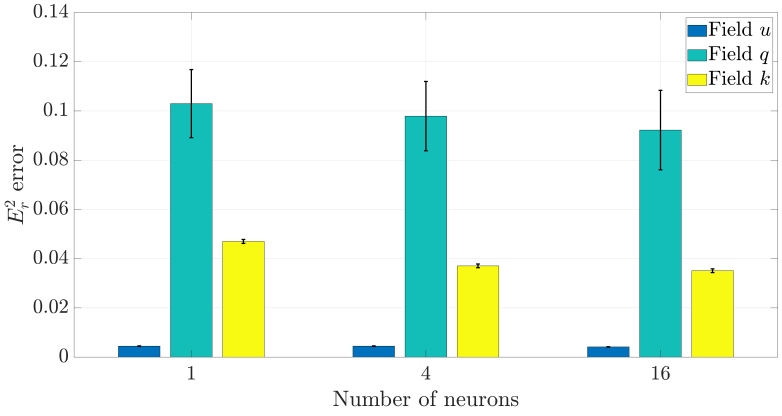


Figure 5.42: Impact of the learning space size on the network convergence. The upper panel shows the value of the CF and the lower panel the value of the weights at each iteration. CF values are smoothed using a constant filter of bandwidth $w = 1000$ for avoiding sharp oscillations in the visualization in the logarithmic scale. Note that a sharp change in the value of the parameters is associated with a significant change in the value of CF.

the model learning). The results are in agreement with our conclusion: lower values of median estimations and larger confidence intervals.



(a) Predictive capacity.



(b) Unravelling capacity.

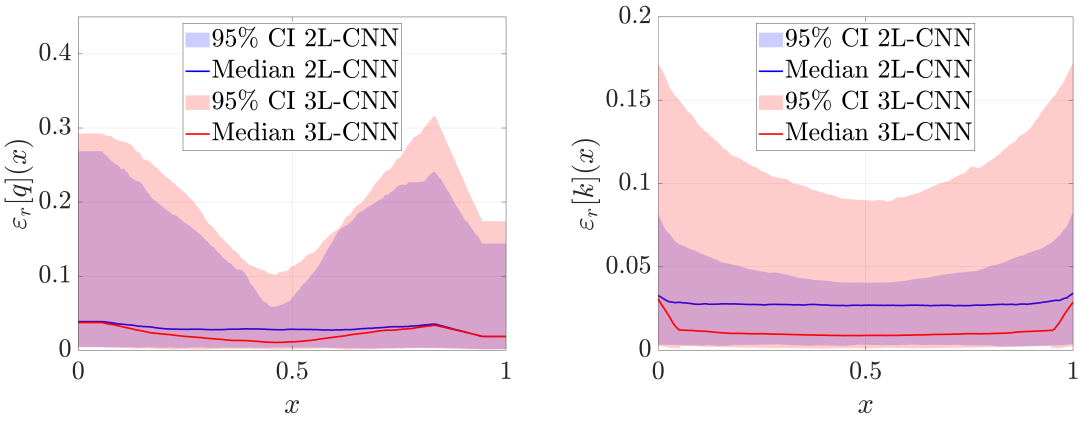
Figure 5.43: Predictive and unravelling capacity of the PGNNIV. The unravelling capacity is evaluated using the $\mathcal{L}_2$ error instead of the relative $\mathcal{L}_2$ error since it is not reported in the same plot.

In summary, an increase in the number of neurons is a good starting point for reaching good enough results, but it is a rude strategy when the goal is to fit the model precisely and avoid overfitting. For these purposes, better and more suited strategies are required (Hou et al., 2017).

5.4 Discussion

The presented framework introduces a new and singular way of combining physical knowledge and the power of the most recent data science techniques to solve physical problems. Although the presented illustrative problems are simple and academic, they highlight all the ingredients and the main features of the methodology. We summarise the following achievements:

- **The capacity of dealing with arbitrary complex models, equations and structures.** We have considered nonlinearities and different degrees of material



(a) Field q. **(b)** Field k.

Figure 5.44: Spatial distribution of the error for q and k fields and 2L-CNN and 3L-CNN. As the different fields involved depend on boundary conditions, the median of the relative error is shown together with a 95% confidence interval.

knowledge such as spatial symmetry. Also, locality has been implicitly considered when establishing a local relation between the flow and the essential field.

- **Flexibility to add some or the whole available physical knowledge to the network.** Several degrees of knowledge have been tested, involving symmetries, the mathematical character of the constitutive operators and their explicit parametric dependence. Also, selecting a model from among several candidates is another possibility.
- **The two-stage character of the methodology.** In the first step, the predictive and explanatory (or unravelling) capacities of the methods are clearly revealed. This process is computationally expensive. However, nowadays, there are a lot of resources to perform this task, such as scalability and parallelisation tools, cloud and distributed computing and adapted hardware technologies such as GPUs and, TPUs and field programmable gate arrays (FPGAs). The second step is a pure evaluation of the output for the desired input. Primary and derived fields and parameters are obtained in one evaluation cost. Only, the post process is related to the interpolation of the nodal values to get the different fields in the whole domain. As an example, Fig. 5.44 was generated using a 100×100 grid, so it includes the resolution of 10^4 nonlinear PDEs. It was, however, generated in less than one second using a personal laptop as it involves only 10^4 network evaluations for each model.

Nonetheless, there is an important limitation that arises from the present study. It is crucial to correctly choose the best strategy when selecting what is known and what is not in the problem to solve. If something is known, the best strategy is to include it in the PGNNIV, explicitly if possible, or implicitly if not. In the limit case, parametric models are the best ones and less expensive to train, so they are the ones that provide the best results, when correctly assumed as it was illustrated with the hydraulics example (see Fig. 5.14) and the homogeneous heat equation (see Tables

5.7, 5.8 and 5.9). Indeed, centuries dedicated to establish models cannot be wrong. This may drive to think that in that limit case nothing new has been presented, but this is simplistic since the model learning and the predictive capacity are acquired in one stroke and once for all and can be continuously improved by new sets of data input, using dynamic data-driven application systems (DDDASs). It is, however, important to note that obtaining high quality data may be more complicated than initially thought, since there is not a simple way to guarantee that internal variables have a pertinent coverage (see, for instance, Fig. 5.43) when the constitutive relation is not *a priori* known).

In any case, once the model is trained, any prediction may be performed later via offline calculation in one single evaluation. This is extremely advantageous for optimisation, inverse problems or stochastic computations based on Montecarlo strategies, among others. In particular, we may establish the whole constitutive model, which is equivalent to build a response surface using ANN procedures or to derive a reduced order model (ROM) using backpropagation, with the particularity that the computational cost is reduced with respect to usual ANN procedures when adding the physical constraints as explained when analysing the performance of PGNNIV.

5.5 Conclusions

We have presented a framework in which we use the technology and methods of ANN for solving physically-based problems, including both discrete and continuum ones. The key point is a cunning split of the equations of the problem into two sets, what is known and what is not, and therefore you want it to be learned. The result is a method that offers both predictive and explanatory capacity: it is possible to predict, in a single evaluation, the state of the system for any prescribed value of the input variable, and to learn about the mathematical structure of the constitutive state equation of the problem in hands. Indeed, good results have been demonstrated for some paradigmatic cases in science and engineering related to constitutive modelling: heterogeneity, nonlinearity, local character, model identification and feature detection.

This approach allows us both to predict the evolution of a physical system and to explain its structure in the language of Physics. Moreover, we have taken advantage of the characteristics of the ANN language for a very general formulation of a broad class of problems whose mathematical model is expressed by means of a system of PDEs.

Of course, it suffers from the typical pros and cons of neural networks. As pros:

- Once the PGNNIV is trained, it allows us to make predictions in an evaluation cost (that is, in real-time). No linear operator inversion, tangent-based operators computation, or iterative procedure are necessary when predicting the state or the evolution of a system.
- The network is trained offline in a very time-consuming process. However, ANN is a mature and hot field in continuous development, and specific hardware and software tools (parallel, cloud and distributed computation, GPUs and TPUs software, mathematical optimisation algorithms, among other software and hardware solutions) are accelerating more and more the training steps.

As main cons, we can mention the following:

- For DL models, the data used in the training process must be large and varied. This explains why topics such as the internet of things (IoT) and the Big Data paradigm are becoming so important in this context. Data quantity but also data quality are required: large amounts of data are not enough, but they have also to be well distributed and uncorrelated for the model to be accurately learned. This is more complicated to be thought, since there is not a simple way to guarantee that internal variables have a pertinent coverage.
- Defining successful neural network models requires a complex and time-consuming process of network topology definition and hyperparameter tuning. Although some efforts have been made in the last years for developing appropriate tools Bardenet et al. (2013); J. Toal et al. (2008); Maclaurin et al. (2015); Snoek et al. (2012), this remains a hard task that strongly depends on the researcher's knowledge, experience or intuition.

- Even if the ANN converges, it is difficult to expect a prediction to be as accurate as when using fully prescribed mathematical models. PGNNIV are therefore strongly recommended for problems where qualitative explanations or major trends are searched, without entering in fine quantitative details.

In addition to these general characteristics of ANN methods, the presented hybrid formulation has shown extra advantages with respect to other existent methods:

- It allows working only with measurable variables. This is crucial as all the DD approaches, such as the presented in Chapter 3 and Chapter 4, in one way or another, make hypotheses and assumptions about the relationships between measurable and non-measurable (internal) variables.
- The presented method ranges from model-free to model-based approaches. In this sense, we talk about model-guided methods.
- As this methodology is physically guided, it allows the explainability of the different phenomena investigated, so it can be framed within the scope of XAI Došilović et al. (2018); Samek et al. (2017).
- It allows obtaining the whole field of internal variables, without any post-processing of the output variable. This is impossible in any other ANN method without additional assumptions.
- As shown in the presented examples, PGNNIV has both predictive and explanatory capacity. Depending on the aim of the scientist, they can emphasise one capability or the other, by easily adding/removing constraints, changing the penalty parameters or modifying the network topology.
- Last, but not least, PGNNIV has shown better performance than ANNs in aspects such as convergence speed-up, data needs, noise filtering, and extrapolation capacity.

As future work, we point in two directions:

- The methodology is new and lacks of sound theoretical results that justify its application for general classes of problems. Indeed, the performance of the trained PGNNIV may be evaluated by analysing the value of the cost function for the test data, or, even better, the value of each of the loss terms, as each one is associated with one physical aspect of the problem in hands (e.g. mass, momentum or energy conservation, etc.). Although the predictive capacity is directly assessable by evaluating the corresponding loss term of the cost function, it is not the case for the explanatory capacity: if input and output data are not rich enough, or the physical content added is insufficient, the network may have good predictive capacity while misleading the constitutive state model. Consider, for example, the case of an elastic clamped beam subjected to axial load. If the displacement is measured only at the loaded node, many possible stiffness ($K(x) = E(x)A(x)/L$) distributions along the beam are possible resulting in the same displacement associated with a given load. The problem is solved by monitoring the whole displacement field or by explicitly imposing stiffness

homogeneity. This fundamental problem, which we call the slack problem, requires further research, as now its treatment is based on the modeller's previous knowledge and intuition on the problem.

- As the methodology is based on a TensorFlow reframing of mathematical problems, it is especially well suited for easy and structured mesh discretisations. For continuum problems, an octree-based formulation is one of the common strategies to face complex geometries, but this would, in principle, greatly increase the input and output size of the problem. These problems also aggravate when going to higher dimensions in what is called the curse of dimensionality (Poggio et al., 2017). It is, therefore, important to think of strategies to bypass this difficulty.

PIDS have just emerged in the last years, but many researchers have come to the conviction that it is the combination of physical knowledge and machine learning tools the appropriate way to adapt the Big Data paradigm to SBESs, overcoming the growing distrust of physical scientists with AI.

Several numerical experiments have been carried out to illustrate some of the characteristics of the method, in particular, those related to the data-set size and noise, its extrapolation capacity and the size of the model networks. The results show trends and features that are common to ML techniques, in particular, the importance of the data-set, the filtering capacity and the problem of overfitting. Nevertheless, the numerical experiments and examples shown in this work should be reinforced by further research exploring the mathematical structure of the PGNNIV problem, depending on the particular selection of the topology for Y (predictive network) and H (explanatory network), as well as the prescription of the known functions $\boldsymbol{I}$, $\boldsymbol{O}$ and $\boldsymbol{R}$. In any case, PGNNIV are very promising for both predicting and explanatory problems formulated in terms of discrete equations or PDEs, ubiquitous in sciences and engineering.

Scientific outcomes
As a result of this work, we published the following scientific articles:

- Ayensa-Jiménez J., Doweidar M. H., Sanz-Herrera J. A., Doblaré M. "Prediction and identification of physical systems by means of Physically-Guided Neural Networks with meaningful internal layers", *Computer Methods in Applied Mechanics and Engineering*, 2021.
- Ayensa-Jiménez J., Doweidar M. H., Sanz-Herrera J. A., Doblaré M. "On the application of Physically-Guided Neural Networks with Internal Variables to Continuum Problems", *ArXiv, Under review.*

Part II

Application to the study of Glioblastoma evolution

6. A parametric Glioblastoma evolution model in microfluidic devices.

Contents

6.1 Introduction

Although mathematical modelling has been demonstrated to be highly effective in many fields in Physics, Chemistry and Engineering, its ability to accurately represent reality in biological problems is still limited. The high dynamic complexity and non-linearity of the relations involved, the many highly-coupled interactions among different phenomena, the difficulty in identifying the initial state and the lack of data both for quantifying parameters and validating results, make the available models either too simple, or, on the contrary, too complex and cumbersome.

Indeed, in many cases, models incorporate too many parameters, sometimes with unknown values or with a wide range of variation in literature (sometimes orders of magnitude) and with important hidden correlations. The parameters are fitted to the particular data available, leading many times to trivial conclusions, mostly embedded in the model assumptions. This prevents the model from being useful for the whole family of similar problems, and makes the conclusions, results and parameters, difficult to generalise. Despite these strong limitations, *in silico* models, grounded in new biological knowledge and driven by rigorous experimental and clinical data, have become invaluable tools to integrate knowledge across different biological scales, to perform quantitative analyses, and to test hypotheses in a cheap and fast way.

In cancer modelling, in particular, several results have been derived from mathematical approaches, quantifying, for example, the effect into tumour evolution of oxygen, biochemical molecules, extracellular matrix (ECM) stiffness, or cell proliferation rate (Altrock et al., 2015; Byrne, 2010; Deisboeck and Stamatakos, 2010). In this chapter, we focus on glioblastoma (GBM), the most aggressive and lethal among the primary glioma tumours, and also the most frequent, accounting for 17% of all primary brain tumours (Brat, 2012). Survival of patients with this type of tumour who undergo the first-line standard treatments (surgery followed by adjuvant chemotherapy and local radiation) has a median of 14 months since diagnosis and the 5-year survival rate is less than 10% (Oike et al., 2013).

GBM progression is characterised by fast cell proliferation around blood vessels, eventually provoking their collapse, leading to hypoxia. Consequently, a necrotic core is formed around the vessel and the surviving cells migrate towards more oxygenated regions (Brat et al., 2004; Rong et al., 2006), restarting the process of proliferation and creating waves of migrating tumour cells, which are known as pseudopalisades (Brat et al., 2004) and appear in GBM histologies surrounding the necrotic core. This process of successive local hypoxia and cell migration has been proposed as one of the main driving forces of GBM invasion and aggressiveness (Brat and Van Meir, 2004). There have been some attempts to build mathematical models to describe how these tumours grow and respond to therapies, both for *in vitro* experiments, and for *in vivo* models (Hatzikirou et al., 2005; Kim et al., 2016; Rejniak, 2016; Scribner et al., 2014; Suarez et al., 2012; Swanson et al., 2011). In Rejniak (2016), significant aspects, such as the importance of the hypoxic environment in the formation of cellular pseudopalisades (Ayuso et al., 2017) and tumour vasculature (including angiogenesis and vessel cooption), the role of biophysical and biomechanical properties of the ECM in tumour cell invasion, or the role of microenvironmental niches and sanctuaries in

the emergence of acquired drug resistance in tumours were reviewed. Other works focus on analysing the effect of mechanical cues in GBM evolution (Huang and Ingber, 2005; Nagelkerke et al., 2015).

Nevertheless, these models are mostly non-linear, involve highly-coupled multiphysic interactions, and include many parameters. In many occasions, those parameters are difficult to measure and have strong hidden correlations. Moreover, it is usual to have a lack of data both for quantification and validation of the parameters and results (Bearer et al., 2009). Our aim is, in addition to establishing a general computational framework for GBM simulation in microfluidic devices, to present the possibilities in this context offered by a probabilistic methodology that is able to separate the correlated effects found in that study, and to get a more accurate and reliable representation of the experimental results in the parametric space. With that purpose, we approximate the multidimensional probability density function (PDF) of the parameters by means of an appropriate *copula*. *Copulae* allow considering separately the marginal distributions and the dependence between variables in multivariate statistical problems, including those with high correlation. This permits using general models for the marginal distributions, while the variable dependence model can be different (Ang and Chen, 2002). *Copulae* are today used in a wide range of areas in economic sciences and engineering. The most recent models have been successfully applied in portfolio management and optimization (Boubaker and Sghaier, 2013), actuarial analysis (McNeil and Frey, 2017), quantitative finance and risk theory (Kole et al., 2007; Meucci, 2011). A particularly hot topic is the study of climate-agent time series (Munkhammar and Widén, 2017; Solari and Losada, 2011), hydrology (Arya and Zhang, 2017; Laux et al., 2009) and weather and climate research (Laux et al., 2011; Schoelzel and Friederichs, 2008). Some efforts have been made in transportation research (Zou and Zhang, 2016) and traffic policy (Spissu et al., 2009). Recently, *copulae* have been successfully applied in reliability analysis in civil (Kilgore and Thompson, 2011), mechanical and structural (Bartoli et al., 2011), offshore, (Dong et al., 2011) and software (Pham, 2003) engineering. In biology, *copulae* have been used in the field of genetics (Kim et al., 2008) to model gene dependencies. *Copulae* could be an adequate tool to improve the analysis of highly-correlated multiparametric mathematical models such as those appearing in biology, with the added value of providing key information for the optimal design of new experiments with the highest information possible, thus reducing time and cost not only in *in vitro* experiments but also in scarce and costly *in vivo* cases.

In this chapter, we present a new mathematical framework to model the behaviour of cell processes *in vitro* using microfluidic devices, and we particularise it to model the process of GBM progression. Section 6.2 describes the global mathematical framework, with the different equations, parameters, and interactions between the corresponding functions and variables. For the implementation using a finite elements (FE) framework, the reader may consult Appendix B, where additionally the code is validated and an illustrative 3D simulation is shown. Then, in Section 6.3 we address the problem of parameter analysis in the mathematical modelling of nonlinear and correlated multiparametric models, such as the *in vitro* (microfluidic) cell processes, using nonlinear statistical techniques. This statistical approach, based on the concept of *copula* allows for a greater understanding of the hidden structural

correlations between parameters, statistical estimation of the model parameters (including confidence regions) and reliable predictions of the experimental outcomes. The whole approach is then particularised for the *in vitro* processes associated with different stages of GBM evolution. We introduce the particular GBM model in which the main cell processes involved (cell proliferation, differentiation, migration), are related to the oxygen concentration. This leads to a model with a high number of parameters. Then, an extensive analysis of these parameters is made, both, from literature, and by correlating the associated *in silico* results with those derived from a specific microfluidic at-home lab assay: the appearance of auto-induced necrotic core far from the blood vessels in high-density cell regions (Ayuso et al., 2016). The parametric analysis is performed using the statistical tools previously presented. Finally, we also analyse two additional configurations: local hypoxia inducing an oxygen gradient that forces GBM cells to migrate and proliferate with non symmetric and symmetric configurations. These processes are likely the responsible for cell fast migration from an occluded vessel and the subsequent pseudopalisade formation around another vessel, producing a new occlusion (Ayuso et al., 2017; Brat and Van Meir, 2004). These experiments model, therefore, important scenarios of brain cancer evolution.

In summary, the objective of this chapter is to demonstrate the potential of mathematical models, if a proper parametric analysis is conducted, to predict the evolution of cell cultures in microfluidic devices, and in particular, of GBM, as well as to identify the weaknesses and limitations of the parametric model presented when using as a computational tool.

6.2 Mathematical framework for simulating cell cultures in microfluidic devices

In this section, we describe the general mathematical framework for simulating cell cultures in microfluidic devices. We define the different equations that control the evolution of each phenotype cell concentration and the concentration of each chemical substance present in the microfluidic device along time. In general, we shall consider the possible interdependence between the different cell types and the chemical species. Dead cells are considered here as a particular phenotype without capacity for proliferation, migration, or differentiation, but that can be dragged by the fluid flow. In general, we shall consider also the influence of temperature, stiffness, strain, and electric potential, although we consider that this influence is uncoupled with the rest of the equations. We assume that the mechanical variables (fluid velocity, strain, and stiffness of the hydrogel) as well as temperature and electric potential are known or have been computed previously by solving a mechanical, thermal or electromagnetic problem. This framework can be extended by coupling these latter variables with the cell number and species concentrations.

6.2.1 Balance equations for cell populations and species

Cell populations

Cell concentration (number of cells per unit volume), for each type of cell population, is represented as continuous fields $C_i(\boldsymbol{x}, t)$, $i = 1, \ldots, n$ where n is the number

of cell populations dependent on space $\boldsymbol{x} \in \Omega \subset \mathbb{R}^d$ and time $t \in \mathbb{I} \subset \mathbb{R}$, being d the dimension of the problem (that is $d = 1$, $d = 2$, or $d = 3$). We note as $\boldsymbol{C}(\boldsymbol{x}, t) = (C_1, \cdots, C_n)^\intercal$. The master equation that regulates cell population evolution is the transport equation with source terms considering the possible three standard reaction-convection-diffusion phenomena. For the i-th cellular phenotype $C_i, i = 1, \ldots, n$, this equation writes:

$$\frac{\partial C_i}{\partial t} + (\boldsymbol{v} \cdot \boldsymbol{\nabla}) C_i + \boldsymbol{\nabla} \cdot \boldsymbol{q}_i = C_i F_i - \sum_{\substack{j=1 \\ j \neq i}}^{n} C_i F_{ij} + \sum_{\substack{j=1 \\ j \neq i}}^{n} C_j F_{ji}, \quad i = 1, \ldots, n, \quad \boldsymbol{x} \in \Omega, \tag{6.1}$$

where $\Omega \subset \mathbb{R}^d$ represents the domain of study, $\boldsymbol{v}$ is the fluid velocity (convection term), $\boldsymbol{q}_i$ is the flux vector (number of cells at each point per unit surface and per unit time) associated with the migration of phenotype i (diffusion and taxis terms), F_i is the source-term corresponding to the population growth (number of new cells per unit cell and per unit time), F_{ij} is the source (reaction) term corresponding to phenotype switching, that is, the number of cells that differentiates from phenotype i to phenotype j per unit cell and time. All biological phenomena that influence cell migration (different types of taxis) will be modelled by expressions affecting the flux vector $\boldsymbol{q}_i$.

Boundary conditions are defined in the boundary of the domain Ω, $\partial\Omega = \Gamma_{D_i} \cup \Gamma_{R_i}$, where Γ_{D_i} corresponds to the part of the boundary where the concentration C_i is known (Dirichlet boundary part):

$$C_i = f_i, \quad i = 1, \ldots, n, \quad \boldsymbol{x} \in \Gamma_{D_i}, \tag{6.2}$$

while Γ_{R_i} corresponds to the Robin boundary region, where the following general expression is fulfilled:

$$\kappa_i C_i + \frac{\partial C_i}{\partial \boldsymbol{n}} = g_i, \quad i = 1, \ldots, n, \quad \boldsymbol{x} \in \Gamma_{R_i}. \tag{6.3}$$

Species concentrations

Similarly, the transport equation for the i-th chemical species $C'_i, i = 1, \ldots, m$, including the reaction-convection-diffusion phenomena, writes:

$$\frac{\partial C'_i}{\partial t} + (\boldsymbol{v} \cdot \boldsymbol{\nabla}) C'_i + \boldsymbol{\nabla} \cdot \boldsymbol{q}'_i = \sum_{j=1}^{n} C_j F'_{ij}, \quad i = 1, \ldots, m, \quad \boldsymbol{x} \in \Omega. \tag{6.4}$$

Again, $\boldsymbol{v}$ is the fluid velocity, $\boldsymbol{q}'_i$ is the flux vector associated with chemical species i and F'_{ij} the net source-term corresponding to production/consumption of species i per unit cell of phenotype j. Chemical phenomena influencing species transport are again modelled using flux vectors.

The boundary conditions are defined again in $\partial\Omega = \Gamma_{D'_i} \cup \Gamma_{R'_i}$ with C'_i satisfying in the Dirichlet part, $\Gamma_{D'_i}$:

$$C'_i = f'_i, \quad i = 1, \ldots, m, \quad \boldsymbol{x} \in \Gamma_{D'_i}, \tag{6.5}$$

and in the Robin part, $\Gamma_{R'_i}$, C'_i:

$$\kappa'_i C'_i + \frac{\partial C'_i}{\partial \boldsymbol{n}} = g'_i, \quad i = 1, \ldots, m, \quad \boldsymbol{x} \in \Gamma_{R'_i}. \tag{6.6}$$

6.2.2 Physical models for fluxes and sources

In order to close the system of partial differential equations (PDEs) given by Eqs. (6.1) and (6.4), we need to model the biological transport phenomena, that is, to make explicit the relations between the flux vector fields $\boldsymbol{q}_i$ and $\mathbf{q}_i'$ and other physical variables that promote or inhibit cell migration (and/or substance diffusion), such as temperature (thermotaxis), electric potential (electrotaxis), substrate stiffness (durotaxis), strain (tensotaxis)[17], or chemical species concentration (chemotaxis) as well as cell concentrations themselves (cell pedesis, proper diffusion by random walk).

In addition to cell migration, growth, and differentiation of cellular phenotypes and consumption and production of chemical species, that is, the reaction-terms, can be also dependent on similar variables, so they have to be expressed in terms of the same fields.

From now on, we will denote as $\boldsymbol{C}$, $\boldsymbol{C}'$, $\boldsymbol{p}$ the set of cell phenotypes, chemical species and mechanical parameters respectively, that is:

$$\boldsymbol{C} = \begin{bmatrix} C_1 \\ C_2 \\ \vdots \\ C_n \end{bmatrix}, \tag{6.7}$$

$$\boldsymbol{C}' = \begin{bmatrix} C_1' \\ C_2' \\ \vdots \\ C_m' \end{bmatrix}, \tag{6.8}$$

$$\boldsymbol{p} = \begin{bmatrix} p_1 \\ p_2 \\ \vdots \\ p_k \end{bmatrix}. \tag{6.9}$$

6.2.2.1 Source-terms in cell population equations

First, the source-terms in Eq. (6.1), which include proliferation F_i and differentiation F_{ij}, will be analysed

Proliferation

The proliferation of the i-th phenotype can be expressed using an equation of the type:

$$F_i = F_i(\boldsymbol{C}, \boldsymbol{C}', \boldsymbol{p}, \theta), \quad i = 1, \ldots, n. \tag{6.10}$$

Here, θ is the temperature field and $p_1, \ldots, p_k$ are mechanical parameters characterising the substrate (stiffness, anisotropy if required, strain, etc.). In Eq. (6.10) the influence of features such as nutrient concentration or growth factors is included.

For instance, two very familiar proliferation models are:

[17] These two latter are usually included under the term mechanotaxis

- The exponential growth model where F_i is written as

$$F_i(\boldsymbol{C}, \boldsymbol{C}', \boldsymbol{p}, \theta) = r(\theta), \tag{6.11}$$

  being r the growth rate, depending only on the temperature, independent on the current concentration, and leading therefore to an exponential growth of the number of cells per unit volume.

- The logistic growth model, where F_i is written as

$$F_i(\boldsymbol{C}, \boldsymbol{C}', \boldsymbol{p}, \theta) = r_{\max}(\theta) \left(1 - \frac{\sum_{j=1}^{n} C_j}{c_{\text{sat}}}\right), \tag{6.12}$$

  with $r_{\max}$ the maximum growth rate, depending also on the temperature, and c_{sat} the saturation parameter.

Other common models are the Gompertz model (Gompertz, 1825) or its generalisations, Weibull and Frechet models (Rocha and Aleixo, 2013), Richards model (Richards, 1959), which is a generalisation of the logistic model, or the general family of hyperbolastic models (Tabatabai et al., 2005). A brief comparison between the general features of some of these models may be found in Jarrett et al. (2018).

Differentiation

The differentiation of the i-th phenotype to the j-th phenotype can be modelled using a similar equation:

$$F_{ij} = F_{ij}(\boldsymbol{C}, \boldsymbol{C}', \boldsymbol{p}, \theta), \quad i = 1, \ldots, n. \tag{6.13}$$

This very general expression is usually simplified to

$$\mathcal{F}_{ij}(\boldsymbol{C}, \boldsymbol{C}', \boldsymbol{p}, \theta) = \frac{1}{\tau} \Phi(h_{\text{phys}}(\boldsymbol{C}, \boldsymbol{C}', \boldsymbol{p}, \theta)), \tag{6.14}$$

where τ is a characteristic time, Φ is an activation function, for example, the Heaviside function ($\Phi(x) = 0$ if $x \leq 0$ and $H(x) = 1$ if $x > 0$), or the sigmoid function ($\Phi(x) = \frac{1}{2}(1 + \tanh(x))$), and h_{phys} is a function defining the domain of physiological behaviour of a cell, that is, if $h_{\text{phys}}(\boldsymbol{C}, \boldsymbol{C}', \boldsymbol{p}, \theta) \leq 0$, the cell is in its physiological state without stressed or pathological behaviour.

For instance, we can use a p-norm threshold approach where:

$$h_{\text{phys}}(\boldsymbol{C}, \boldsymbol{C}', \boldsymbol{p}, \theta) = ||\boldsymbol{X} - \boldsymbol{X}_{\text{th}}||_p, \tag{6.15}$$

where $\boldsymbol{X} = (C_1', \ldots, C_m', \theta, p_1, \ldots, p_k)^\intercal$ and $\boldsymbol{X}_{\text{th}}$ is a vector of physiological thresholds. A very simple example of these models is the one known as the threshold model (selecting the maximum norm, $p \to \infty$), where:

$$h_{\text{phys}}(\boldsymbol{C}, \boldsymbol{C}', \boldsymbol{p}, \theta) = \max\{X_l - X_{l,\text{th}}, 1 \leq l \leq m + 1 + k\}, \tag{6.16}$$

Using this model, the switch occurs when at least one of the chemical species influencing the metabolic behaviour of the cell falls below a certain threshold.

6.2.2.2 Migration-terms in cell population equations

Now, migration-terms (biologically related transport) are defined. The flux vector $\boldsymbol{q}_i$ is postulated as a linear decomposition:

$$\boldsymbol{q}_i = \boldsymbol{q}_{\mathrm{D},i} + \boldsymbol{q}_{\mathrm{M},i} + \boldsymbol{q}_{\mathrm{E},i} + \boldsymbol{q}_{\mathrm{T},i} + \boldsymbol{q}_{\mathrm{C},i}, \quad i = 1, \ldots, n. \tag{6.17}$$

where the flux vectors $\boldsymbol{q}_{\mathrm{D},i}$, $\boldsymbol{q}_{\mathrm{M},i}$, $\boldsymbol{q}_{\mathrm{E},i}$, $\boldsymbol{q}_{T,i}$ and $\boldsymbol{q}_{C,i}$ are associated with diffusion, mechanotaxis, electrotaxis, thermotaxis and chemotaxis phenomena, respectively.

Random pedesis

We model cell random pedesis using a Fickean diffusion model, that is:

$$\boldsymbol{q}_{\mathrm{D},i} = -\boldsymbol{K}_{\mathrm{D},i} \boldsymbol{\nabla} C_i, \quad i = 1, \ldots, n. \tag{6.18}$$

Here, matrix $\boldsymbol{K}_{D,i}$ is the diffusion matrix that can be expressed as:

$$\boldsymbol{K}_{D,i} = \boldsymbol{K}_{\mathrm{D},i}(\boldsymbol{C}, \boldsymbol{C}', \boldsymbol{p}, \theta), \quad i = 1, \ldots, n. \tag{6.19}$$

Mechanotaxis

The mechanotaxis-term can be expressed in a similar manner, but using the mechanical parameters as the responsible of the cell migration:

$$\boldsymbol{q}_{\mathrm{M},i} = \sum_{j=1}^{k} \boldsymbol{K}_{\mathrm{M},i,j} \boldsymbol{\nabla} p_j, \quad i = 1, \ldots, n. \tag{6.20}$$

where $\boldsymbol{K}_{\mathrm{M},i,j}$ is the mechanotactic motility matrix of cell phenotype i with respect to mechanical parameter p_j, expressed as:

$$\boldsymbol{K}_{\mathrm{M},i,j} = \boldsymbol{K}_{\mathrm{M},i,j}(\boldsymbol{C}, \boldsymbol{p}, \boldsymbol{\nabla p}, \theta), \quad i = 1, \ldots, n. \tag{6.21}$$

Electrotaxis.

For electrotaxis, a similar expression can be written:

$$\boldsymbol{q}_{\mathrm{E},i} = \boldsymbol{K}_{\mathrm{E},i} \boldsymbol{\nabla} V, \quad i = 1, \ldots, n. \tag{6.22}$$

where $\boldsymbol{K}_{\mathrm{E},i}$ is the electrostactic motility matrix and V is the electric potential. As before:

$$\boldsymbol{K}_{\mathrm{E},i} = \boldsymbol{K}_{\mathrm{E},i}(\boldsymbol{C}, \boldsymbol{p}, \theta, V, \boldsymbol{\nabla} V), \quad i = 1, \ldots, n. \tag{6.23}$$

Thermotaxis.

Thermotaxis can be modelled using analogous equations:

$$\boldsymbol{q}_{\mathrm{T},i} = \boldsymbol{K}_{\mathrm{T},i} \boldsymbol{\nabla} \theta, \quad i = 1, \ldots, n. \tag{6.24}$$

where $\boldsymbol{K}_{T,i}$ is the thermotactic motility matrix:

$$\boldsymbol{K}_{\mathrm{T},i} = \mathcal{K}_{\mathrm{T},i}(\boldsymbol{C}, \boldsymbol{p}, \theta, \boldsymbol{\nabla} \theta), \quad i = 1, \ldots, n. \tag{6.25}$$

Chemotaxis

The chemotaxis-term is expressed analogously

$$\boldsymbol{q}_{\mathrm{C},i} = \sum_{j=1}^{m} \boldsymbol{K}_{\mathrm{D},i,j} \boldsymbol{\nabla} C'_j, \quad i = 1, \ldots, n. \tag{6.26}$$

Here, $\boldsymbol{K}_{\mathrm{C},i,j}$ is the chemotactic motility matrix with respect to the chemical species j. As before, it is possible to express the matrix as:

$$\boldsymbol{K}_{\mathrm{C},i,j} = \boldsymbol{K}_{\mathrm{C},i,j}(\boldsymbol{C}, \boldsymbol{C}', \boldsymbol{\nabla C}', \boldsymbol{p}, \theta), \quad i = 1, \ldots, n, \quad j = 1, \cdots, m. \tag{6.27}$$

It is common to use saturation models for chemotaxis (Kim et al., 2016), getting:

$$\boldsymbol{K}_{\mathrm{C},i,j}(\boldsymbol{C}, \boldsymbol{C}', \boldsymbol{\nabla C}', \boldsymbol{p}, \theta) = \frac{\chi_i C_i}{\sigma_j + \lambda_j ||\boldsymbol{\nabla} C'_j||} \tag{6.28}$$

where σ_j and λ_j are parameters depending on the chemical species j and χ_i is a sensitivity parameter depending on the cellular phenotype i.

6.2.2.3 Source-terms and diffusion for chemical species

Source-terms in Eq. (6.4) include production and consumption of species. The net production/consumption of the i-th chemical species by the j-th cell phenotype can be modelled using an equation of the kind:

$$F'_{ij} = F'_{ij}(\boldsymbol{C}, \boldsymbol{C}', \theta), \quad i = 1, \ldots, m, \quad j = 1, \ldots, n. \tag{6.29}$$

The species are considered to be inert, so the only flux term considered is diffusion:

$$\boldsymbol{q}'_i = -\boldsymbol{K}'_{\mathrm{D},i} \boldsymbol{\nabla} C'_i, \quad i = 1, \ldots, m. \tag{6.30}$$

Here, matrix $\boldsymbol{K}'_{\mathrm{D},i}$ is the diffusion matrix that will be expressed as

$$\boldsymbol{K}'_{\mathrm{D},i} = \boldsymbol{K}'_{\mathrm{D},i}(\boldsymbol{C}, \theta, \boldsymbol{p}), \quad i = 1, \ldots, m. \tag{6.31}$$

6.2.3 Extracellular matrix remodelling coupling

One last step in this global framework corresponds to regeneration considerations. Mechanical parameters of the substrate $p_1, \ldots, p_k$ can be seen as constant parameters, such as elastic parameters (Young modulus E, Poisson coefficient ν, hyperelastic constants...) or may be thought as evolving parameters. In the latter situation, a dynamic approach to the problem is adopted, being necessary to define an evolution relationship:

$$\dot{p}_i = R_i(\boldsymbol{C}, \boldsymbol{p}, \theta), \quad i = 1, \cdots, k. \tag{6.32}$$

The PDE system given by Eqs. (6.1) and (6.4), together with the ordinary differential equation (ODE) given by Eq. (6.32) is solved numerically using a FE approach. The details about the implementation and a code validation may be found in Appendix B.

6.3 A stochastic approach for analysing parametric models using *copulae*

6.3.1 Rationale of the approach

Once we have defined a general mathematical model for cell evolution in microfluidic devices, we have to define specific models for each individual phenomenon from the described above, that is, cell growth and differentiation, cell pedesis, mechanotaxis, electrotaxis, thermotaxis and chemotaxis, species diffusion and consumption and ECM remodelling. These models will be described by specific parameters such as the growth rate in the exponential growth model, the physiological thresholds in the differentiation process, or the different motility matrices involved.

All this would lead to a parametric model characterised in terms of some mathematical equations (represented by $\boldsymbol{F}$), some model parameters (represented by $\boldsymbol{\theta}$) and some operation parameters (represented by $\boldsymbol{\lambda}$). We are going to define a framework for dealing with such situations.

6.3.1.1 Deterministic and stochastic models

Let us suppose that our problem may be represented by the following mathematical relationship:

$$\boldsymbol{u} = \boldsymbol{F}(\boldsymbol{\lambda}, \boldsymbol{\theta}), \tag{6.33}$$

with

- $\boldsymbol{u}$ (an m-dimensional vector) the output variable, i.e. the outcome of the experiments, that we measure.
- $\boldsymbol{\lambda}$ the variables which we can control when performing the experiments (such as environmental variables, geometrical parameters, or boundary conditions).
- $\boldsymbol{\theta}$ the model parameters, that we cannot control and whose values must be determined ($\boldsymbol{\theta} \in \Omega$, with Ω the parametric space of dimension n).
- $\boldsymbol{F}$ the mathematical model, that relates the experimental configuration $\boldsymbol{\lambda}$ with the output variables $\boldsymbol{u}$ in terms of the set of parameters $\boldsymbol{\theta}$.

In relation to the accuracy and precision of the model, it is possible to define three levels of analysis:

1. The model is perfect and the experimental measures are noise-free.
2. The model is perfect and the experimental measures are noisy.
3. The model is not perfect and the measurements are noisy.

Only the third case is, in general, realistic[18] in complex problems as the one here analysed. In addition, it is difficult to define universal values for the parameters in

[18] Recall the aphorism popularised by George E. P. Box: "All models are wrong, but some are useful" (Box, 1979).

biological problems, since they are highly-dependent on the particular experimental context.

As a consequence of all the previous observations, it is more appropriate to consider a stochastic approach, and reformulate Eq. (6.33) as:

$$\boldsymbol{U} = \boldsymbol{F}(\boldsymbol{\lambda}, \boldsymbol{\Theta}) \tag{6.34}$$

where $\boldsymbol{U}$ and $\boldsymbol{\Theta}$ are now random vectors of dimensions m and n respectively.

The proposed approach is, therefore, suitable when the following conditions are satisfied:

- Many coupled phenomena are present, being difficult to design experiments able to isolate each of them (complexity).
- The measurement space is large and it is possible to perform a sufficiently big number of experiments N (data availability).

From a mathematical point of view, these two statements may be reformulated as:

- The model $\boldsymbol{F}$ includes many parameters ($n \gg 1$) and/or is non-separable.

  The separability of a model is evaluated by the possibility of approximating $\boldsymbol{F}$ as:

  $$\boldsymbol{F}(\boldsymbol{\lambda}, \boldsymbol{\theta}) \simeq \boldsymbol{F}^M(\boldsymbol{\lambda}, \boldsymbol{\theta}) = \sum_{i=1}^{M} \prod_{j=1}^{n} F_{i,j}(\boldsymbol{\lambda}, \theta_j). \tag{6.35}$$

  The lower M, the easier to define a set of different experimental configurations $\mathcal{S} = \{\boldsymbol{\lambda}^j\}_{j=1,\dots,k}$ to isolate each of the parameters θ_j by solving separately each equation $\boldsymbol{u}^j = \boldsymbol{F}^M(\boldsymbol{\lambda}, \boldsymbol{\theta})$. Although this separability definition is not very rigorous, it is enlightening enough for our purposes.
- The dimension of the measurement space is high ($m \gg 1$)[19] and/or the sample size is large enough ($N \gg 1$).

6.3.1.2 Parametric models in computational biology

Previous research in computational biology has mainly focused on the value of the parameters or, in the best case, on their (individual) uncertainty. However, in many cases, the fitting process is very complex and the parameters are highly correlated due to, at least, two facts:

- **Sample variability**: Different physical phenomena may have an inherent correlation supported by physical considerations, being this correlation independent of the experiments performed or the model used. For example, when working with biological models, the parameters have always a high variability as they are related to cell populations that may be from different patients, commercial lines or that may have been subjected to different process for culture and preparation, among many other causes of variability.

[19] Without loss of generality, we consider that m is, actually, the reduced dimensionality of the space or, in other words, that all variables of the ambient space are independent.

- **Model complexity**: The non-separability of the model and/or the experiments does not allow to isolate the particular mechanisms. The mathematical parameters related to these phenomena (e.g. oxygen diffusion and cell oxygen consumption coefficients) should present a strong correlation, although this correlation does not have a physical meaning, being inherent to the model or to the experimental set-up.

Thanks to the flexibility, portability, automation, integration and miniaturization of the microfluidic experiments, a huge amount of data may be generated. Accordingly, this type of experiments is a perfect domain of application for the framework presented herein.

6.3.2 *Copula*-based parametric model analysis

6.3.2.1 Concept of *copula*

In probability and statistics, a *copula* is an n-multivariate probability distribution function $\boldsymbol{U}$ whose marginals, U_i, are uniform distributions on $[0, 1]$ (Jaworski et al., 2010). They were introduced by Sklar (1959). As the marginal distributions are known, a *copula* describing the structural dependence between variables is enough to perfectly define the model.

As mentioned, a *copula* is a function $C : I^n \to I$, where $I = [0; 1]$ such that:

- For $u_1, \ldots, u_n \in I$, and if $u_i = 0$ for some $1 \leq i \leq n$:

$$C(u_1, \ldots, u_n) = 0. \tag{6.36}$$

- For $u_j \in I, 1 \leq j \leq n$:

$$C(1, \ldots, 1, u_j, 1, \ldots, 1) = u_j. \tag{6.37}$$

- C is n-non decreasing, i.e., for each $B = \prod_{i=1}^{n} [x_i; y_i] \subset I^n$, the C-volume of B is non-negative:

$$\int_B dC(u) = \sum_{\boldsymbol{z} \in \times_{i=1}^{n} \{x_i; y_i\}} (-1)^{\#\{k : z_k = x_k\}} C(\boldsymbol{z}) \geq 0. \tag{6.38}$$

We can distinguish between parametric and non-parametric *copulae*. Here, we use a hybrid approach, as we fit the marginal distributions by means of kernel estimators (Wand and Jones, 1994) of the PDFs and use a parametric *copula*. With this approach, the required data-set grows as $\mathcal{O}(n)$ where n is the space dimension.

6.3.2.2 Fitting and model validation

Let us suppose we have a data-set of values for different experiments, $\boldsymbol{\lambda}^i$, characterised in terms of a resultant mean value $\boldsymbol{\mu}^i$ and a variance-covariance matrix $\boldsymbol{\Sigma}^i$, $i = 1, \ldots, N$, obtained from different measurements associated with the experimental configuration i. As the assumed model $\boldsymbol{F}$ is known, it is possible, for each piece of data i, to obtain the set of parameters $\boldsymbol{\theta}^i$ which best fits it.

Once $\boldsymbol{\theta}^i$, $i = 1, \ldots, N$ are obtained, the next step is the adjustment of the marginal distributions. The values θ_j^i, $j = 1, \ldots, n$, are used for fitting the marginal random variable Θ_j whose cumulative distribution is assumed to be G_j. Here, we can follow either a parametric (that is, $G_j(x) = G_j(x; \boldsymbol{\alpha}_j)$) or a non-parametric approach (which is the one followed in this work). The values θ_j^i are therefore transformed into uniformly distributed ones via the standard transformation $y_j^i = G_j(\theta_j^i)$. As $\boldsymbol{y}^i$ are considered uniformly distributed with a joint dependence, it is possible to fit this structural dependence using parametric *copulae*.

To summarise, the steps of the training process are:

1. Problem minimisation to obtain $\boldsymbol{\theta}^i$. We have to minimise the residual function R^i:
$$R^i(\boldsymbol{\theta}) = \left(\boldsymbol{F}(\boldsymbol{\lambda}^i, \boldsymbol{\theta}) - \boldsymbol{\mu}^i\right)^\intercal (\boldsymbol{\Sigma}^i)^{-1} \left(\boldsymbol{F}(\boldsymbol{\lambda}^i, \boldsymbol{\theta}) - \boldsymbol{\mu}^i\right) \tag{6.39}$$
where the Mahalanobis distance (De Maesschalck et al., 2000) has been used to take into account the sample variability (see Chapter 3 for more details). Assuming that $\boldsymbol{\Sigma}^i = {\sigma^i}^2 \boldsymbol{I}$, Eq. (6.39) can be rewritten as:
$$R^i(\boldsymbol{\theta}) = \frac{1}{{\sigma^i}^2} \left\| \boldsymbol{F}(\boldsymbol{\lambda}^i, \boldsymbol{\theta}) - \boldsymbol{\mu}^i \right\|^2 . \tag{6.40}$$
2. Kernel density estimation of the marginal distributions from the data θ_j^i.
3. Transformation into uniformly distributed values y_j^i.
4. *Copula* fitting of the $\boldsymbol{y}$ data to capture the joint dependence.

The presented sequence of steps allows moving from a data-set $\mathcal{D} = \{\boldsymbol{\theta}^i\}_{i=1,\ldots,N}$ to a probabilistic model for the random vector $\boldsymbol{\Theta}$: the marginal kernel densities and the *copula* encoding the structural dependence.

To avoid overfitting, we follow a typical train-test approach: we divide the pairs $\boldsymbol{\lambda}^i$ - $\boldsymbol{u}^i$ (where $\boldsymbol{u}^i$ includes $\boldsymbol{\mu}^i$ and $\boldsymbol{\Sigma}^i$) in two separate subsets, one used for training and the other used for testing.

If we consider now the test data-set, the procedure is:

1. Problem minimisation to obtain $\boldsymbol{\theta}^i$, by minimising the residual given by Eq. (6.39).
2. Tests of the statistical fitting:

- Marginal fitting: q-q plots, histograms, empirical cumulative distribution functions (ecdf), boxplots, parametric or non-parametric statistical tests (Kottegoda and Rosso, 2008).
- Joint 2 vs 2 correlations: correlations, scatterplots, parametric statistical tests for correlations (Kottegoda and Rosso, 2008).
- Whole joint structural dependence: multivariate parametric and non-parametric statistical tests (Fan, 1997).

6.3.2.3 Model analysis and parameter estimation

Once the distribution of the random vector $\boldsymbol{\Theta}$ is learned, the model is known from a probabilistic point of view. The first straightforward application is parameter estimation[20]. It is clear that a point estimate of the model parameters is given by:

$$\hat{\boldsymbol{\theta}} = \mathbb{P}[\boldsymbol{\Theta}] \tag{6.41}$$

where $\mathbb{P}$ is a central tendency operator, for example, the expectation operator $\mathbb{E}$, minimising the L^2 squared norm dispersion (its minimum is the variance), or the geometric median operator $\mathbb{M}$, minimising the L^2 norm dispersion (its minimum is the mean absolute deviation).

However, it is more interesting to perform a confidence region estimation. As suggested in Hyndman (1996), we use the so-called highest density regions (HDR) because of their easy interpretation, straightforward generalisation to multi-dimensional spaces and direct computation. Recall that, under some distributional assumptions (e.g. normality assumption), HDR computation is reduced to other standard confidence region computation techniques (e.g. χ^2 quantile tolerance ellipsoids). HDR computation enables reliable parameter estimation since, given a significant level threshold α, it is possible to define an HDR region in which the parameters are located with a $p = 1 - \alpha$ probability. This may be performed for single parameters, or, in general, k-tuples of parameters.

This methodology is also applicable to conditional distributions. Let us suppose that we know the value of a certain subset of parameters $\boldsymbol{\theta}^*$ and let us define $\boldsymbol{\theta} = (\boldsymbol{\theta}', \boldsymbol{\theta}^*)$. Knowing the distribution $\boldsymbol{\Theta}$, that is obtained after the fitting-validation procedure, it is possible to define the conditioned distribution of $\boldsymbol{\Theta}$ given $\boldsymbol{\Theta}^* = \boldsymbol{\theta}^*$ by its density f' defined in terms of the density f of $\boldsymbol{\theta}$:

$$f'(\boldsymbol{\theta}'|\boldsymbol{\theta}^*) = \frac{f(\boldsymbol{\theta}', \boldsymbol{\theta}^*)}{\int f(\boldsymbol{\eta}, \boldsymbol{\theta}^*)\, d\boldsymbol{\eta}}, \tag{6.42}$$

so all HDR computations are now applied to the distribution of $\boldsymbol{\Theta}$ given $\boldsymbol{\Theta}^* = \boldsymbol{\theta}^*$ by replacing f by f'.

[20] It is important to emphasize that with "parameter estimation" we refer to the parameters of the mathematical model, not to the parameters of the distributions used in the statistical characterization (actually, the statistical characterization may be non-parametric), that may be estimated via common statistical inference techniques.

6.3.2.4 Design of experiments

The techniques within design of experiments (DoE) aim at maximising the information obtained from each performed experiment, in order to reduce the number of them required (Fisher, 1937a). In particular, in this work, we use the techniques within the Bayesian experimental design (BED), based on the Bayesian interpretation of probability.

BED aims at maximising the expected utility of the experiment outcome (Chaloner and Verdinelli, 1995). The utility function expresses how useful is the information provided by an experiment. Of course, the optimal experiment design depends on the chosen utility criterion. In this work, the definition of the utility function is based on the Shannon entropy or information entropy (Shannon, 1948).

Under these assumptions, the utility of an experiment $\boldsymbol{\lambda}$ is defined as the prior-posterior gain in Shannon information. That is, the additional information that the experimental configuration $\boldsymbol{\lambda}$ provides about our model parameters. The utility $U(\boldsymbol{\lambda})$ then writes:

$$U(\boldsymbol{\lambda}) = \int_{\mathcal{U}} \int_{\Omega} f(\boldsymbol{\theta}, \boldsymbol{u}|\boldsymbol{\lambda}) \log f(\boldsymbol{u}|\boldsymbol{\theta}, \boldsymbol{\lambda}) \, d\boldsymbol{\theta} d\boldsymbol{u} - \int_{\mathcal{U}} f(\boldsymbol{u}|\boldsymbol{\lambda}) \log f(\boldsymbol{u}|\boldsymbol{\lambda}) \, d\boldsymbol{u}, \tag{6.43}$$

where $\boldsymbol{u} \in \mathcal{U}$ is the experimental observation and $\boldsymbol{\theta} \in \Omega$ is a vector of parameters to be determined. $f(\boldsymbol{u}|\boldsymbol{\theta}, \boldsymbol{\lambda})$ is the PDF of obtaining an experimental outcome $\boldsymbol{u}$ given the experimental configuration $\boldsymbol{\lambda}$ and the model parameters $\boldsymbol{\theta}$ and $f(\boldsymbol{\theta}, \boldsymbol{u}|\boldsymbol{\lambda})$ is obtained as follows, being $f(\boldsymbol{\theta})$ the prior PDF over the parameters $\boldsymbol{\theta}$:

$$f(\boldsymbol{\theta}, \boldsymbol{u}|\boldsymbol{\lambda}) = f(\boldsymbol{\theta}) f(\boldsymbol{u}|\boldsymbol{\theta}, \boldsymbol{\lambda}). \tag{6.44}$$

If we assume that $\boldsymbol{u}$ has a multivariate normal distribution (what is indeed not necessary but has been here considered for illustration purposes) with variance - covariance matrix $\boldsymbol{\Sigma} = \sigma^2 \boldsymbol{I}$, and knowing that the entropy of a multivariate normal distribution of dimension n is only dependent on the standard deviation σ (Ahmed and Gokhale, 1989), we have the following expression for the utility:

$$U(\boldsymbol{\lambda}) = -\frac{n}{2} \log\left(2\pi \mathrm{e} \sigma^2\right) - \int_{\mathcal{U}} f(\boldsymbol{u}|\boldsymbol{\lambda}) \log f(\boldsymbol{u}|\boldsymbol{\lambda}) \, d\boldsymbol{u}, \tag{6.45}$$

where e is the Euler number.

As we work under the assumptions detailed above, Eq. (6.45), representing the utility of an experimental configuration $\boldsymbol{\lambda}$, may be computed via numerical integration.

6.4 Particularisation for the Glioblastoma model

The presented general framework for simulating cell cultures and for analysing parametric models is sufficiently general to model a broad range of biological problems. However, it is complex in terms of the solution of the equations (highly-nonlinear, strongly-coupled, and very stiff equations), number of parameters involved (many times unknown or difficult to measure) and very difficult to validate. This is why this general framework will be next particularised and simplified for the problem

of interest, that is the GBM evolution, keeping the most important influences and discarding those that can be considered comparatively negligible.

Here, we shall see how to simplify it focusing the research only on the most relevant aspects of our particular experiments. For example, as we will see, the dependence on external variables $\boldsymbol{H}$ is not considered in our experiments, so these equations of the general framework will not be used.

6.4.1 Experimental data

We refer to previous works in our group Ayuso et al. (2016, 2017) as well as the published version of this chapter (Ayensa-Jiménez et al., 2020b) for a further explanation of the details of the experiments, that were not carried out by the author of this thesis. Here we will include only the basics for a good understanding of the data used when fitting the parameters.

In order to form a 3D structure, oxygen impermeable microfluidic devices (BEOnChip Ltd.) consisting of a central chamber and two lateral microchannels were used (Fig. 6.1). They had different dimensions and were made of SU-8, polystyrene or cyclic olefin polymer, using different fabrication processes (Ayuso et al., 2016, 2017). 3D distribution of cells was achieved within the central chamber, using collagen hydrogel.

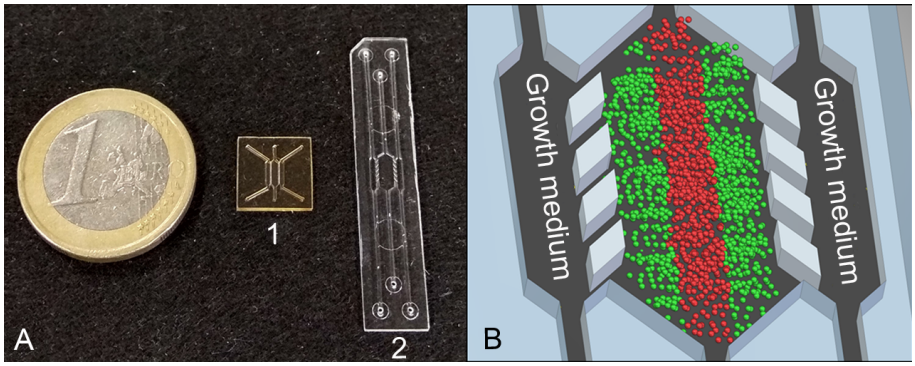


Figure 6.1: Description of the microdevice. A: Microfluidic devices. A1 - SU-8 device, A2 - polystyrene/COP device. B: Schematic view of the central region of the polystyrene/COP microdevice and necrotic core formation. Modified from Ayuso et al. (2016).

Laser confocal and fluorescence images were acquired at different focal planes within each microdevice using a Nikon Eclipse Ti-E C1 confocal microscope. Images were analysed using Fiji software (`http://fiji.sc/Fiji`). Fluorescence intensity across the central microchamber of the microdevice was quantified, in accordance with the software instructions, by selecting a rectangular region across the central microchamber after creating the projection image. In order to transform fluorescence intensity into cell concentration, this latter is assumed to be proportional to the fluorescence intensity. The constant of proportionality is calculated assuming that the integral of the initial cell concentration along the chamber equals the total amount of cells.

In order to produce the necrotic core formation, a high density of cells (40×10^6 cells/mL) was embedded in the collagen hydrogel and injected within the central

microchamber. Growth medium was refreshed every day and the culture was maintained for 6 days. Nutrients and oxygen are not able to reach the central part of the device due to cell consumption close to the microchannels, thus causing cell death in the central region and the appearance of an autoinduced necrotic core (Fig. 6.2), mimicking the parts of the tumour far from functional blood vessels (Ayuso et al., 2016). Visualisation of the necrotic core was performed by calcein/propidium iodide staining.

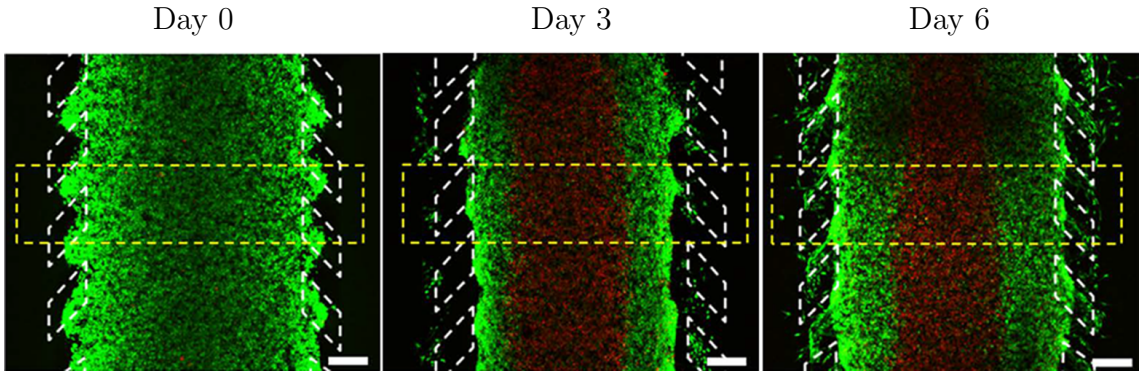


Figure 6.2: Necrotic core formation. U-251 cells were seeded at the concentration of 40×10^6 cells/mL within the central microchamber. Growth medium was perfused every day through the lateral channels. Viable cells were stained green with calcein AM and dead cells were labelled red with propidium iodide (modified from previous work (Ayuso et al., 2016)). Scale bar is 400 μm.

To promote pseudopalisade formation, cells were seeded at a low density (4×10^6 cells/mL) within the central microchamber and one lateral channel was blocked, while constant medium flow was perfused through the other lateral channel. As the region next to the sealed channel was hypoxic, cells migrated towards the perfused channel, rich in nutrients and oxygen (Fig. 6.3). In the control device, both lateral channels were left open and migration was not observed (Ayuso et al., 2017).

Finally, in the case of double pseudopalisade formation, cells were seeded again at low density (4×10^6 cells/mL) within the central microchamber. In this case, the medium was perfused through both lateral channels and refreshed every day during 21 days. Hypoxic conditions in the centre of the microchamber induced cell migration towards the perfused channels and invasion of both of them (see Fig. 6.4).

6.4.2 Mathematical equations

6.4.2.1 General evolution equations

Transport equations

The mathematical model used here for modelling GBM evolution is presented, following the general framework presented in Section 6.2. Due to the typology of the experiments and for simplicity, the problem may be approximated as one-dimensional, disregarding differences along the direction parallel to the lateral channels (see Fig. 6.5). We consider two cell phenotypes (dead cells and alive cells) interacting in the microfluidic device with one chemical species, i.e. oxygen, acting as a regulator of

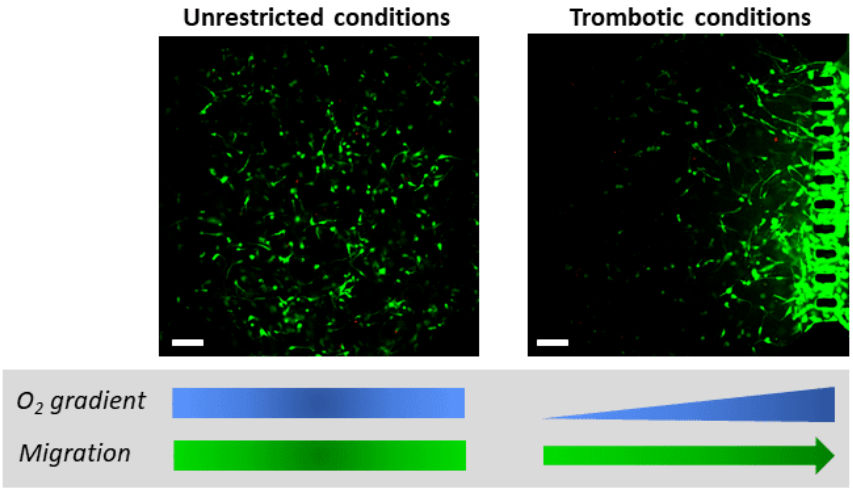


Figure 6.3: Pseudopalisade formation. U-251 MG cells at 4×10^6 cells/mL were cultured within the microdevice. To mimic thrombotic conditions, medium flow was enabled to flow only through the right microchannel. Under unrestricted conditions, medium was refreshed once a day, through both lateral microchannels (modified from previous work (Ayuso et al., 2017)). Scale bar is 400 μm.

cell processes. These assumptions come from previous experiments (Ayuso et al., 2016) that showed that the distribution of other nutrients (glucose) is not responsible for changes in the cells configuration, being oxygen the main (and almost unique) stimulus for cell changes.

Even though some papers consider three (Martínez-González et al., 2012) or four (Ayuso et al., 2017) cell phenotypes, here, only two phenotype populations (alive and dead cells) have been considered, thus disregarding possible changes in phenotype along the duration of our experiments. This does not mean that all cells in the chamber equally proceed in terms of proliferation, migration or oxygen consumption, since all these processes depend as well on the particular conditions of the surrounding environment, but that all cells respond equally when they are subjected to the same local environmental conditions. We consider with this assumption that cell adaptation requires longer periods under stressing conditions to modify permanently their internal machinery. Another reason for this assumption is that, in absence of gene expression techniques, it is impossible to distinguish between differentiation into a different phenotype or a change in the cell behaviour as a reaction to environmental changes, so, considering one single phenotype for alive cells results in fewer parameters and a better understanding of the role of the different phenomena and parameters, an easier calibration and less uncertainty.

The variable defining the number of cells for each population at each point and time is their respective concentration $u_i = C_i$ (cells/mL), $i = 1, 2$, where $i = 1$ for alive cells and $i = 2$ for dead cells. Similarly, we call $u_0 = O_2$ the continuum field of oxygen concentration (in mmHg). Thus, we shall denote by $\boldsymbol{u} = (u_0, u_1, u_2)$ the

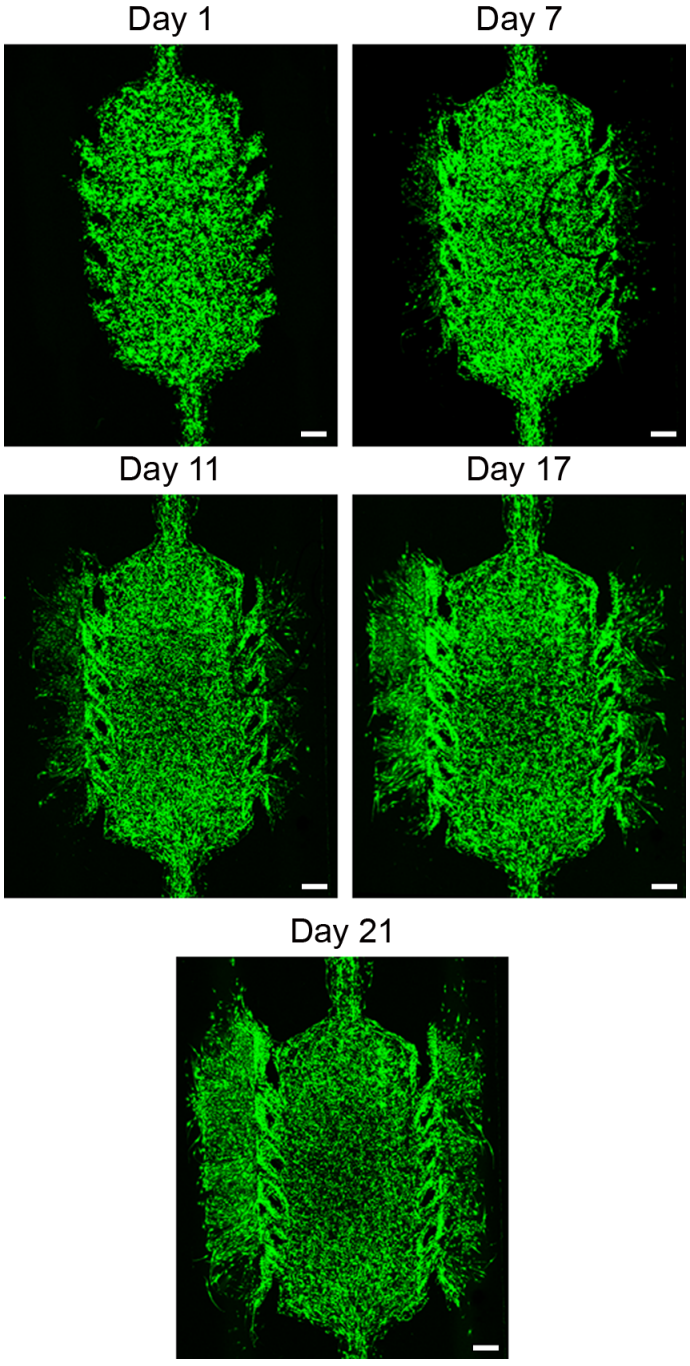
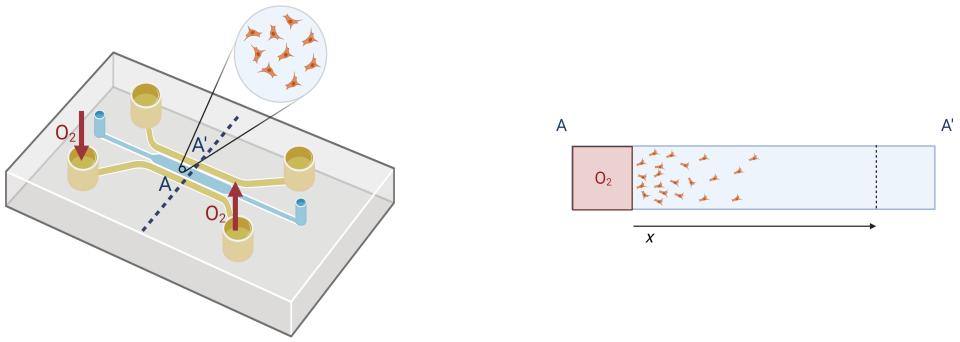


Figure 6.4: Double pseudopalisade formation. EGFP transduced U-251-MG cells were embedded within the central microchamber at the concentration of 4×10^6 cells/mL. Growth medium was changed every day and the evolution of the cell culture over time was observed. Scale bar is 400 μm.



(a) Scheme of the experimental configuration. **(b)** 1D approximation of the cell culture.

Figure 6.5: Typical experimental configuration for modelling cell cultures. Due to the big length of the lateral channels with respect to the width of the chamber, the geometry of the model is assumed one-dimensional with length the width of the chamber, L. The cell and the oxygen concentrations are associated with continuum fields $C_n = C_n(x,t)$ and $C_d = C_d(x,t)$ respectively. At the location of the lateral channels, $x = 0, L$, boundary conditions for both fields have to be supplied. Created with BioRender.com.

vector of field variables. The master equation that regulates each variable evolution is the transport equation including source terms:

$$\frac{\partial u_i}{\partial t} + \frac{\partial f_i}{\partial x} = s_i, \quad i = 0, 1, 2, \tag{6.46}$$

with f_i the flow term that will include diffusion and chemotaxis for cells and diffusion for oxygen and s_i the source term associated with production (proliferation) or loss (death of cells and consumption of oxygen). Note that Eq. (6.46) is, in general, nonlinear and should embed the coupling between the evolution of the different cell populations regulated by the oxygen concentration that may influence proliferation, migration and death, and oxygen consumption kinetics. In our microfluidic device, with a controlled production of the hydrogel, we can assume that it is homogeneous and remains with the same properties all along the experimental or, alternatively, that the potential changes in those properties do not significantly affect the cell properties or the oxygen diffusivity. For alive cells, migration is split in oxygen mediated chemotaxis and pedesis. Dead cells are considered as an inert population. Besides, growth and death rates are also assumed to be dependent on nutrients and oxygen environment. More in detail, the biological hypotheses of the model are:

- Flow term associated with cells include random pedesis (diffusion) and oxygen-mediated chemotaxis (towards zones with a higher oxygen concentration).
- Living cells follow the *go-or-grow* paradigm, that is, when the oxygen level is low (hypoxia) its behaviour is mainly migratory and when the oxygen level is high (normoxia) its behaviour is mainly proliferative.
- Both cell growth and migration are constrained to the space availability.
- The two main mechanisms causing living cells death are apoptosis (programmed death to control cell development and/or growth) and necrosis (massive death that

occurs when cells are exposed to extreme variations in physiological conditions). In our case, necrosis will occur due to extremely low levels of oxygen (anoxia) so oxygen is the only variable mediating the overall cell death.

- Dead cells are considered as an inert population so they do not move or disappear throughout the duration of the experiment.
- Inside the microfluidic device, cells are embedded in a homogeneous ECM (collagen hydrogel), whose mechanical properties, and temperature are kept constant and, therefore, do not affect the different model parameters.
- Oxygen changes are due solely to consumption by the normoxic population (cellular respiration), in addition to the one supplied by the lateral channels.

With all these assumptions, the flow terms may be written as:

$$f_0 = -D_0 \frac{\partial u_0}{\partial x}, \tag{6.47a}$$

$$f_1 = -D_1 \frac{\partial u_1}{\partial x} + K_1 \Pi_{\mathrm{go}} F_{\mathrm{go}} u_1 \frac{\partial u_0}{\partial x}, \tag{6.47b}$$

$$f_2 = 0, \tag{6.47c}$$

where D_0 is the oxygen diffusion coefficient, D_1 the normoxic phenotype diffusion coefficient, K_1 the chemotaxis coefficient for the normoxic population and $\Pi_{\mathrm{go}} = \Pi_{\mathrm{go}}(u_0)$ and $F_{\mathrm{go}} = F_{\mathrm{go}}(u_1, u_2)$ are nonlinear dimensionless corrections, accounting for the effect of the oxygen and cell concentrations on cell migration itself, respectively, that we will discuss later.

Regarding the source terms, we can write:

$$s_0 = -\alpha_0 \Pi_{\mathrm{c}} u_1, \tag{6.48a}$$

$$s_1 = \alpha_1 \Pi_{\mathrm{gr}} F_{\mathrm{gr}} u_1 - \alpha_{12} \Pi_{\mathrm{d}} u_1, \tag{6.48b}$$

$$s_2 = \alpha_{12} \Pi_{\mathrm{d}} u_1, \tag{6.48c}$$

where α_0 is the oxygen consumption rate associated with the normoxic population, α_1 is the proliferation rate and α_{12} is the death rate. Again, $\Pi_{\mathrm{gr}} = \Pi_{\mathrm{gr}}(u_0)$, $F_{\mathrm{gr}} = F_{\mathrm{gr}}(u_1, u_2)$, $\Pi_{\mathrm{d}} = \Pi_{\mathrm{d}}(u_0)$ and $\Pi_{\mathrm{c}} = \Pi_{\mathrm{c}}(u_0)$ are nonlinear correction functions modelling the effect of oxygen and cell concentrations on cell growth, cell death and oxygen consumption, to which we will return later. Recall that the apoptotic or necrotic processes are included here as specific differentiation types to the specific phenotype of dead cells.

Boundary and initial conditions

Eq. (6.46) has to be complemented with the corresponding boundary conditions for the oxygen and normoxic phenotype[21]. We assume here the general case of Robin-like boundary conditions, that is:

$$I_i(x^*, t)\,(u_i - g_i(x^*, t)) + J_i(x^*, t)\,(f_i - h_i(x^*, t)) = 0. \quad i = 0, 1. \tag{6.49}$$

[21] Note that as the necrotic phenotype has no flux term, its equation reduces to an ODE

In the previous equation, $x^* = 0, L$, where L is the width of the chamber. $I_i = I_i(x^*, t)$ and $J_i = J_i(x^*, t)$ are functions characterising the boundary permeability to cell movement or oxygen flow through the boundary, and $g_i(x^*, t)$ and $h_i(x^*, t)$ functions defining the controlled value of cell or oxygen concentration and flux at the boundaries. Note that, if $I_i = 1$ and $J_i = 0$, we have Dirichlet boundary conditions (cell population concentration prescribed at the boundary) and, if $I_i = 0$ and $J_i = 1$, we have Neumann boundary conditions.

Finally, the initial conditions for oxygen and each cell population concentration have to be defined:

$$u_i(x, t = 0) = u_i^0(x), \quad i = 0, 1, 2, \tag{6.50}$$

where $u_i^0(x)$ is a known function.

6.4.2.2 Nonlinear corrections

In order to particularise the general equations presented for modelling the population and species evolution in the *in vitro* experiments made on GBM cells, it is necessary to particularise the nonlinear corrections of the model, that is, Π_{go}, Π_{gr}, Π_{d}, F_{go}, F_{gr} and Π_{c} in terms of the field variables $\boldsymbol{u}$.

Functions F_{gr} and F_{go} are corrections for cell growth and migration due to space constraints, Π_{c} is related to oxygen consumption kinetics and functions Π_{go} and Π_{gr} are used for modelling how the different cell mechanisms are activated depending on oxygen levels, $D_0 = D_{O_2}$ is the oxygen diffusion coefficient, $D_1 = D_n$ is the diffusion of the normoxic cell population coefficient, $\chi = K_1$ is the normoxic cell population chemotaxis coefficient, $1/\alpha_1 = \tau_{\text{gr}}$ is the characteristic proliferation time, $1/\alpha_{12} = \tau_{\text{d}}$ is the death characteristic time and $\alpha = \alpha_0$ is the oxygen consumed per unit time and cell.

Since cell populations adapt their behaviour to oxygen supply and space availability, two major corrections should be considered in the migration term:

- Cellular motility is only possible when the surrounding tissue is not cell saturated (Stramer and Mayor, 2017).
- Migration following the oxygen gradient happens only when the oxygen supply is below a critical threshold, activating the cell motility mechanism (Carreau et al., 2011; Lu and Kang, 2010).

According to these two major assumptions, a rectified linear unit (ReLU) activation function was here used to take into account each of these two phenomena, so the chemotaxis corrections may be written as:

$$\begin{aligned} F_{\text{go}}(u_1) &= \phi_-(u_1; c_{\text{sat}}), \\ \Pi_{\text{go}}(u_0) &= \phi_-(u_0; O_2^H), \end{aligned} \tag{6.51}$$

with

$$\phi_-(x; \theta) = \begin{cases} 1 & \text{if} \quad x \leq 0 \\ 1 - \frac{x}{\theta} & \text{if} \quad 0 \leq x \leq \theta, \\ 0 & \text{if} \quad x > \theta \end{cases} \tag{6.52}$$

where θ is a threshold parameter.

Here O_2^H is the hypoxia-induced migration activation threshold, representing the oxygen level below which cell migration is activated and c_{sat} is the cell saturation concentration.

The proposed model is in line with the *go-or-grow* dichotomy established in GBM literature (Hatzikirou et al., 2012). Cell energetic resources are spent either in cell migration or in cell proliferation. However, cell proliferation also depends on other needs as nutrient supply or availability of space to grow and split. According to this, we propose a model combining logistic growth and the *go-or-grow* paradigm based on oxygen supply. We define the growth corrections as:

$$\begin{aligned} F_{\text{gr}}(C_1, C_2) &= \rho(u_1 + u_2; c_{\text{sat}}), \\ \Pi_{\text{gr}}(u_0) &= \phi_+(u_0; O_2^H), \end{aligned} \tag{6.53}$$

with

$$\phi_+(x;\theta) = \begin{cases} 0 & \text{if} \quad x \leq 0 \\ \frac{x}{\theta} & \text{if} \quad 0 \leq x \leq \theta, \\ 1 & \text{if} \quad x > \theta \end{cases} \tag{6.54}$$

and ρ is the logistic correction factor:

$$\rho(x;\theta) = 1 - \frac{x}{\theta}. \tag{6.55}$$

The function Π_{gr} is responsible for the *go-or-grow* dichotomy and F_{gr} is the logistic model for cell population growth.

Cell death is a natural process depending on many factors and agents and has an inherent stochastic nature (Galluzzi et al., 2018). Anoxia is one fundamental cause of cell death (Sendoel and Hengartner, 2014). Here, a two-parameter sigmoid model is used, able to capture necrosis and apoptosis phenomena:

$$\sigma_-(x;\theta,\Delta\theta) = \frac{1}{2}\left(1 - \tanh\left(\frac{x-\theta}{\Delta\theta}\right)\right), \tag{6.56}$$

where θ is a threshold parameter and $\Delta\theta$ is a sensitivity parameter. They can be seen as a pair of location-spread parameters summarising the stochastic behaviour of the considered phenomenon. With this notation:

$$\Pi_{\text{d}}(u_0) = \sigma_-(u_0; O_2^A, \Delta O_2^A), \tag{6.57}$$

with O_2^A and ΔO_2^A the location and spread parameters associated with the oxygen concentration inducing cell death.

Finally, oxygen consumption is a complex phenomenon related to the oxidative phosphorylation that occurs in the membrane of cellular mitochondria (Meister, 1956). Many authors have considered a zero-order consumption function, i.e. a constant consumption rate independent of oxygen concentration O_2 (Pogue et al., 2001; Secomb et al., 1993; Tannock, 1972). A more realistic assumption is describing the consumption function using the Michaelis-Menten model for enzyme kinetics

(Martínez-González et al., 2012; Patel et al., 2001). This model is a correction of the linear consumption and states that:

$$r(x;K) = \frac{x}{x+K}, \tag{6.58}$$

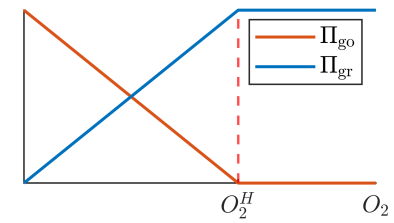
where K is a model parameter. This type of equation was observed for the oxygen consumption rate in the late 1920s and early 1930s (Tang, 1933). This equation describes more accurately the consumption at low oxygen concentrations and is compatible with previous constant consumption rate models, thus allowing the possibility of comparison with previous studies.

Using this notation and the one introduced in our mathematical formulation, we can write:

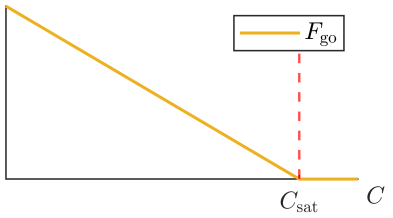
$$\Pi_{\mathrm{c}}(u_0) = r(u_0; k_{\mathrm{m}}) \tag{6.59}$$

where k_{m} is the oxygen concentration at which the reaction rate is half of the rate in a fully oxygenated medium, therefore related to the oxidative phosphorylation kinetics, and the cell structure and morphology (size and number of mitochondria, etc.) and the diffusion process in the cytoplasm.

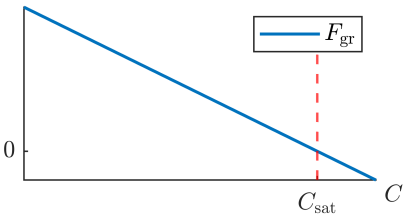
The different nonlinear correction functions are illustrated in Fig. 6.6.



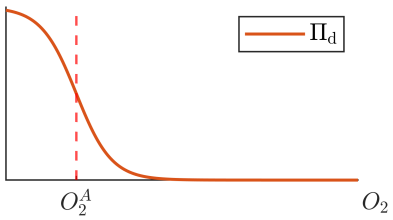
(a) *Go-or-grow* corrections.



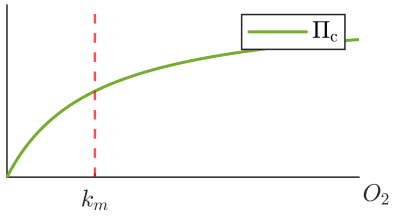
(b) Migration saturation correction.



(c) Growth saturation correction.



(d) Death correction.



(e) Oxygen consumption correction.

Figure 6.6: Nonlinear correction functions. The nonlinear correction functions are shown, together with their shape parameters.

6.4.2.3 Details about boundary and initial conditions

In the microfluidic device, the culture chamber is connected to the oxygen supplying channels by means of small cavities. The volume and the number of these cavities depend on the microfluidic device design and they are directly related to potential cell losses during the experiment. Actually, when cell populations arrive to the interface between these cavities, some of them may reach the channel and leave the culture. To take into account this phenomenon, we have considered Robin boundary conditions. In principle, since both sides have the same design (number and width of the interface microcavities), there is no reason for considering differences in cell losses (in percentage) between both sides. Therefore, as there is no cell supply through the lateral channels the boundary condition writes:

$$u_1(x^*, t) + J_1 f_1(x^*, t) = 0, \tag{6.60}$$

where $x^* = 0, L$. With regard to the dead cell population, homogeneous Neumann boundary conditions are considered since this population does not migrate either by diffusion or chemotaxis, so we have:

$$u_2(x^*, t) = 0. \tag{6.61}$$

Regarding the oxygen supply, we shall consider two possibilities, associated with two different conditions: when oxygen is supplied normally, Dirichlet boundary conditions are considered, that is, we shall assume that the oxygen concentration at the channels remains constant and known throughout the experiment:

$$u_0(x^*, t) = O_2^*, \tag{6.62}$$

where O_2^* is a known value.

On the other hand, when a channel is sealed, we assume that oxygen provision is negligible, so Neumann boundary conditions are considered:

$$f_0(x^*, t) = 0. \tag{6.63}$$

Finally, we assume that, at time $t = 0$, all cells are alive and the cell population concentration is known throughout the whole culture chamber. That is, $C_1(x, t = 0) = C_1^0(x)$ is known (measured experimentally) and $C_2(x, t = 0) = 0$. Moreover, the oxygen profile is assumed to be constant along the chamber and equal to the concentration in the channels, due to the small characteristic time of oxygen diffusion within the hydrogel compared to the characteristic time of cell processes:

$$u_0(x, t = 0) = O_2^*. \tag{6.64}$$

6.4.2.4 Discretisation parameters

The differential equation (6.46) with boundary conditions (6.49) and initial conditions (6.50) results in a nonlinear parabolic differential equation in time, with only one space dimension. This equation was solved following the approach presented in Appendix B. The domain length (associated with the microfluidic device) and mesh size used for the simulation of each experiment are summarised in Table 6.1.

Experiment	Chamber length $L\,[\mu m]$	Mesh size $\Delta x\,[\mu m]$
Necrotic core formation	2000	3.0
Pseudopalisade formation	916	4.8
Double pseudopalisade formation	2897	12.0

Table 6.1: Domain and mesh size for the different simulations. The mesh is adapted to the domain size and to the measurement resolution.

6.4.3 Model parameters

In this section, we discuss the values used in literature for each of the parameters in our model. We found that many of them are essentially unknown or with high ranges of variation. Our effort goes in the direction of discriminating which works define some of these parameters in similar conditions trying to identify the most likely values within the intervals identified in the literature.

6.4.3.1 Literature review

In the available literature, it is difficult to identify the precise values of such parameters, due to the diversity of models and experimental conditions. Consequently, we include this review clarifying the process for the parameters definition or calculation, often after a reference-crossing process.

Cell diffusion coefficient (D_n).

The cell diffusion coefficient is a parameter related to the undriven cellular motility. Cell motility is frequently evaluated in experimental works from a global point of view, that is, including random motility and hypoxia-induced chemotaxis. In this work, however, both phenomena are taken into account separately so diffusion acts as a pure regularisation term while chemotaxis is the main driving force in cell migration. Therefore, only diffusion coefficients associated with healthy tissues in perfect oxygenation conditions will be taken into account.

According to Tjia and Moghe (2002), this parameter depends on the substrate mechanical properties. For a standard collagen ECM, similar to the culture hydrogel here used, a value of $1 \times 10^{-9}\,\mathrm{cm}^2/\mathrm{s}$ is proposed. Martínez-González et al. (2012) propose a value of $6.6 \times 10^{-12}\,\mathrm{cm}^2/\mathrm{s}$, and in Martínez-González et al. (2015) a value of $5 \times 10^{-10}\,\mathrm{cm}^2/\mathrm{s}$ is assigned, one order of magnitude lower than the mean of the values reported by Rockne et al. (2010) ($5 \times 10^{-9}\,\mathrm{cm}^2/\mathrm{s}$). Wang et al. (2009) discuss this value for different locations in the brain, observing that glioma cells migrate quicker in white matter than in grey matter, highlighting also the important variation of this coefficient with the tumour stage and after chemotherapy and radiotherapy, ranging all values from $3 \times 10^{-7}\,\mathrm{cm}^2/\mathrm{s}$ to $5 \times 10^{-5}\,\mathrm{cm}^2/\mathrm{s}$ (median of $3 \times 10^{-6}\,\mathrm{cm}^2/\mathrm{s}$). Hathout et al. (2016), use values from $5 \times 10^{-7}\,\mathrm{cm}^2/\mathrm{s}$ to $2 \times 10^{-6}\,\mathrm{cm}^2/\mathrm{s}$ during the tumour initial state.

Chemotaxis coefficient (χ).

This coefficient is difficult to estimate when considering chemotaxis as an isolated phenomenon (Agosti et al., 2018a). Ford and Lauffenburger (1991) define $\chi = \chi_0 f(O_2)$

with χ_0 ranging from $1.5 \times 10^{-5}\,\mathrm{cm}^2/\mathrm{s}$ to $7.5 \times 10^{-4}\,\mathrm{cm}^2/\mathrm{s}$ depending on the complex affinity while several expressions are proposed for f. For example, a hyperbolic tangent dependence is presented, based on a probabilistic mechanobiological model for individual bacteria (Rivero et al., 1989).

Many other works define the chemotaxis coefficient with respect to the normalised concentration $\frac{O_2}{O_2^v}$ where O_2^v is the vessel oxygen supply pressure. Agosti et al. (2018a) assume a value of $1.5 \times 10^{-4}\,\mathrm{cm}^2/(\mathrm{mM} \cdot \mathrm{s})$ for an oxygen concentration in vessels O_2^v of 0.07 mM (Vital-Lopez et al., 2011). Therefore, χ is of the order $2 \times 10^{-7}\,\mathrm{cm}^2/(\mathrm{mmHg} \cdot \mathrm{s})$ assuming an oxygen supply in vessels of $40-60$ mmHg (Kimura et al., 1996; Wilson, 2008). With the same conversion between oxygen concentration and pressure, a value between $3 \times 10^{-10}\mathrm{cm}^2/(\mathrm{mmHg} \cdot \mathrm{s})$ and $1 \times 10^{-9}\mathrm{cm}^2/(\mathrm{mmHg} \cdot \mathrm{s})$ is adopted by Agosti et al. (2018b). Finally, Bearer et al. (2009) propose a chemotaxis coefficient of $10^5\,\mu\mathrm{m}^2/\mathrm{d}$, which gives an equivalent value of $2 \times 10^{-10}\mathrm{cm}^2/(\mathrm{mmHg} \cdot \mathrm{s})$.

Hypoxia-induced migration activation threshold (O_2^H).

In our model, hypoxia induced cell migration is relevant only when the oxygen pressure is under a certain threshold O_2^{th}. According to previous works on GBM simulation (Martínez-González et al., 2012, 2015), cells are considered under hypoxia conditions, when the oxygen pressure is under 7 mmHg (approximately $12-18\%$ of the blood vessel oxygen pressure). Agosti et al. (2018a) consider a threshold of $15-50\%$ of blood vessel oxygen pressure and later (Agosti et al., 2018b) a threshold of 30% is used. In the review paper Vaupel et al. (1989), a ratio of $12-25\%$ between healthy and tumorous tissue oxygen pressure is considered.

Growth characteristic time (τ_{gr}).

This is also a very context-dependent parameter, since the cell metabolism highly varies between cell types and individuals. In addition, our proposed logistic model implies that the measured growth time in the particular experimental conditions depends on the cell concentration, and therefore could vary with considered values reported in literature. Nevertheless, some growth characteristic times reported in literature for logistic, exponential or Gompertz growth models are here discussed. Gerlee and Anderson (2007) consider a growth time of 16 h for a cell automaton model, based on a previous work (Calabresi and Schein, 1993). Other authors propose a value of 24 h (Frieboes et al., 2007; Swanson et al., 2000) using an exponential model (so the growth characteristic time is underestimated). A logistic model is used by Agosti et al. (2018a), with a characteristic time between 48 h and 2000 h, closer to the values obtained by Wang et al. (2009) (with median 408 h) and by Rockne et al. (2010) (mean of 450 h, using magnetic resonance imaging (MRI) techniques). In Agosti et al. (2018b) a value of 300 h is proposed based on a Gompertz growth model (Laird, 1964). Among the 36 tumours simulated by Hathout et al. (2016) a range between 240 h and 1200 h was used. Finally, Martínez-González et al. (2015) propose values between 336 h and 576 h using a Fisher-Kolmogorov approximation, and later Martínez-González et al. (2012) consider values between 24 h and 48 h based on experimental studies (Berens and Giese, 1999; Giese et al., 2003; Ke et al., 2000).

In our work, based on the *go-or-growth* assumption, the growth characteristic time is infinite in absence of oxygen and decreases until the oxygen concentration exceeds the hypoxia threshold. Thus, our model captures this variability from hypoxic to normoxic media, where growth is accelerated and therefore characteristic times are smaller.

Cell concentration saturation (c_{sat}).

An important variability is found in the literature when referring to this parameter, with a range that covers several orders of magnitude. For example, Rockne et al. (2010), propose a value of $10^{11}\,\mathrm{cell/cm^3}$ whereas Hathout et al. (2016) use the value of $10^{8}\,\mathrm{cell/cm^3}$ according to previous experimental works (Herculano-Houzel and Lent, 2005).

This parameter depends on the mechanical and structural properties of the medium and on nutrients supply so its variability is natural. In any case, it does not have a major impact in simulations for cell concentrations much lower than the saturation capacity.

Death characteristic time (τ_{d}).

Even in the case where no cell-concentration dependence is considered, death characteristic time also varies between studies since it is directly measured, without considering, for example, oxygenation conditions, as for the growth characteristic time. In the automaton model from Gerlee and Anderson (2007), an average apoptosis probability of 0.18 is obtained, resulting in a death characteristic time of 72 h as proposed by Frieboes et al. (2007). Agosti et al. (2018a) propose values between 160 h and 400 h, and in Agosti et al. (2018b) of 600 h. Finally, other works use two different phenotypes to model the tumour population (normoxic and hypoxic), but they assume that once the cell has arrived to hypoxic conditions, its death characteristic time is fixed. It is assumed as 48 h in Martínez-González et al. (2012) or as 7 d in Martínez-González et al. (2015).

We model death with a sigmoid function, which integrates both death causes: apoptosis, which is mainly stochastically mediated and necrosis, induced by oxygen lack. This model explains better the variability found in literature, ranging from 72 h in anoxia to 600 h in normoxia, via the two parameters regulating cell switch, discussed below.

Anoxia-induced death location parameter (O_2^A).

In many mathematical models it is assumed that the hypoxia threshold, inducing migration or proliferation (and therefore the fundamental parameter explaining the *go-or-grow* dichotomy), and the anoxia threshold (as an indicator of necrosis) are the same (Agosti et al., 2018a,b); whereas other authors distinguish between both phenomena. Martínez-González et al. (2012, 2015) select a value of 0.7 mmHg for the anoxia level, as explained in previous works (Brown and Wilson, 2004), corresponding to approximately $1-2\%$ of vessel oxygen concentration ($40-60$ mmHg). Vital-Lopez et al. (2011) consider that with 15% of normal concentration (12 mmHg in brain (Vaupel et al., 1989)), the death probability has a value of 50%, resulting in a value of 1.8 mmHg.

Anoxia-induced death spread parameter (ΔO_2^A).

This parameter illustrates the variability of the cell death phenomenon. High values of ΔO_2 are related to random death, that is, apoptosis mediated by other effects not considered in this model, whereas low values of ΔO_2^A are related to death dominated by necrosis, i.e. death only occurs when cells are under the anoxia threshold. Martínez-González et al. (2012, 2015) adopt a value of 0.1 mmHg while Vital-Lopez et al. (2011) consider a dimensionless slenderness parameter of $s = 200$ which turns into 3 mmHg when considering our model, thus considering a higher variability in cell death rate.

Oxygen diffusion coefficient (D_{O_2}).

The oxygen diffusion coefficient is classically known to be around $10^{-5}\,\mathrm{cm}^2/\mathrm{s}$ at 37°C. Daşu et al. (2003) propose a value of $2 \times 10^{-5}\,\mathrm{cm}^2/\mathrm{s}$ according to previous studies (Pogue et al., 2001; Tannock, 1972) that assign an intermediate value between oxygen diffusion in water and muscle (Krogh, 1922). Gerlee and Anderson (2007); Grote et al. (1977) use a value of $1.8 \times 10^{-5}\,\mathrm{cm}^2/\mathrm{s}$. Recent computational patient-specific studies (Agosti et al., 2018b) assume a value of $10^{-5}\,\mathrm{cm}^2/\mathrm{s}$. It is important to note that, in the present work, hydrogels used in microfluidic devices try to reproduce soft human tissue, so similar values can be used.

Oxygen consumption rate (α).

The maximum value of α is much debated (Dewhirst et al., 1994; Gullino et al., 1967; Olive et al., 1992; Vaupel et al., 1989) ranging from $2\,\mu\mathrm{L}/(\mathrm{min}\cdot\mathrm{g})$ (Vaupel et al., 1989) to $55\,\mu\mathrm{L}/(\mathrm{min}\cdot\mathrm{g})$ (Gullino et al., 1967). There are several possible explanations for this large range of values reported as explained by Daşu et al. (2003), such as the influence of the tissue metabolic characteristics in the consumption rate, the variations of the temperature and pressure conditions when measuring the oxygen volume or experimental reasons associated with the measuring method. The most often quoted value is 15 mmHg/s for the maximum consumption rate in healthy tissue (Daşu et al., 2003; Tannock, 1972; Thomlinson and Gray, 1955). This consumption rate gives a maximum diffusion distance of 143 μm (Thomlinson and Gray, 1955), for a blood vessel with 40 mmHg. Assuming that a healthy tissue has a concentration of $0.2c_{\mathrm{sat}}$ (Martínez-González et al., 2015), we obtain $7.5 \times 10^{-7}\,(\mathrm{mmHg}\cdot\mathrm{cm}^3)/(\mathrm{cell}\cdot\mathrm{s})$. Assuming the same ambient cell concentration, using the value proposed by Agosti et al. (2018a,b) we obtain $2.5 \times 10^{-7}\,(\mathrm{mmHg}\cdot\mathrm{cm}^3)/(\mathrm{cell}\cdot\mathrm{s})$. The consumption selected for the automaton presented by Gerlee and Anderson (2007), based on studies on GBM spheroids (Freyer and Sutherland, 1986), is fixed to $2.3 \times 10^{-16}\,\mathrm{mol}/(\mathrm{cell}\cdot\mathrm{s})$, equivalent to $1.4 \times 10^{-8}\,(\mathrm{mmHg}\cdot\mathrm{cm}^2)/(\mathrm{cell}\cdot\mathrm{s})$ assuming an oxygen background concentration of $c_0 = 1.7\times10^{-8}\,\mathrm{mol}/\mathrm{cm}^2$. A value of $4\times10^{-17}\,\mathrm{mol}/(\mathrm{cell}\cdot\mathrm{s})$ is obtained from the data analysed by Griguer et al. (2008) resulting in $2.5 \times 10^{-9}\,(\mathrm{mmHg}\cdot\mathrm{cm}^2)/(\mathrm{cell}\cdot\mathrm{s})$.

Michaelis-Menten constant (k_{m}).

According to Daşu et al. (2003), the k_{m} constant seems to have little influence on the diffusion at high O_2 concentrations and therefore we use a value of $k_{\mathrm{m}} = 2.5$ mmHg, equal to the hypoxic threshold often used in the Alper and Howard-Flanders equation

describing the oxygen enhancement ratio (Alper and Howard-Flanders, 1956). This value has been chosen in recent simulation models (Martínez-González et al., 2012, 2015).

In Table 6.2 all numerical parameters of the mathematical model are shown with their corresponding variation range extracted from the bibliography.

	Minimal value	**Maximal value**	**Units**
D_n	6.6×10^{-12} (Martínez-González et al., 2012)	5.0×10^{-5} (Wang et al., 2009))	cm^2/s
c_{sat}	1.0×10^{8} (Hathout et al., 2016)	1.0×10^{11} (Rockne et al., 2010)	cell/mL
χ	2.0×10^{-10} (Bearer et al., 2009)	7.5×10^{-4} (Ford and Lauffenburger, 1991)	$cm^2/mmHg \cdot s$
τ_{gr}	5.8×10^{4} (Gerlee and Anderson, 2007)	7.2×10^{6} (Agosti et al., 2018a))	s
τ_d	1.7×10^{5} (Martínez-González et al., 2012)	2.2×10^{6} (Agosti et al., 2018b)	s
D_{O_2}	1.0×10^{-5} (Agosti et al., 2018b)	2.0×10^{-5} (Daşu et al., 2003))	cm^2/s
α	2.5×10^{-9} (Griguer et al., 2008)	7.5×10^{-7} (Martínez-González et al., 2015)	$mmHg \cdot mL/cell \cdot s$
O_2^T	2.5 (Daşu et al., 2003)	2.5 (Daşu et al., 2003)	mmHg
O_2^H	7.0 (Vaupel et al., 1989)	30 (Agosti et al., 2018a)	mmHg
O_2^A	0.7 (Brown and Wilson, 2004)	1.8 (Vital-Lopez et al., 2011)	mmHg
ΔO_2^A	0.1 (Martínez-González et al., 2012)	3.0 (Vital-Lopez et al., 2011)	mmHg

Table 6.2: Range of variability of the parameters in the bibliography. Note that in some cases the range includes many orders of magnitude.

6.4.3.2 Model parameter fitting

The value of each parameter was initially chosen to stay within the ranges found in the bibliography (Table 6.2). In order to calibrate the specific value for each parameter, the formation of the necrotic core experiment was selected as case study. Robin boundary conditions were chosen allowing alive cells to escape from the device. In particular, for the boundary conditions given by Eq. (6.49), according to the experimental results and the symmetry of the experiment, we select $K_1(x^* = 0, L, t) = 1$, $g_1(x^* = 0, L, t) = h_1(x^* = 0, L, t) = 0$ and $J_1(x^* = 0, L, t) = 1.0 \times 10^6\,s/cm$.

Only the parameter J_1, explaining cell leakage at boundaries, was fitted to capture the corresponding results. For dead cells, homogeneous Neumann conditions were established, assuming no migration of dead cells through the boundaries, that is, $K_2(x^* = 0, L, t) = 0$, $h_2(x^* = 0, L, t) = 0$ and $J_2(x^* = 0, L, t) = 1.0\,s/cm$. With respect to oxygen concentration, Dirichlet boundary conditions were chosen, so oxygen pressure in the channels was assumed to remain constant throughout the whole experiment, since medium flow provision through the channels was sufficiently frequent to keep that pressure without important variations despite oxygen diffusion and cell uptake. With that, we write $K_0(x^* = 0, L, t) = 1$, $g_0(x^* = 0, L, t) = 7.0\,mmHg$ and $J_0(x^* = 0, L, t) = 0$. Finally, as initial conditions, we assume that the initial monitoring of the process starts after getting a uniform oxygen pressure in the whole chamber, equivalent to the one present in the lateral channels, that is $O_2(t = 0) = 7\,mmHg$.

A heuristic approach was followed to fit the simulated curves with the experimental results, in order to determine the model parameters. This approach tried to get the best fit of the necrotic core (central region) due to its biological relevance, giving less importance to the fitting around the boundaries since the cell distribution here

is not representative of the *in vivo* situation. To quantify the quality of this fitting procedure, we defined two objective cost functions:

$$T = \frac{1}{c_{\text{sat}}} \sqrt{\frac{1}{2L} \sum_{j=1}^{2} \sum_{i=2}^{5} \int_0^L \left(u_j(x,t_i) - u_j^e(x,t_i)\right)^2 dx}, \tag{6.65}$$

$$D = \frac{1}{c_{\text{sat}}} \max_{i=2,\dots,5} \max_{x\in[0;L]} |u_j(x,t_i) - u_j^e(x,t_i)|, \tag{6.66}$$

where u_j^e is the experimental measurement of the j phenotype ($j = 1$, alive cells, $j = 2$, dead cells), u_j the simulated one and L the chip length.

After the fitting process for the different results obtained in the necrotic core experiment, we arrived to the value set for the parameters shown in Table 6.3, yielding the results shown in Fig. 6.7. The fitting process provided values of $T = 0.17$ and $D = 0.73$, which reinforce the good agreement between the experimental and simulation results.

Symbol	Fitted value	Units
D_n	6.6×10^{-10}	cm^2/s
c_{sat}	5.0×10^{7}*	cell/mL
χ	7.5×10^{-9}	$\text{cm}^2/\text{mmHg} \cdot \text{s}$
τ_{gr}	7.2×10^{5}	s
τ_{d}	1.7×10^{5}	s
D_{O_2}	1.0×10^{-5}	cm^2/s
α	1.0×10^{-9}	$\text{mmHg} \cdot \text{mL}/\text{cell} \cdot \text{s}$
k_{m}	2.5	mmHg
O_2^H	7	mmHg
O_2^A	1.6	mmHg
ΔO_2^A	0.1	mmHg
J_1	1.0×10^{6}	s/cm

Table 6.3: Final parameter ranges. A star means that the fitted value is out of the range found in the literature. In the last line, we include the parameter J_1, which is not a model parameter, but a parameter related to experimental operation, since it is also adjusted.

As it may be observed, the computed results are qualitatively equal and quantitatively very similar to the experimental ones, although there are some significant discrepancies in the alive cell profile, mainly at the centre of the chamber, and in the dead cell profile at the boundaries. These differences are unavoidable due to the effects and interactions missed in the model, such as the heterogeneous distribution of hydrogel, cells and oxygen in the initial state and boundary conditions, and to the highly non-linear character of the equations. As it can be seen in Table 6.2, the range of variation of the parameters in the scientific literature is extremely wide in some cases, being therefore tricky to tune the value of the parameters to obtain better numerical results. All this makes it essential to perform a sensitivity analysis to understand the quantitative impact of each parameter on the representative results and to assess the mathematical model robustness with respect to the parameter fitting.

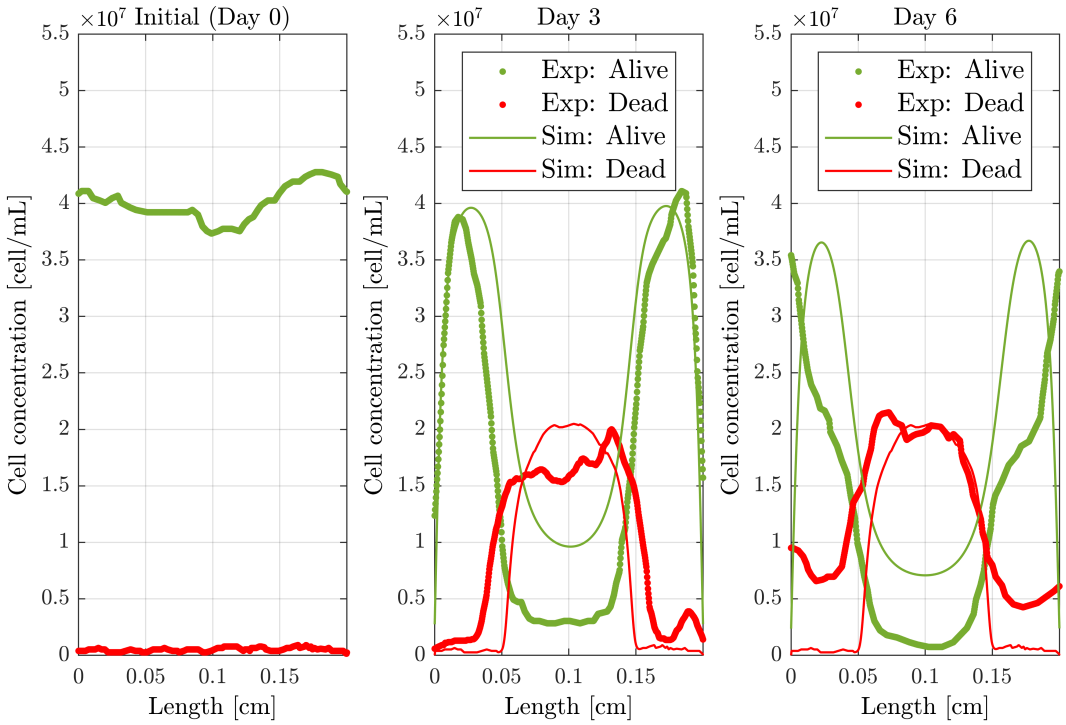


Figure 6.7: Cell concentration profiles for the defined value set. Dead and alive profiles (y-axis) along the length of the chip (x-axis) with the parameters shown in Table 6.2. Sim: Simulated profiles. Exp: Experimental profiles. On day 0, both profiles coincide.

6.4.4 Stochastic parametric analysis

6.4.4.1 Model parameters

Our aim now is to particularise the presented framework to our GBM evolution model described in Section 6.4.2. As we have discussed, one of the main problems in mathematical modelling is the lack of reliable values for the many parameters involved that forces many times to rely on values fitted from different situations, leading sometimes to unreliable conclusions. Although our presented model enables the simulation of different stages of GBM evolution under several experimental conditions, showing robustness, while keeping a small uncertainty range in the results, it is possible to better understand its weaknesses and strengths by performing a deeper parametric analysis using the statistical and probabilistic tools described in Section 6.3.

As our main interest is the *go-or-grow* switch, we will focus our analysis in three model parameters:

- The chemotaxis coefficient χ.
- The growth rate $\alpha_1 = 1/\tau_{\text{gr}}$.
- The hypoxic threshold O_2^H.

The rest of the parameters are considered as known and equal to their adjusted value given in Table 6.3.

Besides, even if we could consider several experimental configurations, due to the available data we consider only one, which is the necrotic core formation experiment. Therefore we will have the particular case $\boldsymbol{\lambda}^i = \boldsymbol{\lambda}$ where $\boldsymbol{\lambda}$ encodes all the parameters (oxygen supply and initial cell concentration profile) associated with the necrotic core experimental configuration.

In summary, using the framework presented in Section 6.3, we have defined:

- The model parameters, $\boldsymbol{\theta} = (\chi, \alpha_1, O_2^H) \in \mathbb{R}^3$.
- The experimental variable, $\boldsymbol{\lambda}$, that in our case is related to boundary and initial conditions.
- The output variable $\boldsymbol{u}$, that is the outcome of the experiment, that is, the measurement of the cell concentrations profiles.
- The mathematical model $\boldsymbol{F}$, formed by the evolution PDEs (6.46) with boundary conditions (6.49) and initial conditions (6.50).

As said, we have restricted the analysis to three model parameters, being the rest of them assumed to be known and constant for the sake of simplicity. For a more general analysis, considering more parameters and different experimental configurations, the reader is invited to consult Ayensa-Jiménez et al. (2021b).

6.4.4.2 Data generation

Since we have dispensed with a source of variability (the variation of the experimental configuration), a set of $M = 6000$ simulations were performed varying the parameters within their order of magnitude. Each parameter, say θ, was individually perturbed as follows: if $\mathcal{N}(\mu, \sigma)$ is a normal random variable with mean μ and standard deviation σ, we generated M samples of $\Theta \sim \mathcal{N}(\theta^*, \theta^*/2)$ (that is, 50% of relative uncertainty). The non-perturbed value of θ^* corresponds to the value obtained after the fitting procedure. The vector parameters that provide solutions with $T < 0.18$ were kept, where T is defined by Eq. (6.65). In other words, our data-set is defined by:

$$\mathcal{D} = \{(\chi^i, \alpha_1^i, (O_2^H)^i),\, i = 1, \ldots, M \,|\, T(\chi^i, \alpha_1^i, (O_2^H)^i) \leq 0.18\}. \tag{6.67}$$

In Fig. (6.8), the data-set $\mathcal{D}$ generated is illustrated together with the nominal value obtained using the fitting procedure (that we will name as nominal value), that is $\chi = 7.5 \times 10^{-9}\,\mathrm{cm}^2/(\mathrm{mmHg}\cdot\mathrm{s})$, $\alpha_1 = 1.4 \times 10^{-6}\,\mathrm{s}^{-1}$ and $O_2^H = 7\,\mathrm{mmHg}$.

The data-set $\mathcal{D}$ was split in a train data-set ($M_{\text{train}} = 2244$) and a test data-set ($M_{\text{test}} = 440$), $\mathcal{D} = \mathcal{D}_{\text{train}} \sqcup \mathcal{D}_{\text{test}}$.

6.4.4.3 Statistical fitting.

Marginal distributions.

First of all, we obtained the fitting of the univariate marginal distributions. Fig. 6.9 shows the kernel estimation of the marginal distribution of the different parameters. We chose a Gaussian kernel for all the estimations with variable bandwidths (reported in the figure). The values are generally concentrated around the nominal value, although the distributions present a variable uncertainty, related to the model complexity and its sensitivity.

Parametric *copula* structure.

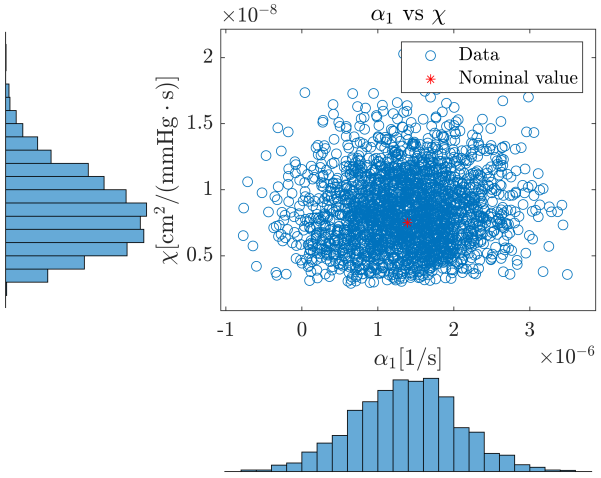
The data were transformed into uniformly distributed values using the cumulative distribution function (CDF) associated with this kernel estimation and a t-Student *copula* is fitted by means of maximum likelihood estimation (MLE). The use of a t-Student *copula* is justified as it allows a different structural dependence for each of the variable pairs considered (Kole et al., 2007) and, besides, it outperforms Gaussian *copula* when estimating the co-occurrence of extreme events Demarta and McNeil (2005). We obtained a *copula* with $\nu = 1.5 \times 10^7$ degrees of freedom and a Pearson correlation matrix of:

$$\boldsymbol{P} = \begin{bmatrix} 1.00 & 0.07 & 0.03 \\ 0.07 & 1.00 & -0.25 \\ 0.03 & -0.25 & 1.00 \end{bmatrix} \tag{6.68}$$

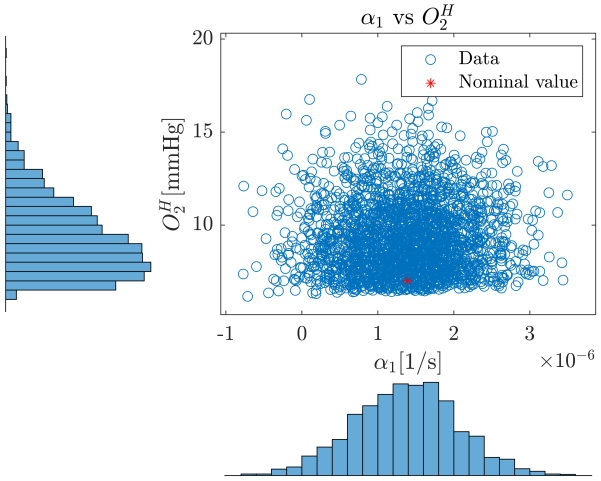
Note that the value obtained for ν is close to the Gaussian limit ($\nu \to \infty$), so the Gaussian copula could have been used instead of a t-Student model.

6.4.4.4 Validation of the results using test data.

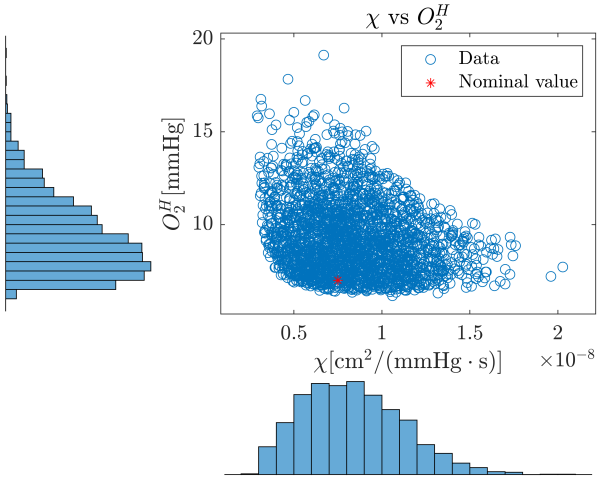
Overfitting is one of the main problems in any statistical or numerical parametric fitting. In our methodology, this was avoided by using a sub-set of the data as test data for validating the models.



(a) Plane (α_1, χ).

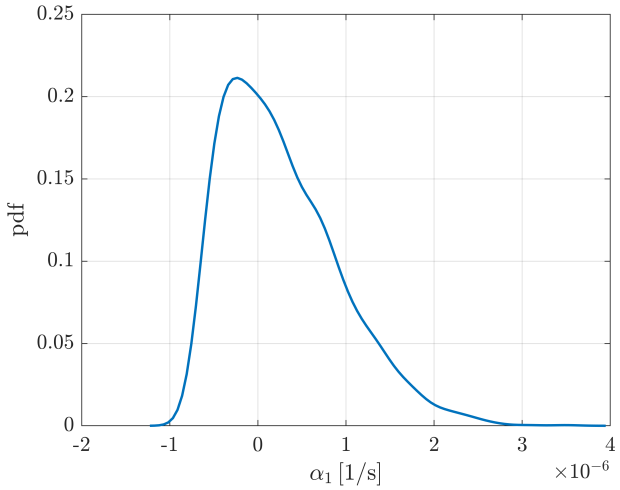


(b) Plane (α_1, O_2^H).

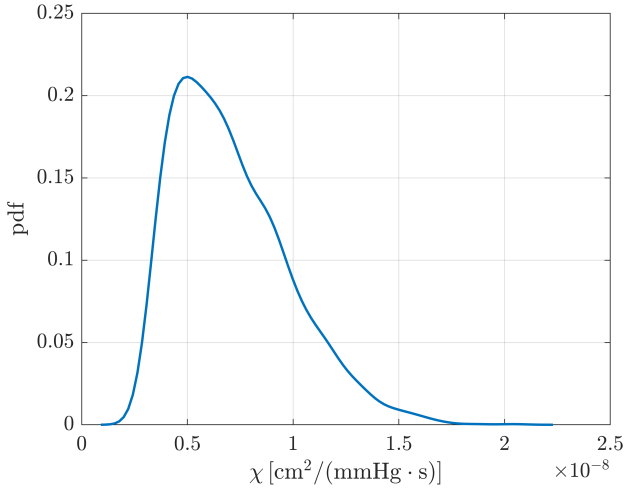


(c) Plane (χ, O_2^H).

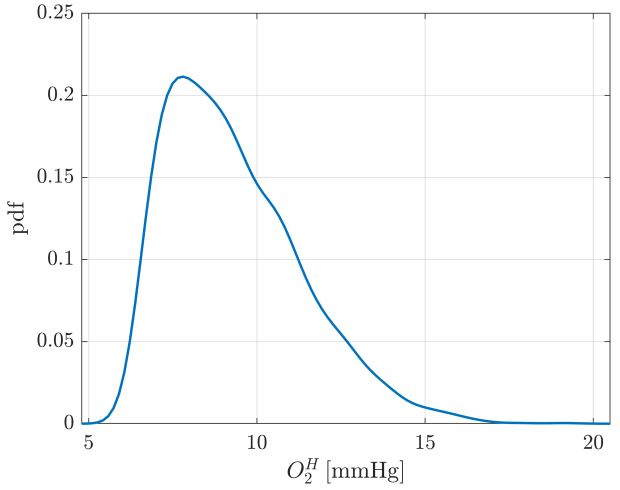
Figure 6.8: Data-set used for the parametric analysis. Data-set generated and nominal value are shown.



(a) α_1, $(w = 1.5 \times 10^{-7}\text{s}^{-}1)$



(b) χ, $(w = 6.7 \times 10^{-10}\,\text{cm}^2/\text{mmHg} \cdot \text{s})$

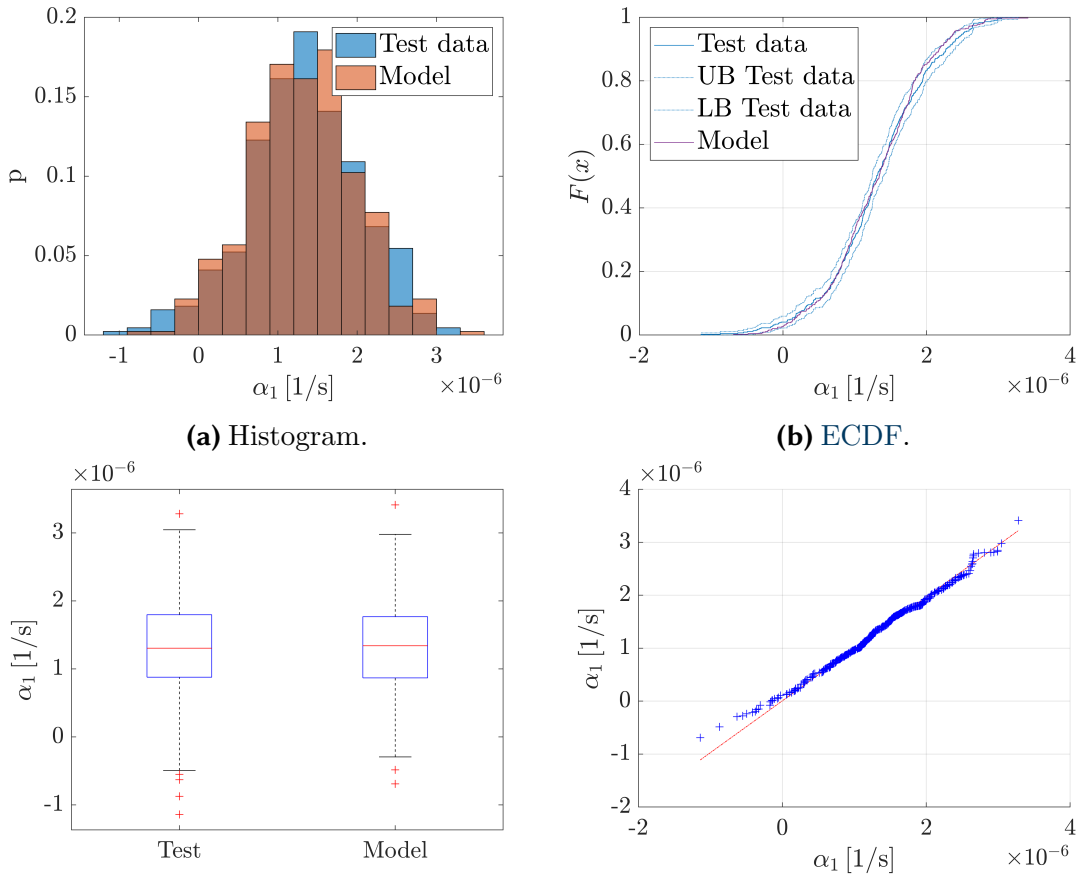


(c) O_2^H, $(w = 4.6 \times 10^{-1}\text{mmHg})$

Figure 6.9: Kernel density estimation of the marginal distributions. The bandwidths used for kernel estimation are indicated for each density estimation.

Marginal distributions.

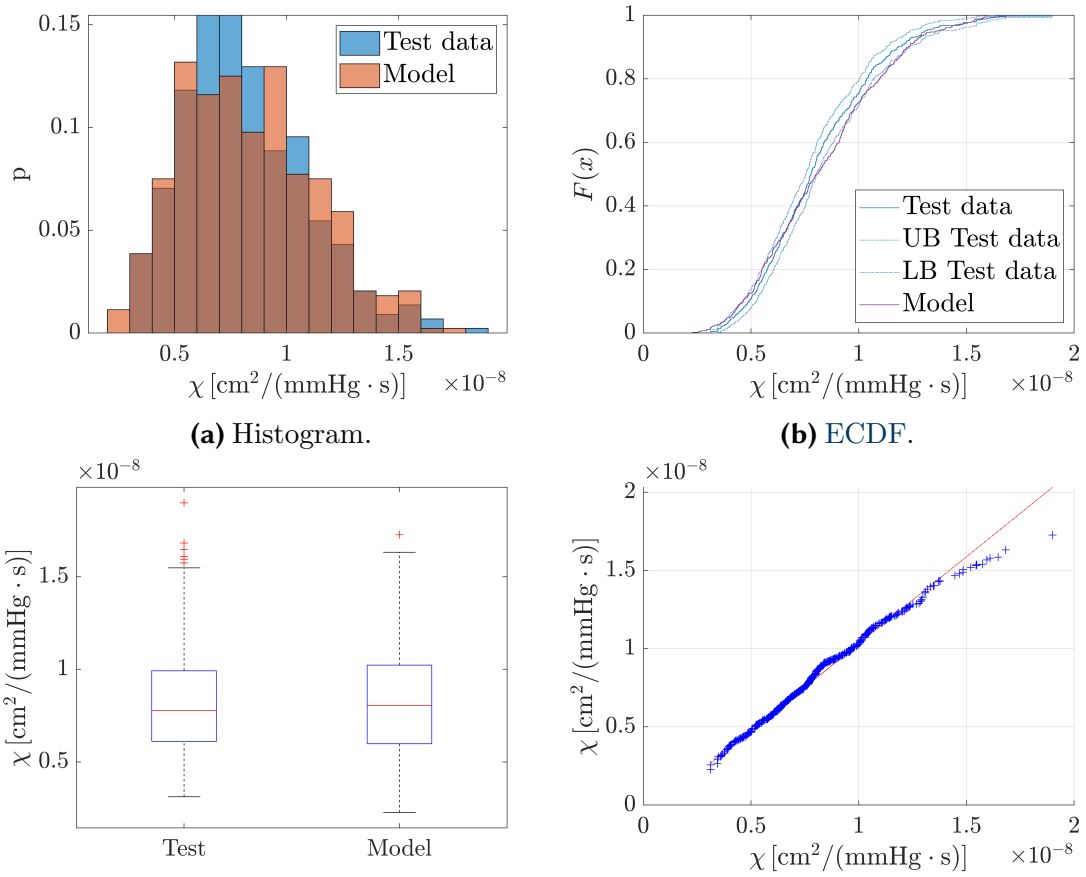
Marginal distributions were validated as pointed out in Section 6.3.2.2. To do so, new "experimental" data were compared to the data generated from the multivariate model. It is important to note that the original data were not used, but, on the contrary, a new data-set was strictly generated from the parametric *copula* and marginal densities, using the same procedure described for the generation of the original data. The histogram of data, the empirical cumulative distribution function (ECDF) of the test data (with 95% confident interval) compared to the model data, the boxplot of both test and model data and the Q-Q plot of the test data, when compared to the model, are shown in Fig. 6.10 for α_1, Fig. 6.11 for χ and Fig. 6.12 for O_2^H. The validation of the whole set of variables has been performed and good agreement was found between the model and test data except, if at all, for the extreme values, at the tail values of the distributions.



(a) Histogram. **(b)** ECDF.

(c) Boxplot. **(d)** Q-Q plot.

Figure 6.10: Comparison between train and test marginals for α_1. The test data show good agreement with the model marginals.



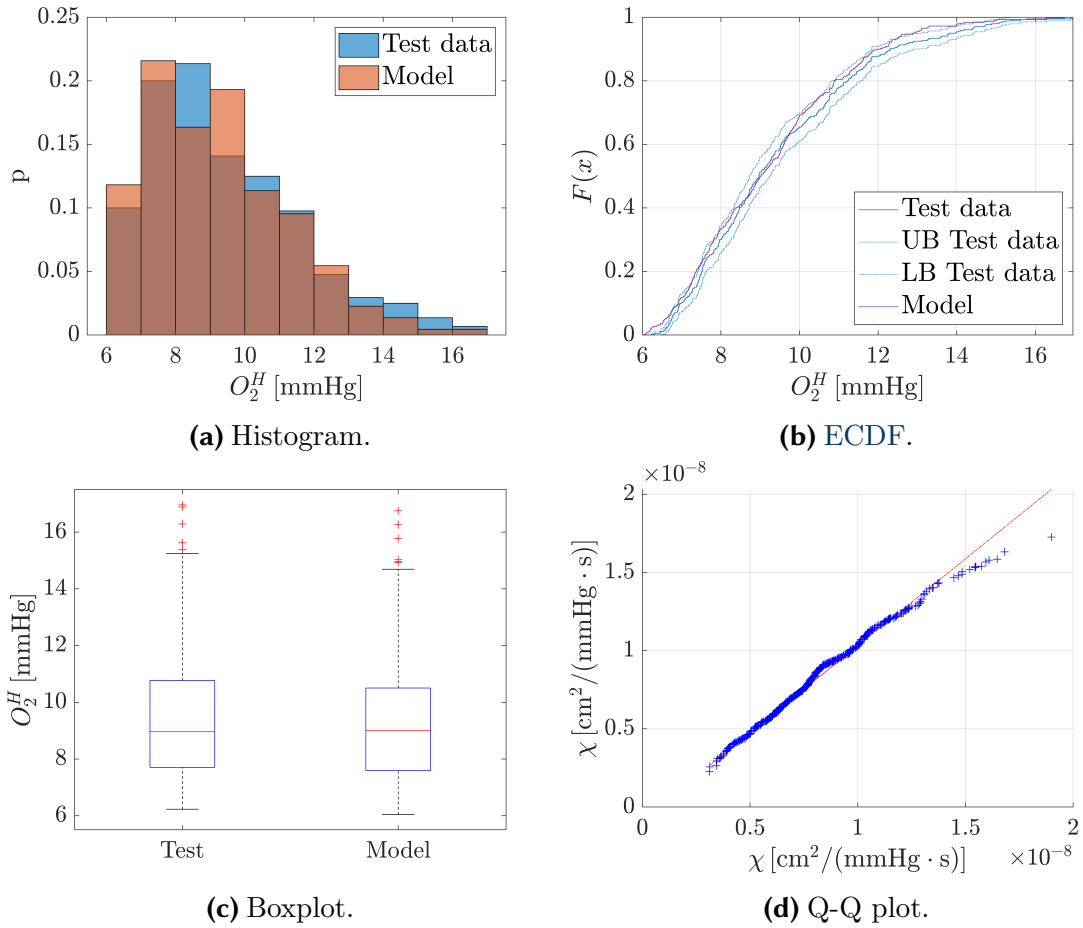
(a) Histogram.

(b) ECDF.

(c) Boxplot.

(d) Q-Q plot.

Figure 6.11: Comparison between train and test marginals for χ. The test data show good agreement with the model marginal, except, if at all, for the extreme values, at the tail of the distributions.



(a) Histogram. **(b)** ECDF. **(c)** Boxplot. **(d)** Q-Q plot.

Figure 6.12: Comparison between train and test marginals for O_2^H. The test data show good agreement with the model marginals.

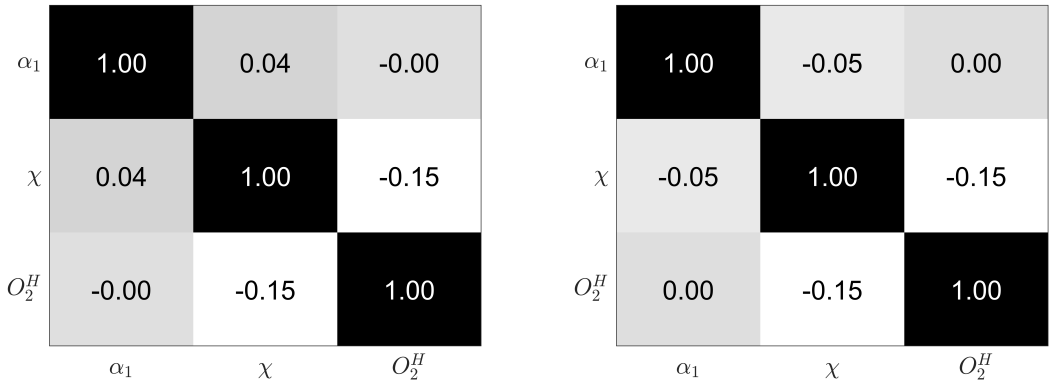
Joint dependencies.

Testing the structural dependence between parameters is not trivial. First, we evaluated merely the differences in the correlation coefficients between the model-based and the test data. In Fig. 6.13, we represent the Kendall τ correlation index between the variables for the model and test data. We observe again a good agreement between the model values of the correlation coefficients (Fig. 6.13a) and those obtained from the sample of the test data (Fig. 6.13b), even though the test sample is finite, which can cause differences between the model and the statistical values.

A deeper analysis may be done by using multivariate goodness of fit. For instance, for checking the nonlinear structural dependence, we analysed the dependence between the uniform variables $U_1 = F_{\alpha_1}(X_1)$, $U_2 = F_{\chi}(X_2)$ and $U_3 = F_{O_2^H}(X_3)$,, where F_{α_1},F_{χ} and $F_{O_2^H}$ are the CDF of the parameters α_1, χ and O_2^H respectively (that had been already validated when testing for the marginals). Given a partition of the unit cube in m^3 regions, we define:

$$D = \sum_{i,j,k}^{m} \frac{(O_{ijk} - E_{ijk} - 0.5)^2}{E_{ijk}}, \tag{6.69}$$

where O_{ijk} and E_{ijk} are the observed and expected values at the regions $i-j-k$. The -0.5 in the numerator of Eq. (6.69) accounts for Yates correction (Adler, 1951). If we choose a significant level α, we may reject the null hypothesis if $D \geq \chi^2_{\nu,1-\alpha}$. with $\chi^2_{\nu,1-\alpha}$ the critical value of the χ^2-distribution of $\nu = m^3$ degrees of freedom corresponding to the α significance level. In our case, selecting $m = 3$, that is, a $3 \times 3 \times 3$ grid, $\chi^2_{27,0.95} = 40$ and $D = 39$, so we cannot reject the null hypothesis (p-value of $p_{\text{val}} = 0.069$).



(a) Kendall τ for the model data.

(b) Kendall τ for the test data.

Figure 6.13: Comparison between train and test correlations. Kendall τ correlation coefficient for each pair of variables for the training and test data is shown, illustrating a good agreement.

6.4.4.5 Complete probabilistic model and Bayesian *a posteriori* corrections.

In order to briefly analyse the aspect of the whole model, in Fig. 6.14 some isosurfaces of the 3D PDF at the space (α_1, χ, O_2^H) are shown.

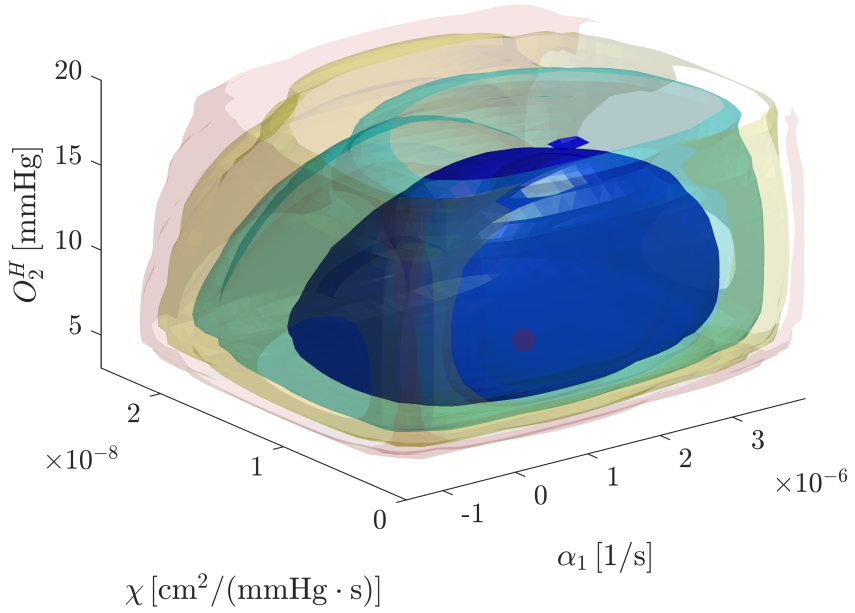
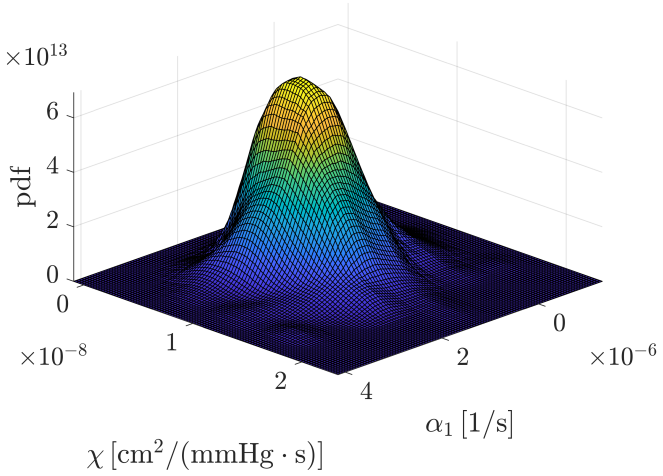
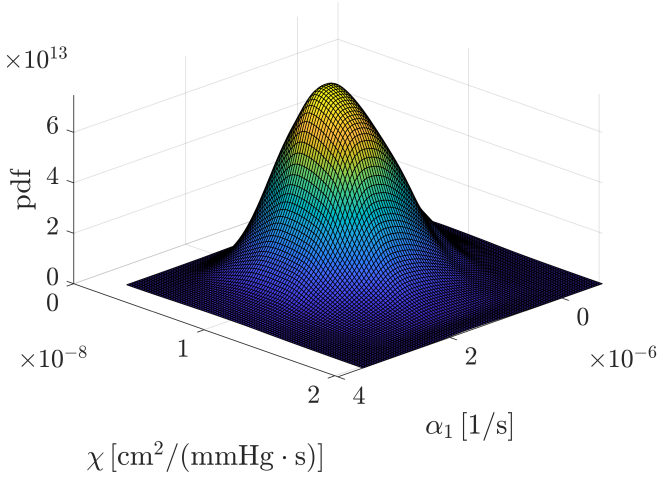


Figure 6.14: Isosurfaces of the 3D probability density function. The isosurfaces have been represented log-spaced between the maximal and minimal value of the density at the represented region. The red dot is the nominal value of the parameters.

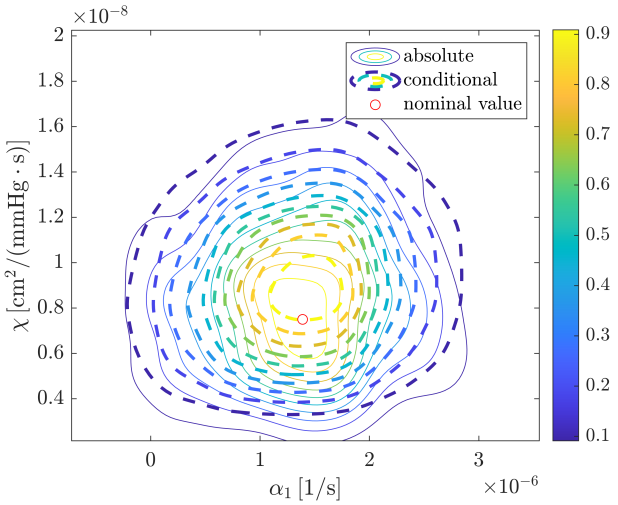
We illustrate the model flexibility in Fig (6.15). Fig. 6.15a represents the bivariate joint distribution of (α_1, χ). Knowing the whole joint distribution function allows us to make *a posteriori* corrections using Bayesian theory and conditional probability as explained in Section 6.3.2.3. If we are interested in the joint distribution of (α_1, χ), assuming that we know the value of $O_2^H = 7\,\text{mmHg}$, the uncertainty of the parameter estimation obviously changes, as can be seen in Fig. 6.15b. In order to compare the impact of setting the rest of the parameters *a posteriori*, contour plots of both distributions, absolute and conditional (normalised between 0 and 1 to compare them more easily) are depicted in Fig. 6.15c.



(a) Bivariate joint distribution of (α_1, χ).



(b) Bivariate joint distribution of (α_1, χ) assuming $O_2^H = 7\,\text{mmHg}$.

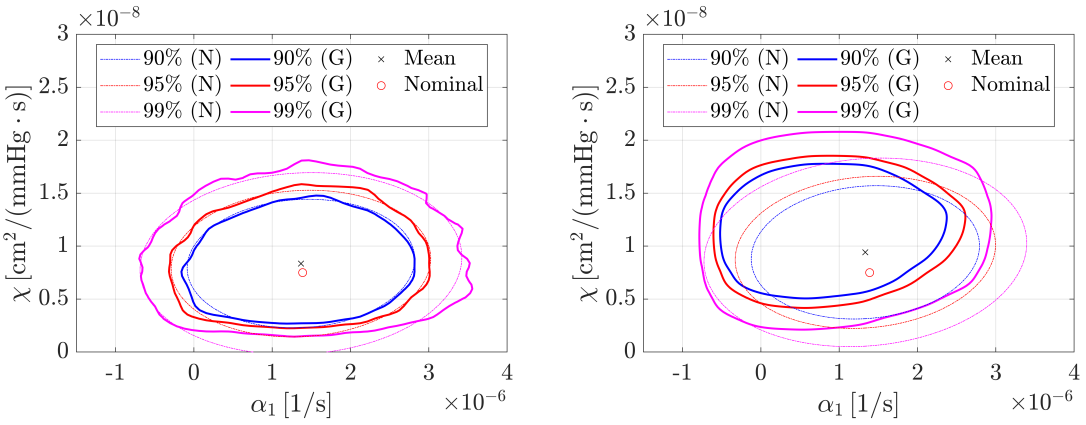


(c) Comparison between the distribution shape of (a) and (b).

Figure 6.15: Bivariate joint distribution functions of (α_1, χ)**.** We show the total joint distribution, one possible conditional distribution and a comparison between both of them.

6.4.4.6 Parameter estimation

In Figs. 6.16, 6.17 and 6.18 we show the p-confident HDR for the model parameters, when $p = 0.90$, $p = 0.95$ and $p = 0.99$. For illustration purposes, we have represented the parameter predictions at each plane (α_1, χ), (α_1, O_2^H) and (χ, O_2^H), showing the effect of conditioning the distribution to the knowledge of the third model parameter. The results are compared with the classical ellipsoid estimation, which is based on the normality assumption. The differences, both in shape and size of the regions, are clear and explained by the complex structural dependence between variables. It is also important to note the high uncertainty related to the hypoxic threshold O_2^H: with the available data, as the cell culture is always under hypoxic conditions, it is difficult to estimate this threshold accurately. In order to achieve better predictions, we would need experiments with higher oxygen levels or to deal with the *go-or-grow* model more carefully.



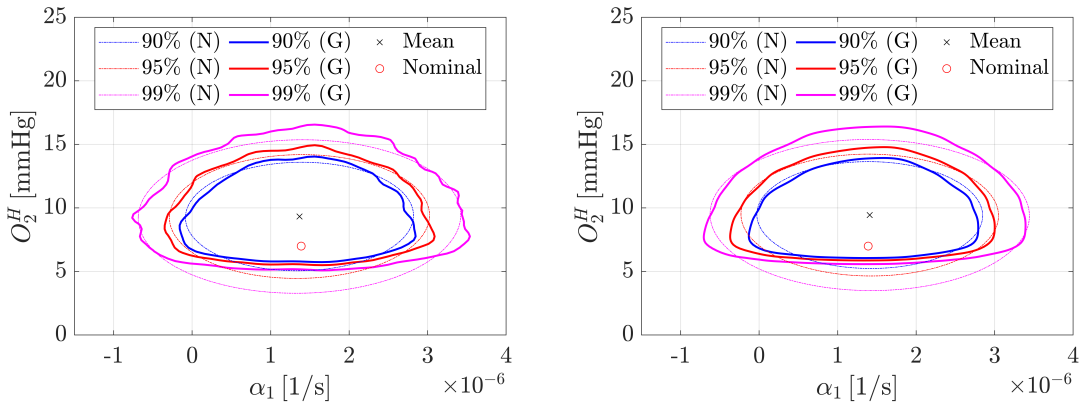
(a) Total distribution. **(b)** Conditional distribution.

Figure 6.16: α_1 - χ mean and HDR estimations. (N): Ellipsoid estimation using normality assumption, (G): General estimation using the *copula* model, Mean: Mean value estimation, Nominal: nominal value. Note that the normality assumption is not always fulfilled and that conditioning has a significant effect.

6.4.4.7 Design of experiments

For the DoE, we assume that we have a microfluidic device of legth $L = 1000\,\mu$m, and we measure the alive cell concentration at 5 given points: $u^k = u_1(x = x_k)$, $k = 1, \ldots, 5$, where $x_1 = 0.015\,$cm, $x_2 = 0.035\,$cm, $x_3 = 0.050\,$cm, $x_4 = 0.065\,$cm, $x_5 = 0.085\,$cm. We work under the homoscedasticity and independence assumptions so that, each concentration measurement is assumed to be normally distributed with $\mu^i = u^i$ and $\sigma^i = \sigma$, $i = 1, \ldots, 5$. Additionally, the uncertainty associated with the measurement of the cell concentration is assumed to be $\sigma = 1 \times 10^6\,$cell/mL.

As we work under the assumptions detailed above, it is possible to use Eq. (6.45) for estimating the utility of an experimental configuration $\boldsymbol{\lambda}$ via numerical integration. For that case we consider as the parameters of the experimental design the oxygen level at the right and left chamber, that is $\boldsymbol{\lambda} = (O_2^{\text{left}}, O_2^{\text{right}})$, where O_2^{left} plays the role of O_2^* at $x^* = 0$ and O_2^{right} plays the role of O_2^* at $x^* = L$ in Eq. (6.62). A convergence analysis was performed, justifying the use of a value of $N_\theta = 10$ (number of sampling



(a) Total distribution. **(b)** Conditional distribution.

Figure 6.17: α_1 - O_2^H mean and HDR estimations. (N): Ellipsoid estimation using normality assumption, (G): General estimation using the *copula* model, Mean: Mean value estimation, Nominal: nominal value. Note that the normality assumption is not always fulfilled and that in that case, conditioning has not a very significant effect.

points for the model parameters) and $N_u = 18$ (number of sampling points for the experimental outcome) for each computation in the numerical integration process[22].

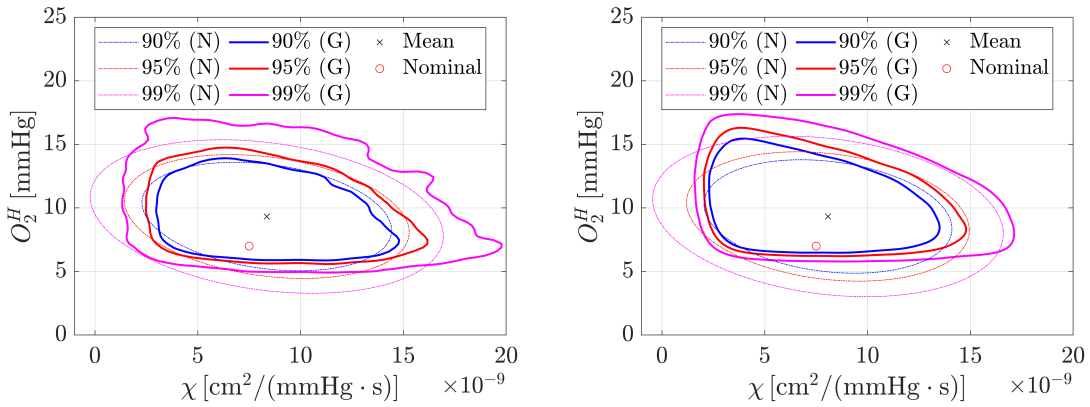
The simulations were performed for ten different oxygen levels at each side of the chip, O_2^{left} and O_2^{right} (from 0 to 9 mmHg) and $C_0 = 10 \times 10^6$ cell/mL. For a richer approach involving more design parameters, the reader may consult Ayensa-Jiménez et al. (2021b). In order to avoid numerical problems, in all simulations, the uniform distributions of the parameters were sampled from $\epsilon = 0.01$ to $1 - \epsilon = 0.99$.

The aim is to determine the experimental configuration with the highest utility, that is, to choose both right and left oxygen flow levels to get the maximum possible information from the new experiment. We focus here on the effect of coupling between parameters and how it affects the utility interpretation and model parameter estimation. For instance, first, let us consider that we want to determine the value of hypoxic threshold O_2^H. In Fig. 6.19 the utility contour plots for different situations are shown, including:

- None of the parameters is known.
- We know the value of the parameter $\alpha_1 = 1.4 \times 10^{-6}\text{s}^{-1}$.
- We know the value of the parameter $\chi = 7.5 \times 10^{-9}\text{cm}^2/(\text{mmHg} \cdot \text{s})$.
- We know the value of the pair of parameters $\alpha_1 = 1.4 \times 10^{-6}\text{s}^{-1}$ and $\chi = 7.5 \times 10^{-9}\text{cm}^2/(\text{mmHg} \cdot \text{s})$.

In Fig. 6.20 we repeat the analysis but focusing on the characterisation of the pair of parameters (α_1, χ). In that case, the two possible situations are:

[22] It is important to note that, with these computations, we do not pretend to compute the exact utility value, but a value that allows comparing the different utility values in a relative way



(a) Total distribution. **(b)** Conditional distribution.

Figure 6.18: χ - O_2^H mean and HDR estimations. (N): Ellipsoid estimation using normality assumption, (G): General estimation using the *copula* model, Mean: Mean value estimation, Nominal: nominal value. Note that the normality assumption is not always fulfilled and that the (nonlinear) correlation between parameters is significant.

- None of the parameters is known.
- We know the value of the parameter $O_2^H = 7\,\text{mmHg}$.

Finally, in Fig. 6.21 we consider the characterisation of the whole triplet (α_1, χ, O_2^H). Note that each of the considered situations corresponds to the partial knowledge of some of the relevant biological features of the cell culture, which may have been determined by other experimental set-ups or test types.

In these figures, we can see the most useful experiments (those configurations corresponding to the highest utility values) and those that lead to a poor adjustment of the model parameters.

This analysis may be performed for different parameter combinations, and for different degrees of knowledge. Tables 6.4, 6.5 and 6.6 summarise all possibilities when exploring the *go-or-grow* behaviour. The cases illustrated in this chapter are reported in the third column.

6.4.5 *In silico* replication of the *in vitro* experiments

We analyse the performance of the parametric model presented when using the set of parameters given in Table 6.3, applied to the three experiments described in Section 6.4.1: formation of a necrotic core in high concentrated cultures, pseudopalisade formation due to oxygen gradient and double pseudopalisade formation in a symmetric configuration. Note that only the necrotic core experiment was used for parameter fitting, so this test is essential to evaluate the ability of the model to generalise. As in our stochastic approach, there is an inherent uncertainty in parameter identification, so a run of $N = 100$ Montecarlo simulations was performed for varying values of the model parameters according to the variability and structural dependence discussed in the previous section.

Parameters to be estimated	Known parameters	Figure
α_1	None	-
α_1	O_2^H	-
α_1	χ	-
α_1	χ, O_2^H	-
χ	None	-
χ	α_1	-
χ	O_2^H	-
χ	α_1, O_2^H	-
O_2^H	None	6.19a
O_2^H	α_1	6.19b
O_2^H	χ	6.19c
O_2^H	α_1, χ	6.19d

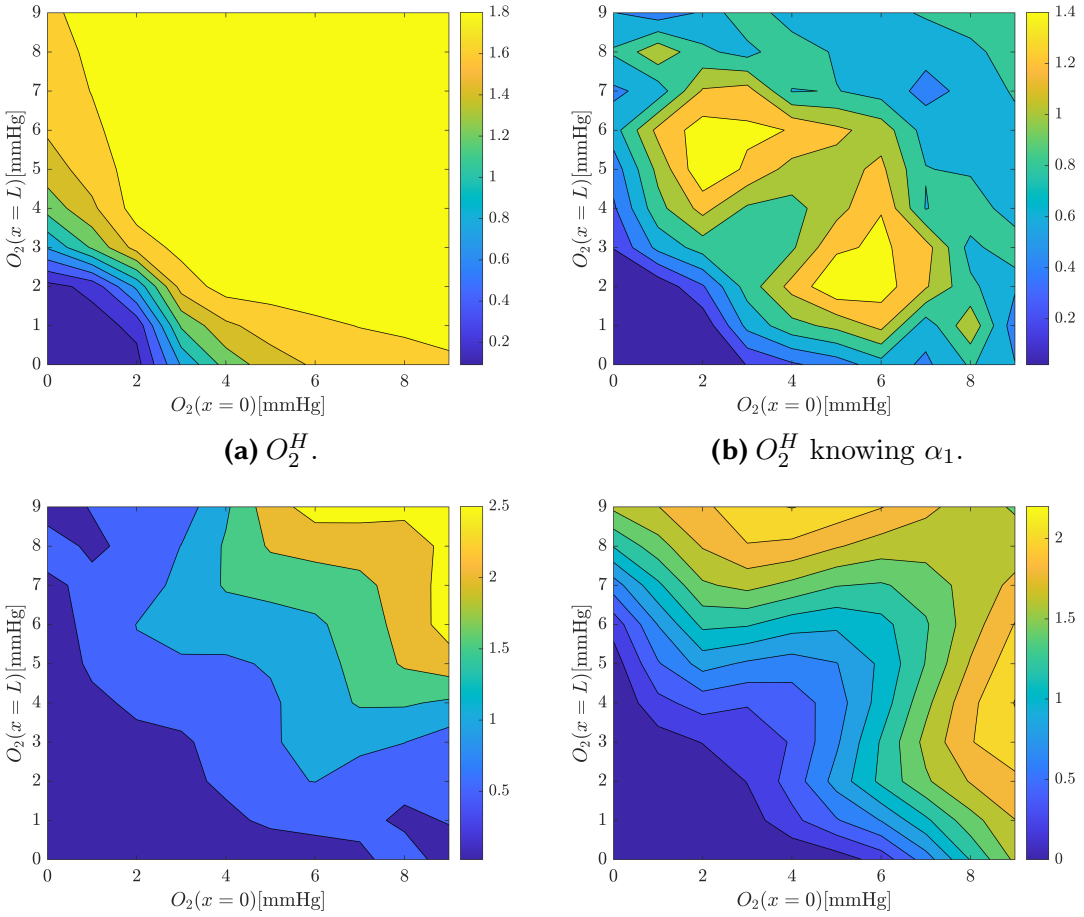
Table 6.4: Possibilities for characterising one parameter. The figures corresponding to the 4 cases computed are indicated.

Parameters to be estimated	Known parameters	Figure
α_1, χ	None	6.20a
α_1, χ	O_2^H	6.20b
α_1, O_2^H	None	-
α_1, O_2^H	χ	-
χ, O_2^H	None	-
χ, O_2^H	α_1	-

Table 6.5: Possibilities for characterising two parameters. The figures corresponding to the 2 cases computed are indicated.

Parameters to be estimated	Known parameters	Figure
χ, α_1, O_2^H	None	6.21

Table 6.6: Possibilities for characterising the three parameters. The figure corresponding to the case computed is indicated.



(a) O_2^H.

(b) O_2^H knowing α_1.

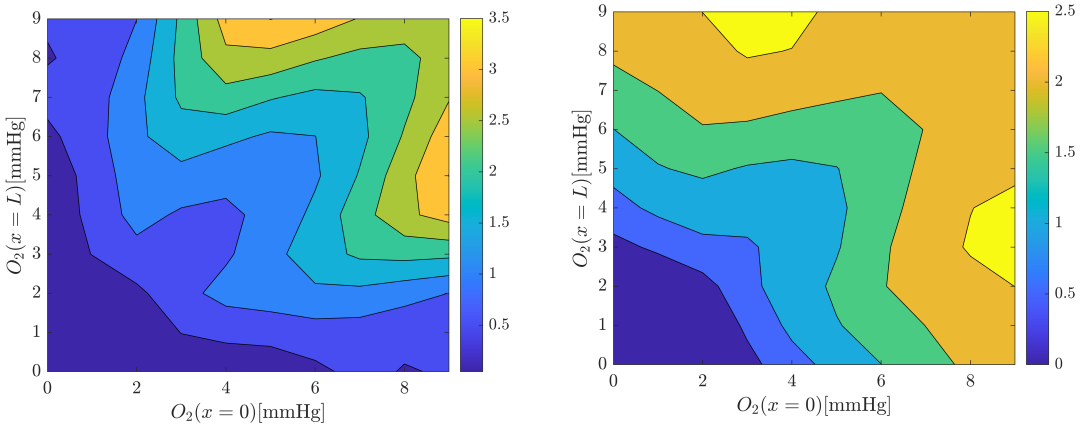
(c) O_2^H knowing χ.

(d) O_2^H knowing α_1 and χ.

Figure 6.19: Utility for fitting O_2^H. The degree of knowledge about the system greatly determines which experiment gives maximal information. Note that the utility has a relative meaning, so the value of the utility cannot be compared between different plots.

The cells boundary conditions remained the same in all experiments except for the value of J, which depends on the cell leakage observed for each microfluidic device. In the necrotic core experiment, $J = 1.0 \times 10^6\,\mathrm{s/cm}$; in the pseudopalisade experiment, $J = 1.0 \times 10^9\,\mathrm{s/cm}$, and in the double pseudopalisade experiment, $J = 1.2 \times 10^7\,\mathrm{s/cm}$. These three values have been adjusted for obtaining good agreement between the experimental curves and the simulations, without modifying the value of the model parameters.

Regarding the oxygen boundary conditions, they were adapted to each experimental configuration: in the formation of the double pseudopalisade they were identical to those already explained for the necrotic core formation; whereas in the pseudopalisade formation, impermeability condition (no flux) was imposed at the sealed channel, while, again, the Dirichlet condition was imposed at the right channel. The value of the oxygen pressure, both at the right side and at the initial time for the whole chamber, was fixed to $O_2^* = 2\,\mathrm{mmHg}$ instead of $O_2^* = 7\,\mathrm{mmHg}$ as in the other experiments. This is justified by the fact that in this experiment the medium was not renewed as in the previous cases, so the oxygenation was assumed to be lower.



(a) (α_1, χ). **(b)** (α_1, χ) knowing O_2.

Figure 6.20: Utility for fitting (α_1, χ)**.** The degree of knowledge about the system greatly determines which experiment gives maximal information. Note that the utility has a relative meaning, so the value of the utility cannot be compared between different plots.

The differences in boundary conditions for the three experiments are summarised in Table 6.7

Experiment	J [s/cm]	O_2^* [mmHg]
Necrotic core formation	1.0×10^6	7
Pseudopalisade formation	1.0×10^9	2
Double pseudopalisade formation	1.2×10^7	7

Table 6.7: Parameters related to boundary conditions.

The obtained results for each experiment together with the median and the 90% confident band (between 5^{th} and 95^{th} percentile) of the simulations are shown in Figs. 6.22, 6.23 and 6.24.

These figures show good agreement between the experimental and simulated results, which will be further explained in the Discussion section. Moreover, considering the parameter variability improves in some cases the estimation of the cell profiles (see Fig. 6.22 and 6.24). The main differences are:

- Our model tends to overestimate the cell concentration at the central region for the necrotic core experiment. This may be caused by an overestimation of the oxygen level in this region.
- There is a discrepancy between the cell concentration at the boundaries (both for alive and dead cells) for the predicted and experimental profiles.

We will discuss these differences in the Discussion section.

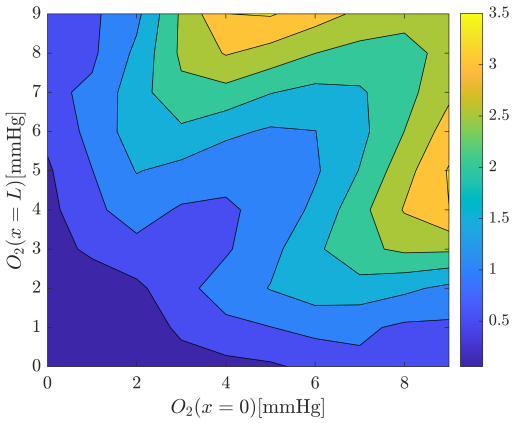
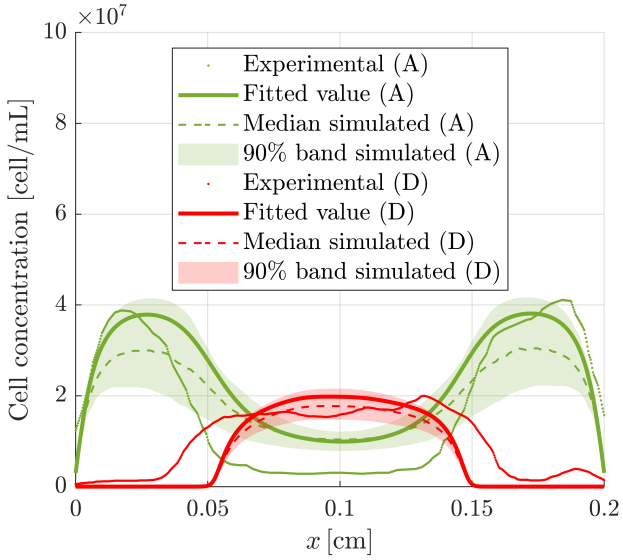
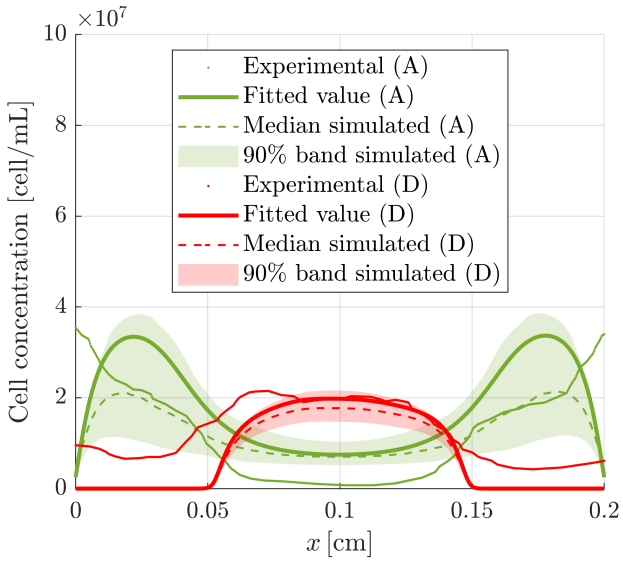


Figure 6.21: Utility for fitting (α_1, χ, O_2^H)**.** The degree of knowledge about the system greatly determines which experiment gives maximal information, although there are some general common trends. Note that the utility has a relative meaning, so the value of the utility cannot be compared between different plots.

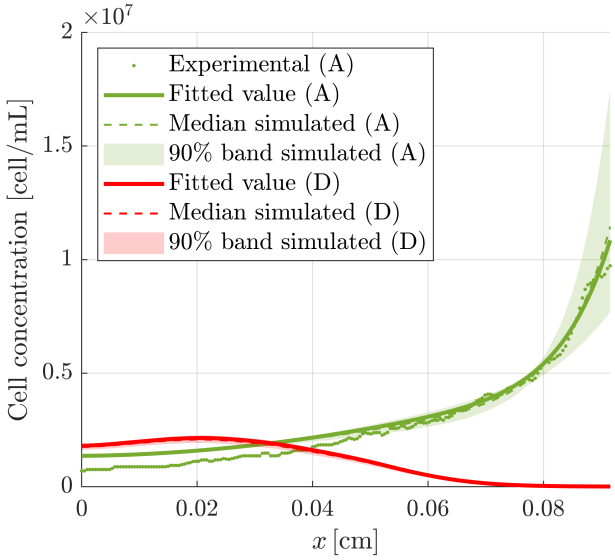


(a) Day 3.

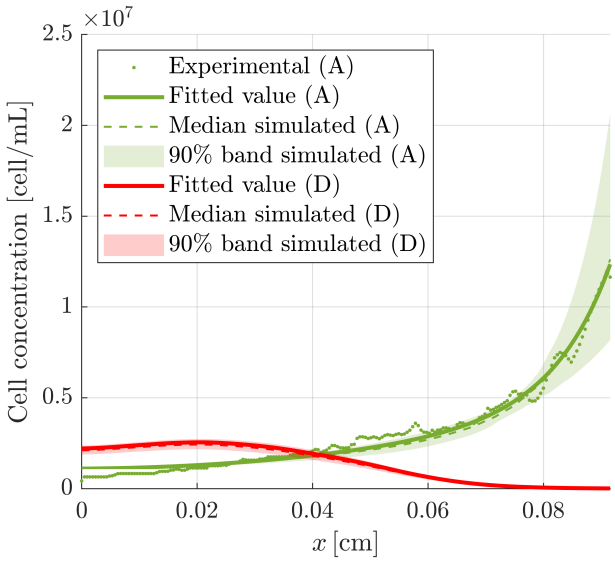


(b) Day 6.

Figure 6.22: Necrotic core formation. Confidence band for the simulated profiles and experimental results (y-axis) along the length of the chip (x-axis) for the necrotic core formation experiment. A: Alive cells. D: Dead cells.

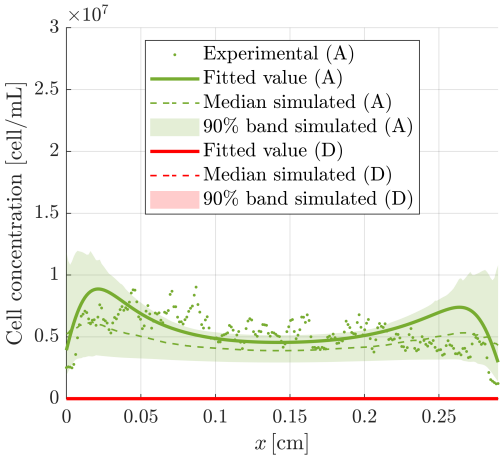


(a) Day 3.

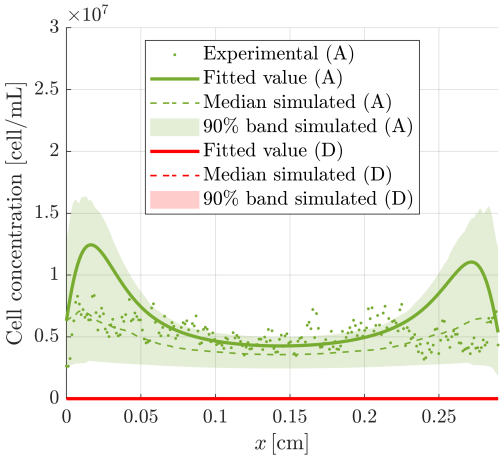


(b) Day 6.

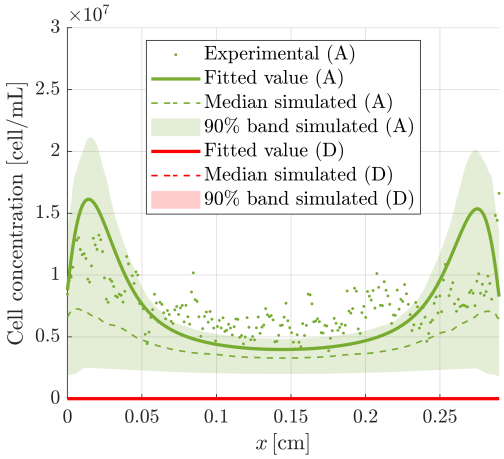
Figure 6.23: Pseudopalisade formation. Confidence band for the simulated profiles and experimental results (y-axis) along the length of the chip (x-axis) for the pseudopalisade formation experiment. A: Alive cells. D: Dead cells.



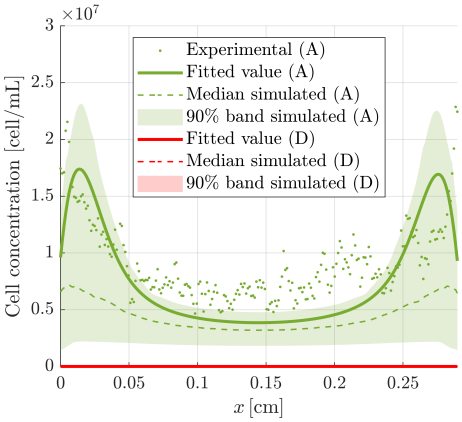
(a) Day 7.



(b) Day 11.



(c) Day 17.



(d) Day 21.

Figure 6.24: Double pseudopalisade formation. Confidence band for the simulated profiles and experimental results (y-axis) along the length of the chip (x-axis) for the low cell concentration experiment. A: Alive cells. D: Dead cells.

6.5 Discussion

6.5.1 Biological discussion

6.5.1.1 Biological evidences enforced by the computational model

GBM is one of the deadliest tumour types, as it is very heterogeneous and resistant to therapy (Aum et al., 2014; Ramirez et al., 2013; Shergalis et al., 2018). Most research studies have been performed in 2D models, on Petri dishes, but these are not able to represent the real situation. Many 3D culture models are now being developed, such as spheroids, organoids, different scaffold-based models, etc., which better mimic cell-cell and cell-ECM interactions (Hoarau-Véchot et al., 2018; Manini et al., 2018). With the development of microfluidics for biological applications, apart from including different components of the tumour micro-environment (TME), we are also able to control the physico-chemical conditions, so microfluidic models are considered the most biomimetical *in vitro* models nowadays (Logun et al., 2018; Paguirigan and Beebe, 2008). Special devices have been developed for GBM research to study the behaviour of GBM cells and therapy response within a biomimetic and controlled microenvironment. With them, realistic behavioural patterns have been obtained, similar to the ones observed in patients (Ayuso et al., 2016; Cui et al., 2020; Xiao et al., 2019; Yi et al., 2019).

In our case, we were able to reproduce autoinduced necrotic core and pseudopalisade formation (Ayuso et al., 2016, 2017), some of the most important characteristics of GBM, in agreement with the GBM formation model of Brat (2012), which identifies blood vessel occlusion and subsequent hypoxia-induced migration towards the functional blood vessel, following oxygen and nutrients gradient, as one of the main drivers of GBM invasion. From our *in vitro* models of pseudopalisades and necrotic core formation in GBM, we set the following phenomena in the model mathematical equations:

- GBM cells migrate in an O_2 pressure gradient following a chemotactic cue from lower to higher pressure values, and with a migration speed that depends on the specific local O_2 pressure value.
- Very high cell concentrations prevent the arrival of sufficient oxygen to regions far from oxygen provision sources, due to the oxygen uptake in the transition zones, thus resulting in an auto-induced necrotic core in those far regions. Lee et al. (2018) explained the importance of shortage of oxygen and nutrients in necrotic core formation. Also, a similar process was observed in spheroid cultures (Däster et al., 2017; Riffle and Hegde, 2017), where the gradient depends on the spheroid diameter, observing the appearance of necrotic core in spheroids bigger than $500\,\mu$m.
- Regarding the parameters of the model, we have shown that:
  - Due to the difficulties in defining an *in silico* model able to deal with all the complex phenomena at the TME level, the range of the model parameters in the literature is very wide. Moreover, the *go-or-grow* behaviour requires a special attention, as there is still much to decipher about the metabolic

pathways involved in this transition. Thereby, the parameters related to cell migration, both random or oxygen mediated, and with cell proliferation, that is, D_n, χ, O_2^H, $\alpha_1 = \tau_{\text{gr}}$ present an important variability.

- We have obtained a parameter set within the range of the literature except for the saturation concentration c_{sat} that, as explained, is very dependent on the experimental operation.
- Keeping constant the random pedesis parameter, D_n, and all the parameters related to the oxygen evolution (D_{O_2}, α and k_{m}), we have seen that the dynamics is determined by the triplet $(\frac{1}{\tau_{\text{gr}}}, \alpha, O_2^H)$ which is for us, and for the moment, our way of encoding the *go-or-grow* metabolic switch.
- For a sensitivity analysis involving other parameters, such as the related to oxygen evolution, the reader may consult the published version of this chapter in Ayensa-Jiménez et al. (2020b).

- Cell adaptation may also have an important role in problems like the one described here but has not been considered so far in our work. This will be part of future developments, although it will require new and specific sets of experiments to capture the corresponding mathematical features and parameters.

Since the model parameters fitting was established under a heuristic basis according to our research experience, the values selected should be interpreted qualitatively as the ones which better describe the most relevant phenomena that take place in GBM evolution *in vitro*, such as necrotic core and pseudopalisade formation. It is important to remark that a preliminary fitting approach based on a formal mathematical optimisation did not provide the best fit in the evolution of the necrotic core (data not shown), demonstrating the difficulty of the problem.

Anyway, despite the parameter uncertainty, it has been possible to reproduce three different experiments with one single set of parameters (see Figs. 6.22, 6.23 and 6.24, profiles corresponding to the fitted value). Therefore, it seems possible to extract fundamental biological conclusions, for instance:

- The initial cell density has a crucial influence on the necrotic core formation. The simulation results (and the experimental curves) related to the necrotic core and the double pseudopalisade experiments are essentially identical except for the fact that the former was obtained with an initial concentration of $C_0 = 40 \times 10^7$ cells/mL and the latter with $C_0 = 4 \times 10^6$ cells/mL. The corresponding results are however qualitatively very different: the necrotic core only appears when having very high cell concentrations. This conclusion may have important biological and therapeutic consequences.
- Cell migration strongly depends on the oxygen level and gradient. Suitable oxygenation conditions do not induce cell migration even for high oxygen gradients. Conversely, under a certain oxygenation level, the oxygen gradient is the main driving agent of cell migration. It has been shown (Kaur et al., 2005; Monteiro et al., 2017) that hypoxic conditions lead to stabilisation of hypoxia-inducible factor (HIF) which regulates many important pathways important for tumour progression, such as invasion and angiogenesis.

The presented mathematical model is, therefore, able to capture some of the main features of some essential phenomena occurring during GBM invasion. Moreover, except for the saturation parameter c_{sat} (which is obviously very dependent on the experimental conditions), the parameter variability is in agreement with that found in the scientific literature. Also, most of the general features observed are similar to the ones obtained in previous experimental (Ayuso et al., 2016, 2017) and computational (Ayuso et al., 2017; Martínez-González et al., 2012, 2015; Swanson et al., 2011; Vital-Lopez et al., 2011) works.

6.5.1.2 Model limitations to explain the experimental data

The model also presents some limitations, some of the most important are the following:

- Small discrepancies between the experimental and computational results were found. One possible explanation is that the accumulation of cells at the boundaries may obstruct the oxygen diffusion, provoking a non-homogeneous O_2 diffusion coefficient. This results in an over-estimation of the oxygen level in the central area of the chamber, which could explain the over-estimation of alive cells, faster cell migration to the boundaries and over-estimation of dead cells.
- Also, there are differences in cell concentration at the boundaries. One possible explanation is that the cells at the boundaries are in a 2D ambient (instead of 3D) so cell proliferation is accelerated, which leads to inaccuracies, as our model was thought for 3D environments and the parameter values are fitted for obtaining accurate results far from the boundaries.
- There is, in general, but mainly in the results associated with the necrotic core experiment, a certain lag between the computational and the experimental responses of cells to oxygen variations. It seems that changes in the oxygen concentration are felt earlier in the simulations. This may be associated with a certain cell memory: the cell may need to accumulate some "stress" before undergoing significant changes in its behaviour. Our GBM model is not able to capture this kind of phenomena. Nonetheless, we have presented a framework where this limitation can be overcome if more phenotypes are considered as in other works (Ayuso et al., 2017; Martínez-González et al., 2012, 2015). This strategy requires, however, a sound classification of the cell phenotypes, based on biological evidence, in a sufficient number of classes, which is difficult and would considerably increase the number of parameters.

6.5.2 Intrinsic model strengths and limitations

6.5.2.1 Benefits of the statistical approach

The train-test methodology based on *copulae* followed in the fitting process has shown that it is possible to establish a gradation in the strength of the parameter dependencies and some uncertainty bounds for the parameters. Once the probabilistic statistical model is adjusted, predicting the actual value of the model parameters is easily carried out. As it is observed in Figs. 6.16, 6.17 and 6.18, the normality assumption for the confidence region estimation is not always a good starting hypothesis at least for two reasons:

- First, it does not take into account the complexity of the relationship between the model parameters (i.e. physical phenomena) and may lead to unrealiable values (meaningless physical magnitudes, such as negative values for the hypoxic threshold or chemotaxis coefficient, as in Fig. 6.16a).
- Secondly, it may mislead with respect to the uncertainty that we actually have for different significance levels.

In any case, the confident region estimation using HDR and a proper probabilistic analysis are very informative about the degree of reliability of the mathematical model used for a biological explanation. These two observations become even more evident when the uncertainty of the model depends on the knowledge about the values of some of the model parameters, as it can be seen when comparing total with conditional distributions. In addition, HDR may inform about the difficulties in estimating the value of a certain parameter, for instance, the hypoxic threshold in our case. Indeed, as the cell culture is in hypoxic conditions it is difficult to specify the exact value of this threshold, because we are able to obtain good agreement between the experimental data and simulations for a broad range of values of this parameter.

Additionally, the knowledge of the model parameters variability (from a probabilistic point of view) allows to predict the outcome of a given experiment and its uncertainty. This can be used not only for model calibration and validation, but also for experimental planning (deciding the appropriate material, equipment or accuracy of the measuring devices and techniques to be used). For example, in Fig. 6.2, it may be seen that the necrotic core experiment requires less accuracy in the measurement of the cell profile in the central part of the chamber for parameter estimation, while the pseudopalisade experiment requires a measurement technique able to detect low alive cell concentrations. It can also be observed that the appearance of significant alive and dead cells at the right side of the chamber in the necrotic core experiment would not be explained by the model parameter variability, but rather by a model limitation. Moreover, the probabilistic knowledge of the model can be further exploited in experimental planning and design by using BED theory. In the analysis performed in this work, there is an important aspect to remark: the knowledge about the system may determine the best experiment (in the sense of information theory) to characterise the model parameters. We have seen that, for determining the parameters related to the *go-or-grow* metabolic switch, moderate gradient experimental settings are generally preferred, but the best operation conditions depend on the knowledge about the different biological aspects. Indeed, Figs. 6.19, 6.20 and 6.21 show how the structural dependence between the model parameters affects the optimal experiment for characterising the *go-or-grow* from the parametric point of view. A summary of the analysis is presented in Table 6.8, where the best experimental configuration is presented for each of the parameters' calibration, together with the maximum utility value.

6.5.2.2 Model limitations

The presented statistical methodology, even if it exceeds the performance of standard deterministic parameter analysis, shows the difficulties of working with such kind of

Parameters to be estimated	**Upper O_2 concentration**	**Lower O_2 concentration**	**Maximum utility value**
O_2	9 mmHg	5 mmHg	1.9
O_2 knowing α_1	5 mmHg	2 mmHg	1.6
O_2^H knowing χ	9 mmHg	6 mmHg	2.7
O_2^H knowing α_1, χ	9 mmHg	4 mmHg	2.2
α_1, χ	9 mmHg	5 mmHg	3.5
α_1, χ knowing O_2^H	9 mmHg	3 mmHg	2.7
α_1, χ, O_2^H	9 mmHg	5 mmHg	3.5

Table 6.8: Most useful experimental configuration for each of the parameters' evaluation. The values are orientative.

parametric models in biology. For instance, Fig. 6.13 illustrates the strength of the correlation relationships, showing that there are some phenomena difficult to isolate from the experimental and/or computational points of view. Thus, it is impossible to characterise their individual effect on cell behaviour if we have limited measurements available on the cell profiles or a poor model. It is then only possible to evaluate their combined resultant effect.

This analysis may be done for each model parameter couple, justifying the approach adopted in this work even if here we have focused on the *go-or-grow* metabolic switch. This dependence is illustrated in the way that each parameter estimation may depend on the knowledge about the value of the rest of the parameters.

Indeed, for the GBM progression model, the two sources of variability and structural correlations are:

- **Cell inherent variability**: Cell motility is induced by the random motion inherent to any cell and several taxis effects driven by external physical or chemical stimuli. Mathematical parameters related to these phenomena (e.g. diffusion and chemotaxis coefficients) appearing in the model equations will present, therefore, a correlation in the different experimental samples. Besides, energetic resources consumed by the cell are finite, so an increase in the migratory activity would probably decrease the proliferative activity and *vice-versa*, showing another clear correlation between the model parameters. Another subtlety resides in the fact that, due to epigenetic changes, the cell may have previously adapted its metabolism to oxygen changes and therefore may react in a different manner to the same external stimuli (Joseph et al., 2015; Musah-Eroje and Watson, 2019), introducing an extra degree of variability.
- **Difficulty to isolate parameters**: With the presented GBM model, and due to the nature of the measurements, it is very difficult to isolate different effects and phenomena. As explained before, it is difficult to ensure if the lack of oxygen is due to a poor diffusion or high consumption, or if the increase in the cell profile close to the boundaries is due to a fast proliferation or a fast migration towards the boundary. Even if we consider that this is due to migration, we would have to ask ourselves if this is because we have a high hypoxic threshold O_2^H or a high chemotaxis coefficient χ.

Hence, to overcome the inherent difficulties of the parametric models, we need different approaches that are capable of decoupling the uncertainty due to the inherent variability to the one related to the parametric fitting. Indeed, only the approaches having this feature would effectively give us information about the problem biological insigth, as they will not be corseted to bias-inducing parametric structures.

6.6 Conclusions

Mathematical modelling of complex cell processes is very challenging due to its intrinsic non-linearity, highly-coupled multiphysic interactions, and the many correlated parameters which are difficult to measure or simply unknown. These parameters are most times obtained for a particular problem under specific conditions, leading in many cases to conclusions directly derived from the modelling assumptions and therefore providing little new information. Also, they are difficult to generalise.

As a result, a proper and extensive parametric analysis is mandatory. This should include an extensive and detailed study of the values reported in the bibliography, a careful sensitivity analysis and a sufficient number of different experiments, not only for calibration but also for validation, avoiding parameter overfitting.

This analysis, although it allows the identification of the optimal set of parameters, is most times difficult to extend to other problems with reasonable accuracy and therefore with a certain validation of its actual physical character and its value range. It is also difficult to discriminate between correlated parameters associated with mechanisms that cannot be isolated in the experiments. Hence, we need additional information both to get a better discrimination between them, and to identify the optimal conditions for additional experiments to provide the maximum information possible in order to get such discrimination.

From the results and discussion presented above, we enumerate the main findings and conclusions of our work:

1. Mathematical modelling and the corresponding computer simulation of the complex cell processes involved, incorporating several interactions, chemical and physical cues, require an extensive literature review and analysis of the fundamental properties of the mathematical equations in simplified conditions, and an in-depth analysis of the model parameters, in order to understand the individual and combined effect of each combination of parameters, both qualitative and quantitatively, in the resulting variables. In that sense, we have proved here that *copulae* are a simple and powerful tool to detect and improve highly-correlated multiparametric mathematical models such as those appearing in Biology, with the added value of providing key information for the optimal design of new experiments with the highest information possible for the problem in hands, thus reducing time and cost not only in our *in vitro* experiments but also in scarce and costly *in vivo* cases.
2. One single type of experiment is not enough to guarantee a proper quantification and understanding of the effect of each model parameter. Some families of

experiments have to be used to fit the parameters, while other families are required to validate and discard spurious parametric solutions. This strategy is fundamental to avoid overfitting and to prevent model-induced effects, result of the fitting procedures.

3. There is a huge variation in the range of many of the parameters found in the literature, sometimes with intervals covering several orders of magnitude, which makes it very difficult to get a reasonable accuracy when modelling experimental tests using only values from bibliography. This can be a result of the high heterogeneity of GBM, the intrinsic variability to which biological problems are subjected when treated using imperfect models, and of the high adaptive capacity of these cells (Dirkse et al., 2019; Friedmann-Morvinski, 2014). The need of suitable methods taking into account parameter variability and their structural dependencies (that is, correlation) is, therefore, compulsory.
4. With a proper parameter identification and analysis, if all the main mechanisms involved are properly considered, it is possible to get an accurate reproduction of experimental tests, provided the experiment is well controlled, the associated variables are accurately measured and the results are correctly interpreted. However, this capacity for reproducing experimental outcomes may not be directly related to a real understanding of the different processes involved or to the identification of which is the dominant one, due to the intrinsic limitations of parametric models.
5. Adopting the presented model as a starting point, there is still room for future development. For instance, the measurement of the oxygen profile would allow us to improve the oxygen diffusion model, taking into account, for example, the oxygen flow obstruction that may be induced by high cell densities. Another important improvement, related to the methodological approach, is to be able to escape from the rigidity of parametric models, that suffer from an *a priori* modelling bias that obscures the acquisition of true knowledge about biological processes.

To summarise, the presented framework is general and allows the analysis of many coupled and highly non-linear physical mechanisms. The effort made in the parametric analysis allows drawing conclusions both qualitative (e.g. pseudopalisade and necrotic core formation) and quantitative (e.g. time scale for necrosis or speed of migration structures). This task is fundamental when working with complex multiparametric models. Nonetheless, this analysis is always conditioned by the choice of the mathematical approach, so the intrinsic model uncertainty is, to some extent, unavoidable. Working with models with so many parameters requires always enough experimental data in sufficiently varied conditions. Here, we have been able to work with three different families of experiments resulting in cell profiles along space and time. This extensive amount of information gives value to our work, which could lay the foundations for future works in the topic, including the incorporation of artificial intelligence (AI) and, in particular, machine learning (ML) tools.

Finally, we can conclude that, even with all this, we are still far from getting sufficient knowledge of all the mechanisms involved in complex biological processes, as well as the interactions and quantification of the corresponding phenomenological parameters.

Only in very specific and well-controlled conditions, and after an extensive analysis of the tests, model and associated parameters, it is possible to expect accurate results if the initial conditions are well-measured and the main mechanisms and interactions are mathematically represented. Despite these drawbacks, mathematical models are today invaluable tools to better understand underlying mechanisms and interactions, to establish trends, to test new hypotheses and to check "what if" situations that are many times impossible to test experimentally due to the impossibility to isolate single effects, measure particular variables or simply because of ethical reasons.

Scientific outcomes
As a result of this work, we published the following scientific articles:

- Ayensa-Jiménez J., Pérez-Aliacar M., Randelovic T., Oliván S., Fernández L., Sanz-Herrera, J. A., Ochoa I., Doweidar M. H., Doblaré M. "Mathematical formulation and parametric analysis of in vitro cell models in microfluidic devices: application to different stages of glioblastoma evolution", *Scientific Reports*, 2020.
- Ayensa-Jiménez J., Pérez-Aliacar M., Randelovic T., Sanz-Herrera J. A., Doweidar M. H., Doblaré M. "Analysis of the parametric correlation in mathematical modeling of in vitro glioblastoma evolution using copulas", *Mathematics*, 2020.

Also, we participated in a book publication writing one book chapter:

- Ayensa-Jiménez J., Doweidar M. H., Randelovic T., Fernández L., Oliván S., Ochoa I., Doblaré M. "On the Simulation of Organ-on-Chip Cell Processes: Application to an In Vitro Model of Glioblastoma Evolution", *Advances in Biomechanics and Tissue Regeneration*, Elsevier, 2019.

7. Data-Driven approach to Glioblastoma evolution.

Contents

7.1 Introduction

In Chapter 6 we have developed a mathematical model incorporating the *go-or-grow* hypothesis, which allowed us to reproduce the glioblastoma (GBM) evolution under different experimental configurations also *in vitro* (Ayensa-Jiménez et al., 2020b), and to derive, from cell culture images, information on the cell behaviour (Pérez-Aliacar et al., 2021).

However, parametric models are corseted by the mathematical relations that describe the *a priori* assumed biological hypotheses, so they present an obvious modelling bias. Besides, in our experiments, we try to understand the intrinsic mechanisms that

control these biological processes; a knowledge that goes further than the numerical value of a specific parameter and is generally related to concepts with biological meaning, such as whether the metabolic change is sharp or smooth, localized or distributed, or presents one or many different levels of transition.

In Chapter 5, we have presented a new promising family of neural networks, named physically-guided neural networks with internal variables (PGNNIV) as a tool to identify, evaluate and derive constitutive models from observable measurements (Ayensa-Jiménez et al., 2020a, 2021a). The fundamental idea is to incorporate the physical knowledge on the system into the network and to concentrate the learning power of artificial neural networks (ANNs) in the intrinsic physical mechanisms that are intended to be found out. Very recently, a similar approach combining neural networks and physical equations (physics-informed neural network (PINN) (Raissi et al., 2019)) has been proposed in Lagergren et al. (2020) as a way to discover hidden mechanistic relationships, using the Fisher–Kolmogorov–Petrovsky–Piskunov equation as a benchmark problem, a concept that has been coined as biologically-informed neural networks (BINN). All the same, PGNNIV offer greater flexibility in the definition of the internal variables of interest, including the non-measurable ones. Thus, they are able to unravel more complex metabolic mechanisms. Additionally, they may deal with problems involving variable external stimuli, that is, different source terms and boundary conditions, something that PINNs cannot afford.

In this chapter, we demonstrate how PGNNIV allow unravelling the mathematical structure that identifies the intrinsic metabolic mechanisms associated with cell changes due to the variation of some measurable fields, as the oxygen concentration around the cell that can be measured in microfluidic devices. This identification of the detailed *go-or-grow* mechanism related to hypoxia allows to accurately predict the cell evolution under highly variable external stimuli, including normoxic, gradient and hypoxic configurations, without the requirement of any initially prescribed parametric relation. This methodology does not only permit the identification of complex metabolic changes but also improving the prediction accuracy of parametric models.

This chapter is structured as follows. The Materials and Methods section describes the formulation here presented. First, the mathematical model for GBM evolution is briefly revised, emphasising how the role of hypoxia is commonly taken into account for modelling the *go-or-grow* paradigm. Then, the model under the PGNNIV framework is presented, detailing the data generation and the network training process. In the Results section, the main results are presented: the unravelling of the metabolic behaviour and the ability to predict the cell evolution under different oxygenation conditions. Finally, in the Discussion section, the present and future of this methodology are discussed, while in the Conclusions, the main results and conclusions are summarised.

7.2 Materials and methods

Our typical experimental configuration for the tests here analysed is the same presented in Chapter 6, as was illustrated in Fig. 6.5. Here, the geometry of the model (assumed as one-dimensional, as the length of the lateral channels is

much larger than the width of the chamber), and the different field variables are represented. Provided that the length scale is large enough, it is possible to identify the cell concentration with a continuum field $u_1 = u_1(x,t)$. Besides, the oxygen concentration is associated with a field $u_2 = u_2(x,t)$. The oxygen is supplied to the cell culture via the lateral channels. In response to this stimulus, cells will undergo migration and/or proliferation along the width of the chamber of length l.

7.2.1 Mathematical model of Glioblastoma cell culture evolution

7.2.1.1 Governing equations

Our starting point is a nonlinear reaction-diffusion system of partial differential equations (PDEs) that governs the evolution of GBM cells and the concentration of oxygen in a microfluidic device, very similar to the one presented in Chapter 6 and published in Ayensa-Jiménez et al. (2020b). In particular, the equations of the fields' evolution are:

$$\frac{\partial u_1}{\partial t} = \frac{\partial}{\partial x}\left(D_1 \frac{\partial u_1}{\partial x} - \chi \Pi_{\mathrm{go}} u_1 \frac{\partial u_2}{\partial x}\right) + \alpha_1 \Pi_{\mathrm{gr}} u_1 \left(1 - \frac{u_1}{c_{\mathrm{sat}}}\right), \tag{7.1a}$$

$$\frac{\partial u_2}{\partial t} = \frac{\partial}{\partial x}\left(D_2 \frac{\partial u_2}{\partial x}\right) - \alpha_2 \left(\frac{u_2}{u_2 + k_{\mathrm{m}}}\right) u_1. \tag{7.1b}$$

The first term of the R. H. S. of Eq. (7.1a) represents the flow term associated with cell culture migration and has two contributions: the non-oriented motility term $D_1 \frac{\partial u_1}{\partial x}$ (modelled here as a random diffusion process) and the chemotaxis term $-\chi \Pi_{\mathrm{go}} u_1 \frac{\partial u_2}{\partial x}$, where cell motility is induced by the oxygen gradient. Π_{go} is a correction factor that will be discussed later. The second term corresponds to the reaction term and is associated with logistic growth (Tsoularis and Wallace, 2002), except for the correction term Π_{gr} that will be also explained later.

With respect to the oxygen evolution equation, Eq. (7.1b), the first term of the R. H. S. is again the flow term, consisting solely of oxygen diffusion, and the second corresponds to the oxygen consumption by cells. The correction between brackets in the second term is the Michaelis-Menten kinetic model (Cornish-Bowden, 2013) and accounts for the kinetics of oxidative phosphorylation that occurs in the membrane of cellular mitochondria (Tang, 1933).

With respect to the model detailed in Chapter 6, we have introduced two minor modifications[23]:

1. We have neglected the necrotic phenotype and also the death term in the normoxic population (that now is the only cell population considered). This does not affect the methodology presented and is done merely for reducing the number of parameters (death characteristic time, anoxia-induced dead location parameter and anoxia-induced death spread parameter) and the number of fields involved, for the sake of simplicity.

[23] The designation of some parameters is different from the used in Chapter 6, to adapt them to the vectorial framework, $D_1 = D_n$, $D_2 = D_{O_2}$, $\alpha_2 = \alpha$, $\alpha_1 = 1/\tau_{\mathrm{gr}}$.

2. We have considered that the chemotaxis correction due to space limitation has no effect and therefore may be considered as equal to the unity, to focus on the phenomenon we are interested in, that is the *go-or-grow* behaviour. As in the previous case, this does not affect the approach as it may be adapted to any system of PDEs.

Eqs. (7.1a) and (7.1b) must be complemented with appropriate boundary and initial conditions. The initial condition is a known cell profile that is seeded in the microfluidic device at the beginning of the experiment (normally constant in the whole chamber). Note that we will refer to this time as $t = 0$ even if it is not necessarily the experiment starting time, identifying the instant when the cell culture profile is measured and the microfluidic device is fully oxygenated:

$$u_1(x, t = 0) = c(x), \tag{7.2a}$$
$$u_2(x, t = 0) = \mathrm{O}_2^*(x), \tag{7.2b}$$

with $c(x)$ a given known function and $\mathrm{O}_2^*(x)$ the ambient oxygen level. The cell culture is subjected to a fixed oxygen concentration at the lateral channels. Besides, we assume that the walls of the culture chamber at the microfluidic devices are impermeable to cells, so only oxygen flow is allowed through them. In that case, the boundary conditions are:

$$\left.\frac{\partial f_1}{\partial x}\right|_{x=0} = 0, \tag{7.3a}$$
$$\left.\frac{\partial f_1}{\partial x}\right|_{x=l} = 0, \tag{7.3b}$$
$$u_2(x = 0, t) = \mathrm{O}_2^{\mathrm{L}}(t), \tag{7.3c}$$
$$u_2(x = l, t) = \mathrm{O}_2^{\mathrm{R}}(t), \tag{7.3d}$$

where we have defined $f_1 = D_1 \frac{\partial u_1}{\partial x} - \chi \Pi_{\mathrm{go}} u_1 \frac{\partial u_2}{\partial x}$ as the cell flow, l is the length of the culture chamber and $\mathrm{O}_2^{\mathrm{L}}(t)$ and $\mathrm{O}_2^{\mathrm{R}}(t)$ are known functions defining the oxygen levels at the two lateral channels on the sides of the the chamber.

At this point, the presented framework has seven model parameters, D_1, D_2, χ, α_1, α_2, c_{sat} and k_{m}. Some of them have a well-identified value in the scientific literature. For example:

- The oxygen diffusion, $D_2 = 1 \times 10^{-5}\,\mathrm{cm}^2 \cdot \mathrm{s}^{-1}$ has been reported in many works (Daşu et al., 2003; Tannock, 1972).
- The Michaelis-Menten constant, $k_{\mathrm{m}} = 2.5\,\mathrm{mmHg}$, is very particular of the specific kinetics of the reaction in hands (Alper and Howard-Flanders, 1956; Daşu et al., 2003).

All other parameters can be easily determined in specific well-controlled experiments. For example:

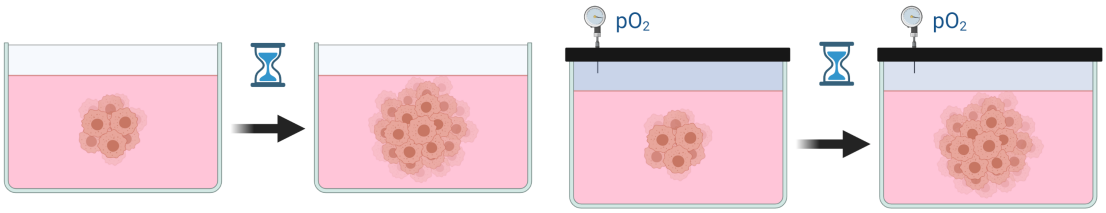
- The parameters related to the logistic cell growth, α_1 and c_{sat}, can be determined in cell growth experiments under fully oxygenated conditions and in absence of oxygen gradient, both in microfluidic devices (Lei et al., 2014; Tao et al., 2015) or using cell spheroids (Vinci et al., 2012).
- The oxygen consumption rate, α_2 is easily obtained by measuring the oxygen pressure at the ambient in an isolated system with a controlled cell culture population and for high oxygenation levels, such that the Michaelis-Menten correction, between brackets in Eq. (7.1b), may be considered as 1. It is even possible to determine both k_{m} and α_2 from the oxygen pressure using an Eadie–Hofstee diagram (Hofstee, 1959) or a Lineweaver–Burk plot (Lineweaver and Burk, 1934).
- The non-oxygen-mediated pedesis constant, D_1, is more complicated to determine as spatial cell cultures are necessary. However, spheroid cultures (Ayuso et al., 2015) and microfluidic devices (Ayuso et al., 2016) offer a great opportunity for cell migration evaluation. If full oxygenation is guaranteed in the whole culture, no oxygen gradient is formed so non-oxygen mediated pedesis is easily computed, for instance, once given α_1, D_1 can be determined by evaluating the cell migration radial velocity V and using the Fisher's model (Fisher, 1937b), $V = 2\sqrt{D_1\alpha_1}$.
- The value of χ is substantially more difficult to determine. Indeed, as the cell migration depends on the oxygen level (and not only on the oxygen gradient), it is difficult to estimate this value from one single experiment or measurement. However, we can measure the cell culture migration under an oxygen gradient in a very localized region where the oxygen level may be considered almost constant (Funamoto et al., 2012). Nevertheless, as it will be discussed later, we are rather interested in the overall value $\chi\Pi_{\text{go}}$. In this relation, χ is a reference value and Π_{go} is a correction term incorporating the effect of hypoxia in the migration.

Fig. 7.1 illustrates some schematic experiments that can be implemented to determine the model parameters appearing in Eq. (7.1) using conventional cell culture techniques and microfluidic devices.

In addition to the model parameters, the evolution of the GBM cell culture is also influenced by the boundary and initial conditions, in terms of the functions $c(x)$, O_2^*, $x \in [0; l]$, $\mathrm{O}_2^{\mathrm{L}}(t)$ and $\mathrm{O}_2^{\mathrm{R}}(t)$, $t \in \mathbb{R}^+$, which play the role of problem data. That is, for the problem to be perfectly defined, we need to specify the functions c, $\mathrm{O}_2^{\mathrm{L}}$ and $\mathrm{O}_2^{\mathrm{R}}$ together with the ambient oxygen pressure O_2^*.

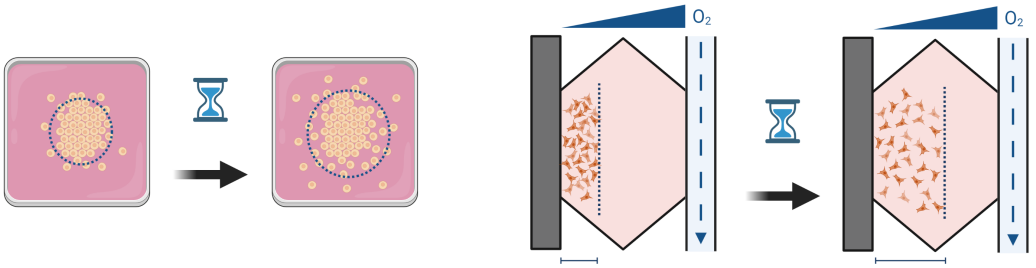
7.2.1.2 *Go-or-grow* activation functions

The metabolic behaviour of the GBM cells, in particular, its response to changes in the oxygen pressure, is mathematically encoded in the functional form of Π_{go} and Π_{gr}. These two functions regulate the activation/deactivation of both processes: migration and proliferation. There is sound evidence in the scientific literature that the switch between the proliferative and migratory activity in a cell population is hypoxia-mediated (Carreau et al., 2011; Lu and Kang, 2010), that is $\Pi_{\text{go}} = \Pi_{\text{go}}(u_2)$ and $\Pi_{\text{gr}} = \Pi_{\text{gr}}(u_2)$. However, there is not much knowledge about the details of this metabolic change. For instance, some questions about the process are:



(a) Determining α_1 and c_{sat}.

(b) Determining α_2 and k_{m}.



(c) Determining D_1.

(d) Determining χ.

Figure 7.1: Scheme of the different experiments that can be performed to obtain the model parameters. Obtaining the source term parameters does not require a spatial cell distribution, although this is necessary for characterising the cell parameters associated with migration. Created with BioRender.com.

- Are migration and proliferation simultaneous or not?
- Is the transition between them smooth?
- Is the transition strictly monotonic?
- Is the transition restricted to a narrow interval in the oxygen concentration region?

In Chapter 6, this transition was modelled by using piecewise linear functions of the rectified linear unit (ReLU) kind, that is:

$$\Pi_{\text{go}}(u_2) = \begin{cases} 1 & \text{if} \quad u_2 \leq 0 \\ 1 - \frac{u_2}{\theta_{\text{go}}} & \text{if} \quad 0 < u_2 \leq \theta_{\text{go}}, \\ 0 & \text{if} \quad \theta_{\text{go}} < u_2 \end{cases} \tag{7.4}$$

and

$$\Pi_{\text{gr}}(u_2) = \begin{cases} 0 & \text{if} \quad u_2 \leq 0 \\ \frac{u_2}{\theta_{\text{gr}}} & \text{if} \quad 0 < u_2 \leq \theta_{\text{gr}}, \\ 1 & \text{if} \quad \theta_{\text{gr}} < u_2 \end{cases} \tag{7.5}$$

Here, θ_{go} and θ_{gr} play the role of oxygen thresholds. Additionally, it has been assumed that $\theta_{\text{go}} = \theta_{\text{gr}}$, so this model implicitly assumes that $\Pi_{\text{go}}(u_2) + \Pi_{\text{gr}}(u_2) = 1$, even if this consideration should, in principle, be modified to rely on a deeper understanding of the cell metabolism and in particular of the cell energy consumption.

The parameter values associated with Eqs. (7.4) and (7.5) provided reasonably accurate results in the characterisation of certain cell cultures. In particular, GBM culture evolution of the cell line U251-MG in microfluidic devices has been well described, even for different experimental configurations using these expressions (Ayensa-Jiménez et al., 2020b). Similar results were obtained now using machine learning (ML) tools (in particular, using convolutional neural network (CNN)), also revealing some limitations of the parametric model (Pérez-Aliacar et al., 2021).

However, the *go-or-grow* model may differ from one GBM cell line to another. Besides, the model should be adapted for other tumour families or different frameworks. Therefore, since the functional relation $u_2 \mapsto (\Pi_{\text{go}}, \Pi_{\text{gr}})$ encodes the cell metabolic changes in response to changes in the oxygen stimulus, its accurate characterisation is crucial for a complete understanding of the evolution of cell cultures, as it describes the changes that take place in the cell population behaviour and consequently, in the tumour progression (Ayuso et al., 2017; Monteiro et al., 2017). If $\mathbf{\Pi} = (\Pi_{\text{go}}, \Pi_{\text{gr}})$, the *go-or-grow* relation, may be written as:

$$\mathbf{\Pi} = \mathbf{\Pi}(u_2), \tag{7.6}$$

where $\mathbf{\Pi} : \mathbb{R}^+ \to \mathbb{R}$ is the unknown functional relation to be learned. Unravelling the one input-two output relation $\mathbf{\Pi}$ is therefore a key aspect in an *in silico* model able to capture tumour progression in an oxygenated medium. However, one main problem arises: as Π_{go} and Π_{gr} are mathematical artefacts that only make sense when considered in Eq. (7.1), they are non-measurable variables, so there is no experimental set-up that permits to measure them directly. Furthermore, the measurement of the oxygen pressure in cell culture is usually difficult due to technical considerations, even if possible under some particular conditions (Lam et al., 2018; Zirath et al., 2018). This adds an extra difficulty when defining or calibrating the model $\mathbf{\Pi}$.

7.2.2 Physically-Guided Neural Network with Internal Variables

7.2.2.1 Concept of PGNNIV

The concept of PGNNIV was widely explained in Chapter 5, and may be seen as a generalisation of the former concept of PINN (Raissi et al., 2019). In this latter, the physics of the problem *informs* the network via the output variables: the physical equations constrain the values of the output variables to belong to a certain physical manifold. For instance, to ensure that they satisfy a given PDE. In other words, the loss function is directly defined in terms of the problem physics. PGNNIV go one step further, as in this case, the physical equations constrain the values reached by an arbitrary number of neurons in some intermediate layers. As a consequence, it is possible to interpret some hidden features and some relationships between internal variables of the problem that now acquire a physical interpretation (Ayensa-Jiménez et al., 2020a, 2021a). The physics does not *constrain*, but only *guides* the network learning capacity, as the measured data may be supplied to endow the network with explanatory capacity.

Going into the details, and following the notations of this thesis, a PGNNIV is a problem formulated in the following archetypal way. Let us consider a system of PDEs that is split into:

$$\begin{aligned} \mathcal{F}(u, v, f) &= 0, \quad \text{in } \Omega, \\ \mathcal{G}(u, v, g) &= g, \quad \text{in } \partial\Omega, \\ \mathcal{H}(u) &= v, \quad \text{in } \Omega, \end{aligned} \tag{7.7}$$

where u and v are the unknown scalar fields of the problem, $\mathcal{F}$ and $\mathcal{H}$ are functionals representing the known and unknown physical equations of the problem in hands. $\mathcal{G}$ is a functional that specifies the boundary conditions, and f and g are known fields. Once discretised, Eq. (7.7) has an analogous representation in finite-dimensional spaces in terms of vectorial functions $\boldsymbol{F}$, $\boldsymbol{G}$ and $\boldsymbol{H}$ and nodal values $\boldsymbol{u}$, $\boldsymbol{v}$, $\boldsymbol{f}$ and $\boldsymbol{g}$. The physically-guided problem is therefore formulated as:

$$\begin{aligned} \boldsymbol{y} = \mathsf{Y}(\boldsymbol{x}); \quad \boldsymbol{v} &= \mathsf{H}(\boldsymbol{u}), \\ \text{s. t.} \quad \boldsymbol{x} &= \boldsymbol{I}(\boldsymbol{u}, \boldsymbol{f}, \boldsymbol{g}), \\ \boldsymbol{y} &= \boldsymbol{O}(\boldsymbol{u}, \boldsymbol{f}, \boldsymbol{g}), \\ \boldsymbol{R}(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{f}, \boldsymbol{g}) &= \boldsymbol{0}, \end{aligned} \tag{7.8}$$

where:

- $\boldsymbol{R}$ are the physical constraints, related to the relationships given by $\boldsymbol{F}$ and $\boldsymbol{G}$.
- $\boldsymbol{I}$ and $\boldsymbol{O}$ are functions specifying the input and the output of the problem, that is, the data used as starting point to make predictions and the data that we want to predict.
- Y and H are models. Y is the *predictive model*, whose aim is to infer accurate values for the output variables for a certain input set and H is the *explanatory model*, whose objective is to unravel the hidden physics of the relation $u \mapsto v$.

A PGNNIV is built when the problem (7.8) is formulated in the language of ANNs, with an appropriate structure and topology for Y and H.

7.2.2.2 Discretised model

Spatial discretisation.

Let us first discretise the Eqs. (7.1). This may be done by using finite differences (FD) or finite elements (FE) as it is usual when working with PDEs (Ganesan and Lingeshwaran, 2017). The one-dimensional character and simple geometry of the cell culture in microfluidic devices under oxygen gradients (Ayuso et al., 2016) allow us to use FD to discretise the governing equations. Then, we define the nodal values of the fields u_1 and u_2 using the vectors $\boldsymbol{u}_1$ and $\boldsymbol{u}_2$, that is, $u_{ij} = u_i(x_j)$ where $x_j = j\Delta x$, $j = 0, \dots, n$, is the spatial coordinate associated with a given discretisation of the domain $[0; l]$, $\Delta x = \frac{l}{n}$. The spatial derivatives may be computed using any FD scheme, resulting in a linear operator $\boldsymbol{D}_x$ Then, Eq. (7.1) results in:

$$\dot{\boldsymbol{u}}_1 = \boldsymbol{D}_x \left(D_1 \boldsymbol{D}_x \boldsymbol{u}_1 - \chi \boldsymbol{\Pi}_{\text{go}} \odot \boldsymbol{u}_1 \odot \boldsymbol{D}_x \boldsymbol{u}_2 \right) + \alpha_1 \boldsymbol{\Pi}_{\text{gr}} \odot \boldsymbol{u}_1 \odot \left(1 - \frac{\boldsymbol{u}_1}{c_{\text{sat}}} \right), \tag{7.9a}$$

$$\dot{\boldsymbol{u}}_2 = \boldsymbol{D}_x \left(D_2 \boldsymbol{D}_x \boldsymbol{u}_2 \right) - \alpha_2 \left(\boldsymbol{u_2} \oslash (\boldsymbol{u_2} + k_{\text{m}}) \right) \odot \boldsymbol{u}_1. \tag{7.9b}$$

We have used the symbols $\odot$ and $\oslash$ for indicating pointwise multiplication and division respectively. It is important to note that $\mathbf{\Pi}_{\text{go}}$ and $\mathbf{\Pi}_{\text{gr}}$ are here vector functions. The framework presented permits considering functional relationships, that is, the value of $\mathbf{\Pi}$ at a point x could depend on the value of the field u_2 at the whole computational domain. However, the underlying nature of the *go-or-grow* framework allows us to consider $\Pi(x) = \Pi(u_2(x)), x \in [0; l]$, or equivalently, for the vector $\mathbf{\Pi}$, $\Pi_j = F(u_{2j})$, permitting to work with sparse graphs for the network topology, according to the considerations discussed in Chapter 5. Consequently, we obtain sparse tensors and operators. In order to adapt the problem to our notations, let us describe Eq. (7.9) as:

$$\begin{aligned} \dot{\boldsymbol{u}} &= \boldsymbol{F}(\boldsymbol{u}, \mathbf{\Pi}), \\ \mathbf{\Pi} &= \boldsymbol{H}(\boldsymbol{u}). \end{aligned} \tag{7.10}$$

Temporal discretisation.

With respect to the time integration, many options are possible. Multistep and Runge-Kutta methods (Lambert, 1991) are some of the most efficient, although they are also computationally expensive. For our purposes, it is enough to consider a two-point scheme. Given the ordinary differential equation (ODE) $\dot{\boldsymbol{y}} = \boldsymbol{f}(\boldsymbol{y})$, we discretise it using the generalised mid-point rule, that is, approximating $\boldsymbol{y}(t + \Delta t) - \boldsymbol{y}(t) \simeq \Delta t \boldsymbol{f}\left(\beta \boldsymbol{y}(t) + (1 - \beta)\boldsymbol{y}(t + \Delta t)\right)$, $\beta \in [0; 1]$. This approximation leads to the discretisation:

$$\boldsymbol{y}^{n+1} = \boldsymbol{y}^n + (\Delta t)\boldsymbol{f}\left(\beta \boldsymbol{y}^n + (1 - \beta)\boldsymbol{y}^{n+1}\right). \tag{7.11}$$

With this notation, taking $\beta = 1$ we recover the forward Euler method and with $\beta = 0$ we recover the backward Euler approach.

Applying Eq. (7.11) to Eq. (7.10) we obtain:

$$\begin{aligned} \boldsymbol{u}^{n+1} &= \boldsymbol{u}^n + (\Delta t)\boldsymbol{F}\left(\beta \boldsymbol{u}^n + (1 - \beta)\boldsymbol{u}^{n+1}, \mathbf{\Pi}\right), \\ \mathbf{\Pi} &= \boldsymbol{H}(\beta \boldsymbol{u}^n + (1 - \beta)\boldsymbol{u}^{n+1}). \end{aligned} \tag{7.12}$$

Finally, we may define the residual $\boldsymbol{R}$ as:

$$\boldsymbol{R}(\boldsymbol{u}^n, \boldsymbol{u}^{n+1}) = \boldsymbol{u}^{n+1} - \boldsymbol{u}^n - (\Delta t)\boldsymbol{F}\left(\beta \boldsymbol{u}^n + (1 - \beta)\boldsymbol{u}^{n+1}, \mathbf{\Pi}(\boldsymbol{u}^{n+1}, \boldsymbol{u}^n)\right). \tag{7.13}$$

The presented framework is generalisable to multistep and Runge-Kutta integrators. For instance, for the latter:

$$\boldsymbol{u}^{n+1} = \boldsymbol{u}^n + (\Delta t)\sum_{i=1}^{s} b_i \boldsymbol{k}_i, \tag{7.14}$$

with

$$\begin{aligned} \boldsymbol{k}_i &= \boldsymbol{F}\left(\boldsymbol{u}^n + (\Delta t)\sum_{j=1}^{s} a_{ij}\boldsymbol{k}_j, \mathbf{\Pi}\right) \quad i = 1, ..., s, \\ \mathbf{\Pi} &= \boldsymbol{H}\left(\boldsymbol{u}^n + (\Delta t)\sum_{j=1}^{s} a_{ij}\boldsymbol{k}_j\right), \end{aligned} \tag{7.15}$$

where a_{ij}, b_i and c_i, $i = 1, \ldots, s$ are the particular coefficients of the selected numerical scheme.

In that case, the residual $\boldsymbol{R}$ may be written as:

$$\boldsymbol{R}(\boldsymbol{u}^n, \boldsymbol{u}^{n+1}) = \boldsymbol{u}^{n+1} - \boldsymbol{u}^n - (\Delta t) \sum_{i=1}^{s} b_i \boldsymbol{k}_i, \tag{7.16}$$

where $\boldsymbol{k}_i$ is given by Eq. (7.15).

7.2.2.3 The Physically-Guided Neural Network

The crucial part of building the PGNNIV is the definition of the network topology, as well as of the input and output layers. As explained when defining the biological problem, there is no way of straightforwardly measuring the variables Π_{gr} and Π_{go}, so these will be our internal variables, $\boldsymbol{v} = \boldsymbol{\Pi}$, while $\boldsymbol{u} = (\boldsymbol{u}_1, \boldsymbol{u}_2)$. The measured variables will be the cell and oxygen profiles at two consecutive time steps, that is:

$$\boldsymbol{I}(\boldsymbol{u}, \boldsymbol{v}) = \boldsymbol{u}^n, \tag{7.17a}$$

$$\boldsymbol{O}(\boldsymbol{u}, \boldsymbol{v}) = \boldsymbol{u}^{n+1}. \tag{7.17b}$$

The reason for defining the input and output variables this way is to achieve accurate predictive capacity, besides the required explanatory capacity. Indeed, once the model has been trained, it is possible to predict the outcome, that is, the cell and oxygen profiles at time $t + \Delta t$, from the ones given at time t. Note that the cell profiles (the output) at time $t + \Delta t$ are obtained in real-time, as there is no need for solving any differential equation (we only need a network evaluation). The predictive and explanatory subnetworks are, therefore:

$$\boldsymbol{u}^{n+1} = \mathsf{Y}(\boldsymbol{u}^n), \tag{7.18a}$$

$$\boldsymbol{\Pi} = \mathsf{H}(\boldsymbol{u}_2). \tag{7.18b}$$

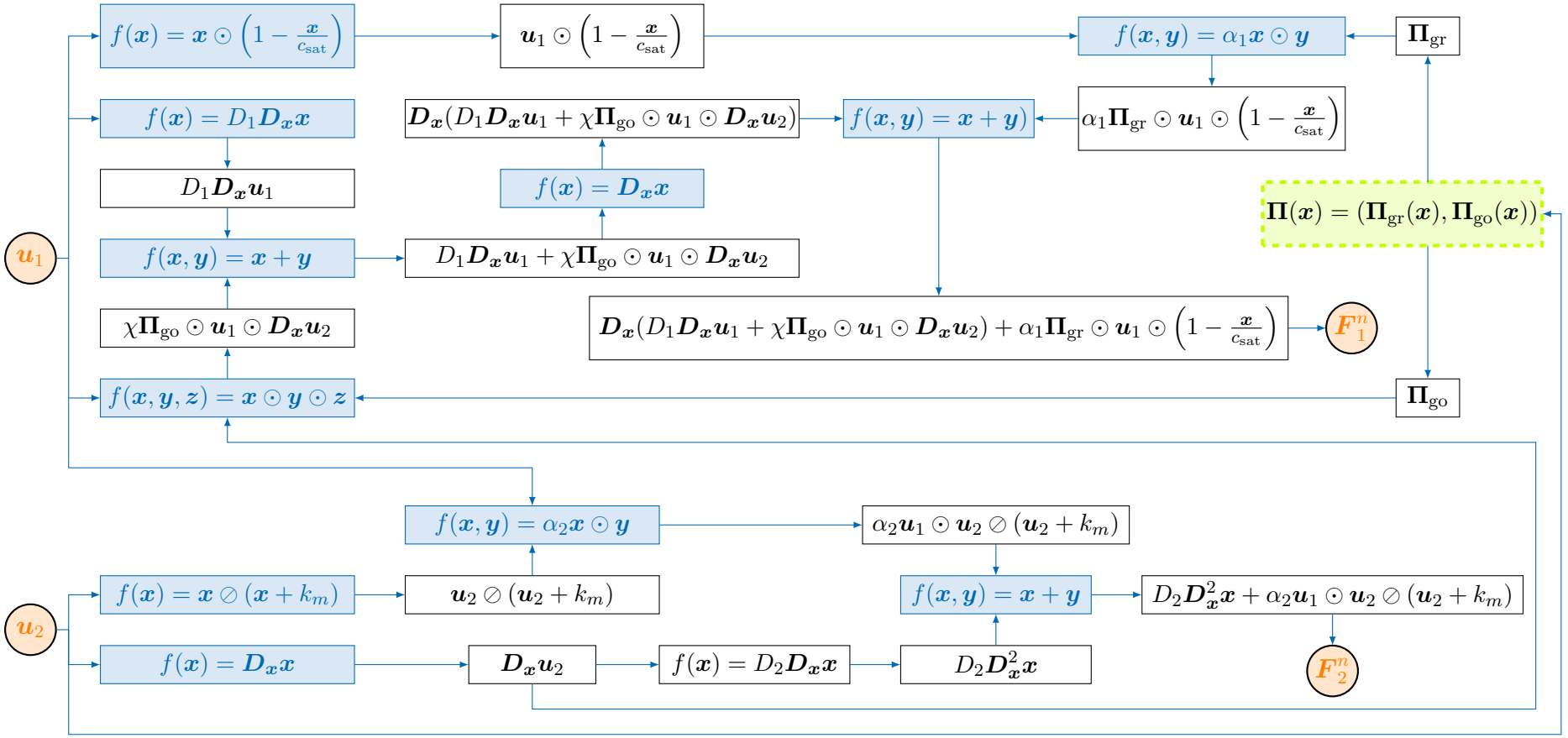
The PGNNIV graph and flow are illustrated in Fig. 7.2. It is important to note that, although the explanatory subnetwork is the juxtaposition of two multilayer perceptrons, it is applied at each nodal value, so it acts as a convolutional network moving through the different collocation points in space. Note that if $\beta = 1$, as it is the case for this work, the input solely corresponds to the field values at time step n.

In a PGNNIV, the network loss function is the combination of a physics-associated term and a data-associated term. However, given the topology of the presented network, illustrated in Fig. 7.2, all known physics of the problem is introduced explicitly in the network by means of the topology. Thus, the loss term is directly computed as:

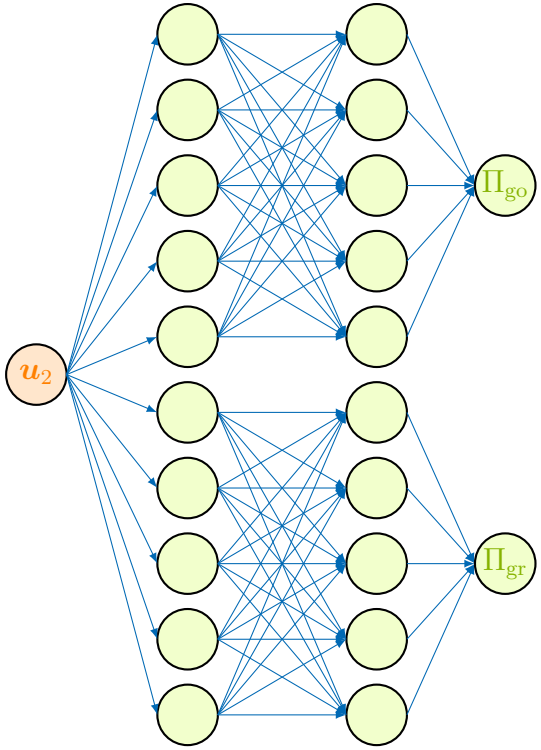
$$L = \sum_{n=1}^{N_{\text{data}}} \sum_{i=1}^{2} \|\hat{\boldsymbol{u}}_i^{n+1}(\boldsymbol{u}^n) - \boldsymbol{u}_i^{n+1}\|^2. \tag{7.19}$$

Recall that, observing the expression of the residual given by Eq. (7.13), Eq. (7.19) may be written as:

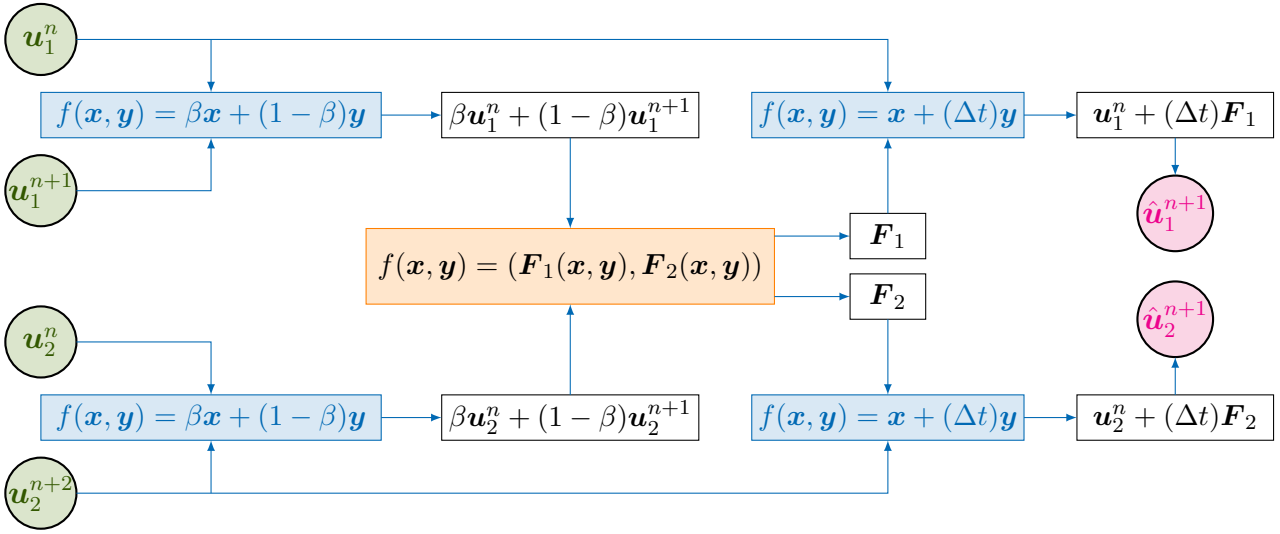
$$L = \sum_{n=1}^{N_{\text{data}}} \|\boldsymbol{R}(\boldsymbol{u}^n, \boldsymbol{u}^{n+1})\|^2. \tag{7.20}$$



(a) Model subnetwork.



(b) Explanatory subnetwork.



(c) Integration network.

Figure 7.2: Structure of the PGNNIV. Network topology and the different operators. The measurable (independent) variables, which are treated as the input variables, are represented in green while the predicted (dependent) variables, which are treated as the output of the network, are represented in magenta. The known operators are represented in blue, the unknown operators are represented in yellow and the hybrid operators are represented in orange.

7.2.3 Data generation and training process

7.2.3.1 Data for model validation

Here, we describe the data that will be used to feed the network. It is important to note that here the data-set is generated synthetically for validation purposes, but in real-life applications, this data-set would be the result of experimental measurements.

Benchmark models.

In order to evaluate the performance of the method let us suppose four different *true* functional relationships for the metabolic model $\mathbf{\Pi} = \mathbf{\Pi}(u_2)$ that are described in Table 7.1. For illustration purposes, we assume for the different models that $\Pi_{\mathrm{gr}}(x) = 1 - \Pi_{\mathrm{go}}(x)$, although the PGNNIV may unravel the metabolic behaviour for *true* models not satisfying this relationship.

Chemotaxis activation function	Growth activation function	Parameters	Model name
$\Pi_{\mathrm{go}}(x) = \mathbb{I}_{[0;\theta]}(x)$	$\Pi_{\mathrm{gr}}(x) = \mathbb{I}_{[\theta;+\infty)}(x)$	θ	Heaviside, Binary step
$\Pi_{\mathrm{go}}(x) = \left(1 - \frac{x}{\theta}\right) \mathbb{I}_{[0;\theta]}(x)$	$\Pi_{\mathrm{gr}}(x) = \frac{x}{\theta}\mathbb{I}_{[0;\theta]}(x)$	θ	Piecewise linear, ReLU
$\Pi_{\mathrm{go}}(x) = \frac{k}{x+k}$	$\Pi_{\mathrm{gr}}(x) = \frac{x}{x+k}$	k	Michaelis-Menten
$\Pi_{\mathrm{go}}(x) = \frac{1}{2}\left(1 + \tanh\left(\frac{x-\theta}{\Delta\theta}\right)\right)$	$\Pi_{\mathrm{gr}}(x) = \frac{1}{2}\left(1 - \tanh\left(\frac{x-\theta}{\Delta\theta}\right)\right)$	$\theta, \Delta\theta$	Sigmoid, Logistic

Table 7.1: Different functional relationships defined for the validation procedure. The different functions include features such as different smoothness and nonlinearities.

We claim that our PGNNIV based on the governing equation (7.1), encoding the known physics of the problem, is able to discover the actual biological metabolic model among the four presented in Table 7.1. This is possible due to the universal learning capabilities of neural networks (Cybenko, 1989; Hornik, 1991).

Profile generation.

The data were generated by simulating cell profiles using Eq. (7.1) with the boundary conditions (7.3). The model was first written using a dimensionless version, obtained by defining $t = T\tau, x = L\xi, u_1 = U_1\upsilon_1$ and $u_2 = U_2\upsilon_2$, where T is a characteristic time, L a characteristic length and U_1 and U_2 are characteristic cell and oxygen concentrations, obtaining:

$$\frac{\partial \upsilon_1}{\partial \tau} = \frac{\partial}{\partial \xi}\left(\bar{D}_1 \frac{\partial \upsilon_1}{\partial \xi} - \bar{\chi}\Pi_{\mathrm{go}}\upsilon_1 \frac{\partial \upsilon_2}{\partial \xi}\right) + \bar{\alpha}_1 \Pi_{\mathrm{gr}}\upsilon_1\left(1 - \frac{\upsilon_1}{\bar{c}_{\mathrm{s}}}\right), \tag{7.21a}$$

$$\frac{\partial \upsilon_2}{\partial \tau} = \frac{\partial}{\partial \xi}\left(\bar{D}_2 \frac{\partial \upsilon_2}{\partial \xi}\right) - \bar{\alpha}_2\left(\frac{\upsilon_2}{\upsilon_2 + \bar{k}_{\mathrm{m}}}\right)\upsilon_1, \tag{7.21b}$$

with boundary and initial conditions:

$$\upsilon_1(\xi, \tau = 0) = \bar{c}(x), \tag{7.22a}$$

$$\upsilon_2(\xi, \tau = 0) = \bar{\mathrm{O}}_2^*(x), \tag{7.22b}$$

$$\left.\frac{\partial \upsilon_1}{\partial \xi}\right|_{\xi=0} = 0, \tag{7.23a}$$

$$\left.\frac{\partial \upsilon_1}{\partial \xi}\right|_{\xi=\bar{l}} = 0, \tag{7.23b}$$

$$\upsilon_2(\xi = 0, \tau) = \bar{\mathrm{O}}_2^{\mathrm{L}}(\tau), \tag{7.23c}$$

$$\upsilon_2(\xi = \bar{l}, \tau) = \bar{\mathrm{O}}_2^{\mathrm{R}}(\tau), \tag{7.23d}$$

where the dimensionless parameters and functions are:

$$\begin{aligned}
\bar{D}_1 &= \frac{D_1 T}{L^2}, & \bar{\chi} &= \frac{\chi T U_2}{L^2}, & \bar{\alpha}_1 &= \alpha_1 T, \\
\bar{c}_{\mathrm{s}} &= \frac{c_{\mathrm{sat}}}{U_1}, & \bar{D}_2 &= \frac{D_2 T}{L^2}, & \bar{\alpha}_2 &= \frac{\alpha_2 T U_1}{U_2}, \\
\bar{k}_{\mathrm{m}} &= \frac{k_{\mathrm{m}}}{U_2}, & \bar{c}(\xi) &= \frac{c(x)}{U_1}, & \bar{\mathrm{O}}_2^{*}(\xi) &= \frac{\mathrm{O}_2^{*}(x)}{U_2}, \\
\bar{\mathrm{O}}_2^{\mathrm{L}} &= \frac{\mathrm{O}_2^{\mathrm{L}}}{U_2}, & \bar{\mathrm{O}}_2^{\mathrm{R}} &= \frac{\mathrm{O}_2^{\mathrm{R}}}{U_2}, & \bar{l} &= \frac{l}{L}.
\end{aligned} \tag{7.24}$$

In addition to the model parameters in Eq. (7.24), we have to consider the ones related to the different *go-or-grow* models described in Table 7.1:

$$\bar{\theta} = \frac{\theta}{U_2}, \quad \Delta\bar{\theta} = \frac{\Delta\theta}{U_2}, \quad \bar{k} = \frac{k}{U_2}. \tag{7.25}$$

All parameters stated in Eqs. (7.24) and (7.25) should have a precise biological meaning and depend on the problem physics. The values of the different parameters are reported in Table 7.2. Here, their value is only illustrative as they are used only for data generation, trying to make relevant all the biological phenomena.

Parameter	Value
$\bar{D}_1$	1
$\bar{\chi}$	1
$\bar{\alpha}_1$	0.5
$\bar{c}_{\mathrm{s}}$	10
$\bar{D}_2$	1
$\bar{\alpha}_2$	0.05
$\bar{k}_{\mathrm{m}}$	2
$\bar{l}$	50
$\bar{\theta}$	2
$\Delta\bar{\theta}$	2
$\bar{k}$	2

Table 7.2: Dimensionless model parameters used for data generation. The values are selected to make relevant all biological phenomena.

The different cell and oxygen profiles were generated by using the method described in Skeel and Berzins (1990), especially suitable for parabolic PDEs. Additional

details may be found in Chapter 6 or in Ayensa-Jiménez et al. (2020b). A mesh size of $\Delta\xi = 1.0$ and a time step of $\Delta\tau = 0.01$ were used. As initial conditions, we set a value of $\bar{c}(\xi) = 2$ and $\bar{\mathrm{O}}_2^*(\xi) = \bar{\mathrm{O}}_2^{\mathrm{L}} + \frac{\bar{\mathrm{O}}_2^{\mathrm{R}} - \bar{\mathrm{O}}_2^{\mathrm{L}}}{\bar{l}}\xi$. The duration of the experiment is $\tau^* = 10$. Therefore, once the temporal series of the fields (cell and oxygen profile) are generated, the output of each simulation is an array of size $[n_t, n_x, n_u]$ with $n_t = \tau^*/\Delta\tau + 1 = 1001$, $n_x = \bar{l}/\Delta\xi + 1 = 51$ and $n_u = 2$.

Feeding the network.

In order to recreate *in silico* different GBM On-Chip experiments, we created different cell profiles by varying the boundary conditions, that is, the oxygen levels $\bar{\mathrm{O}}_2^{\mathrm{L}}$ and $\bar{\mathrm{O}}_2^{\mathrm{R}}$. Two families of configurations were simulated: symmetrical and with oxygen gradient. The eleven *in silico* experiments performed are reported in Table 7.3. Each experiment is treated, from the PGNNIV point of view, as a batch of data as illustrated in Fig. 7.2, panel (c): the batch k, $k = 1, \dots, M$, with $M = 11$, is therefore obtained by considering the k-th experimental configuration and the network is fed using each pair $(\boldsymbol{u}_1^n, \boldsymbol{u}_2^n)$ as input data and each pair $(\boldsymbol{u}_1^{n+1}, \boldsymbol{u}_2^{n+1})$ as output, $n = 0, \dots, n_t - 1$. Each batch is therefore formed by an input of size $[n_t - 1, n_x, n_u]$ (from values $n = 0$ to $n = n_t - 2$) and an output of size $[n_t - 1, n_x, n_u]$ (from values $n = 1$ to $n = n_t - 1$). The objective is then to learn the underlying *go-or-grow* model for a particular experimental condition.

Configuration (k)	$\bar{\mathrm{O}}_2^{\mathrm{L}}$	$\bar{\mathrm{O}}_2^{\mathrm{R}}$
1	0	0
2	0	1
3	1	1
4	2	0
5	2	2
6	3	0
7	3	3
8	4	0
9	4	4
10	5	0
11	5	5

Table 7.3: Experimental configurations used for data generation. The different configurations recreate both symmetric and gradient configurations in low, medium and high oxygenated conditions (these values have to be compared with the model-associated ones, Eqs. (7.25)).

7.2.3.2 Training process

The neural network is trained using $N = 10^3$ epochs. At each epoch, all batches associated with the experimental configurations described in Table 7.3 are used for the network feeding. $p = 80\%$ of data at each batch is used for training purposes and $1 - p = 20\%$ for testing the network. In total, $N \times M$ iterations of the network are considered until reaching convergence. The Adam optimizer (Kingma and Ba, 2014) is selected with a learning rate $r = 0.001$ and an exponential decay rate of $\beta_1 = 0.8$ for the first moment and $\beta_2 = 0.8$ for the second moment are selected.

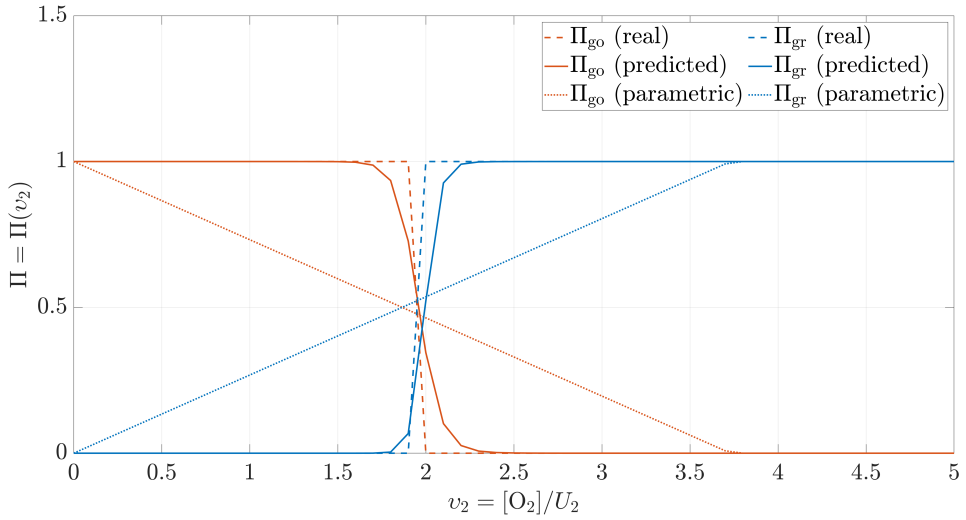
7.3 Results

As in all problems involving PGNNIV, the neural network has both predictive and explanatory capacity. To illustrate both concepts, we will discuss first the explanatory capacity of the network, which is particular to this method. For comparative purposes, we will compare the learned relationship, $\boldsymbol{H}$ in Eq. (7.8) or $\boldsymbol{\Pi}$ in Fig. 7.2 with standard parametric learning, where we postulate the model described by Eqs. (7.4) and (7.5), also assuming $\theta_{\text{go}} = \theta_{\text{gr}}$.

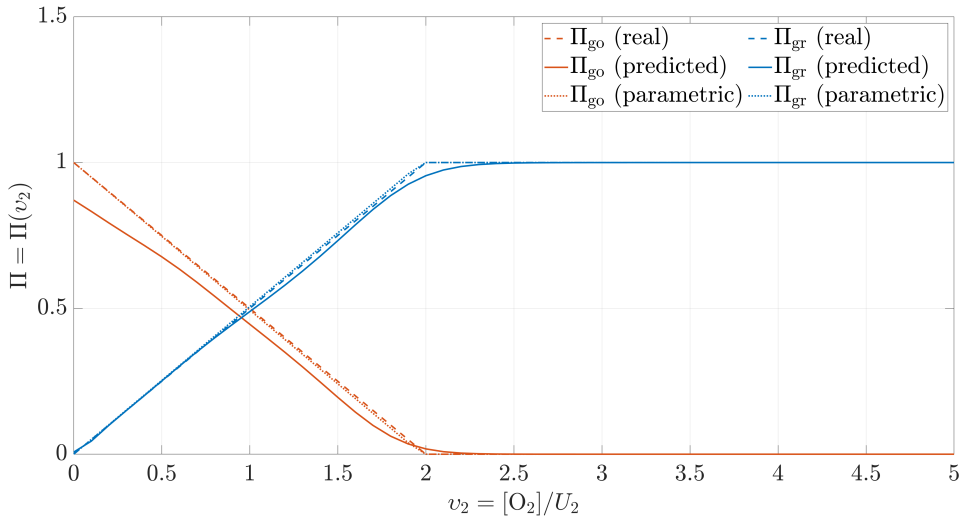
Then, we will comment on the predictive capacity of the network. As this capacity is not particular to the presented method but common to all regression techniques, we will compare our results to those obtained with standard parametric fitting.

7.3.1 Unravelling the metabolic changes of the glioblastoma cells

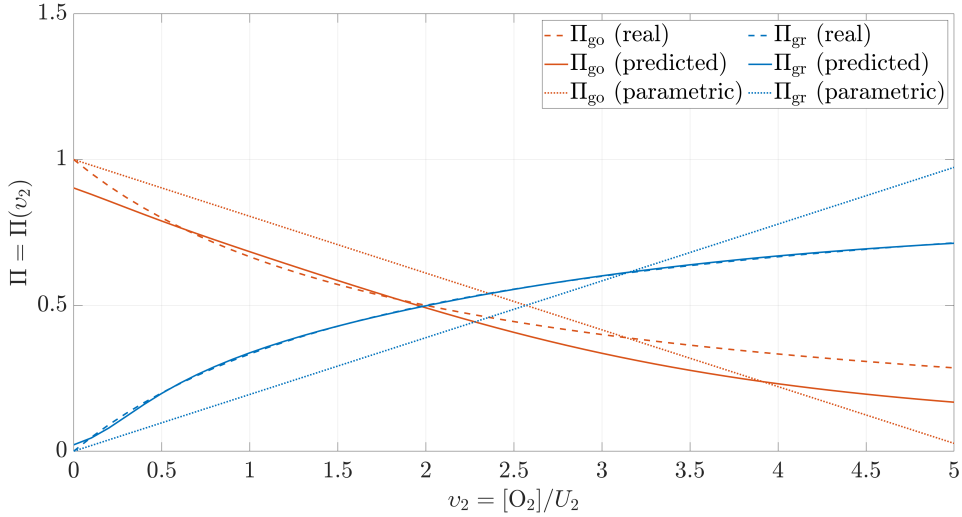
In Fig. 7.3, we depict the learned relationship $\boldsymbol{\Pi}$ for the four *ground truth* models proposed in Table 7.1. In all cases, a good agreement is shown between the real and the predicted models. Only when the parametric family for the *go-or-grow* model is adequately selected, the parametric learning (that is a particular PGNNIV where the function $\boldsymbol{\Pi}$ is parametrised) outperforms model-free PGNNIV, as it has been discussed in Chapter 5 and in Ayensa-Jiménez et al. (2020a, 2021a). Note that when no explicit knowledge is assumed about cell metabolism, it is difficult to either derive or postulate parametric relations such as those in Eqs. (7.4) and (7.5), which are solely used as a mere instrumental tool.



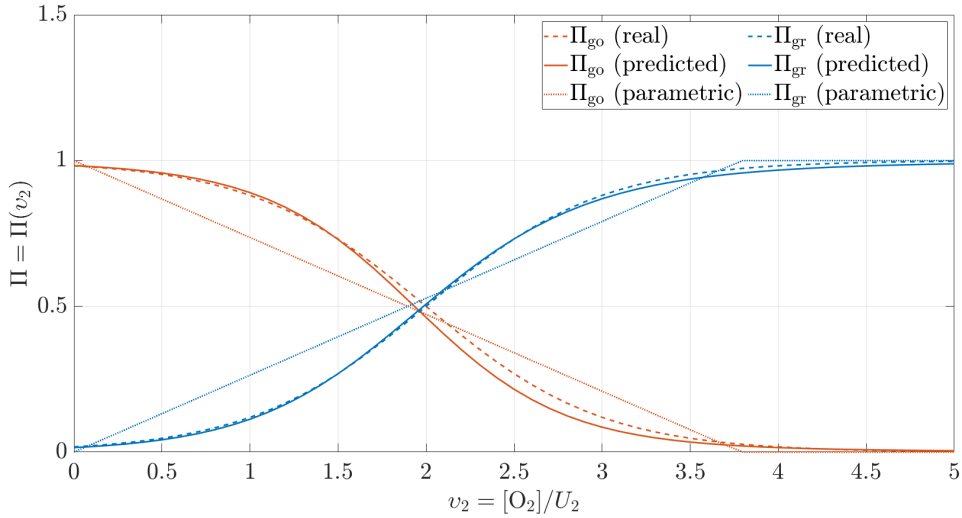
(a) Heaviside model.



(b) ReLU model.



(c) Michaelis-Menten model.



(d) Logistic model.

Figure 7.3: Unravelling capacity of the PGNNIV. For all models in Table 7.1, presenting different nonlinearities, smoothness and scales, the *ground truth* model is recovered correctly, especially in relation to the growth metabolic behaviour.

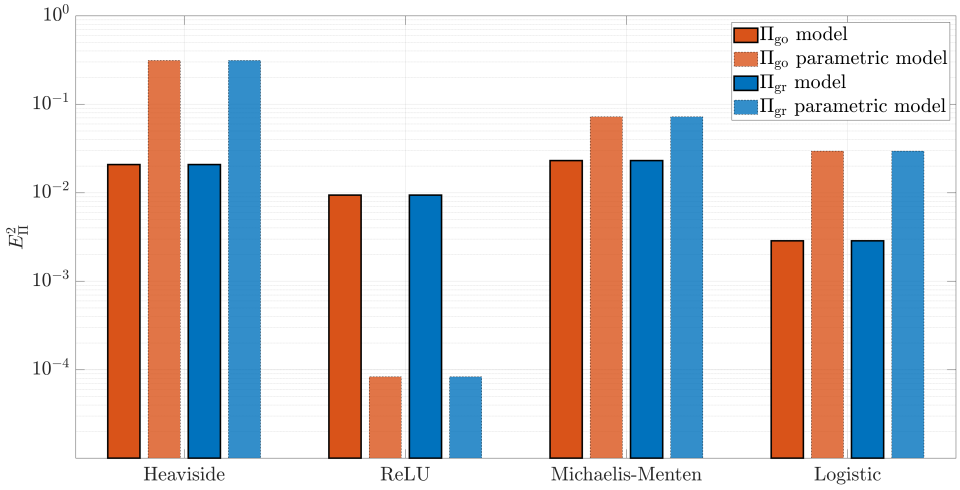


Figure 7.4: Error E^2_Π between the predicted and the real model. The error of the prediction is robust over the different transition functions tested and outperforms any parametric fitting.

Fig. 7.4 shows the errors when unravelling the metabolic behaviour $\mathbf{\Pi}$. Denoting by $\hat{\Pi}$ the model learned by the network, the error is defined as:

$$E^2_\Pi = \int_0^5 \left(\hat{\Pi}(x) - \Pi(x)\right)^2 dx. \tag{7.26}$$

These errors are computed for both Π_{go} and Π_{gr}. Except in the aforementioned case when the parametric family assumed includes the true model, the PGNNIV prediction clearly outperforms standard parametric approaches and keep the errors reduced for a broad family of families.

7.3.2 Predicting cell culture evolution

The aim now is to explore the predictive capacity of the neural network. Once the model $\mathbf{\Pi}$ has been learned, it is represented by the multilayer perceptron (MLP) topology together with all the network parameters (weights and biases). Therefore it may be encapsulated as a one input - two output *black-box* and inserted in any numerical integration scheme. For instance, we can consider any Runge-Kutta integrator of the form given by Eq. (7.14) for the spatially-discretised equations, that is, to follow the approach for data generation described in Section 7.2.3.1 except for the fact that we use the learned model $\mathbf{\Pi}$ instead of any other of those presented in Table 7.1.

For illustrative purposes, let us compare the cell and oxygen profiles for three different boundary conditions: a normoxic configuration where $\bar{O}_2^L = \bar{O}_2^R = 4$, a hypoxic configuration where $\bar{O}_2^L = \bar{O}_2^R = 0$ and a gradient configuration where $\bar{O}_2^L = 0$ and $\bar{O}_2^R = 4$. Our aim is to predict the cell profile after $\tau = 20$. The results are shown in Fig. (7.5), where we have represented for each cell profile the real one (the derived when using directly the function in Table 7.1), the one

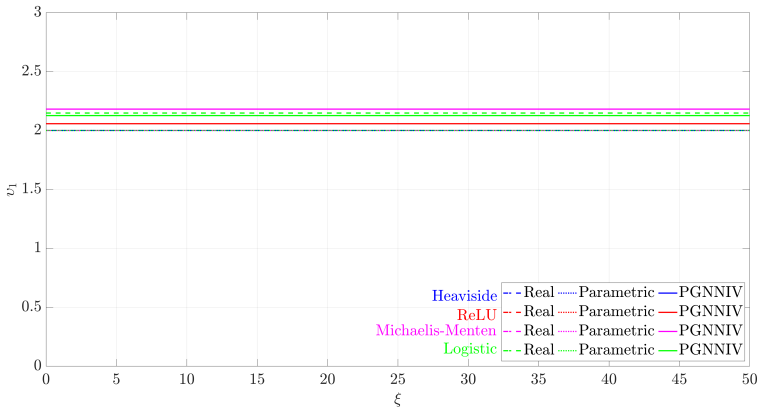
predicted after fitting the parametric model described by Eqs. (7.4) and (7.5) and the one predicted by PGNNIV. For all the models tested, a good agreement is shown between the predicted and the real profiles, and PGNNIV always outperforms the prediction of the parametric models, except, as it was explained before, for the ReLU case. The improvement of the prediction is particularly significant for the gradient configurations (Fig. 7.5, panel (b)). Indeed, the specific features of the model have a greater impact for oxygen levels in the transition between normoxic and hypoxic behaviour since it is in this case when the differences between the different models influence more the cell evolution.

In order to explore quantitatively the improvement, we define the error associated with the cell prediction as:

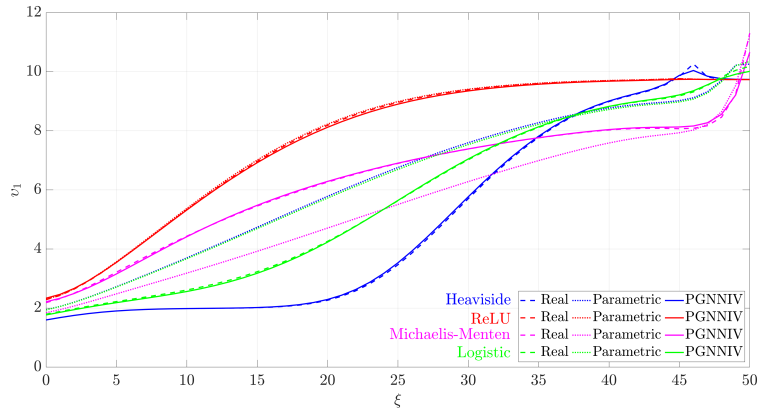
$$E^2_{\text{cell}} = \int_0^{50} \left(\hat{v}_1(\xi, \tau = \tau^*) - v_1(\xi, \tau = \tau^*)\right)^2 d\xi. \tag{7.27}$$

In Fig. 7.6, we compare the error of the cell prediction given by Eq. (7.27) when estimating the cell profile using the parametric approach and the one based on PGNNIV.

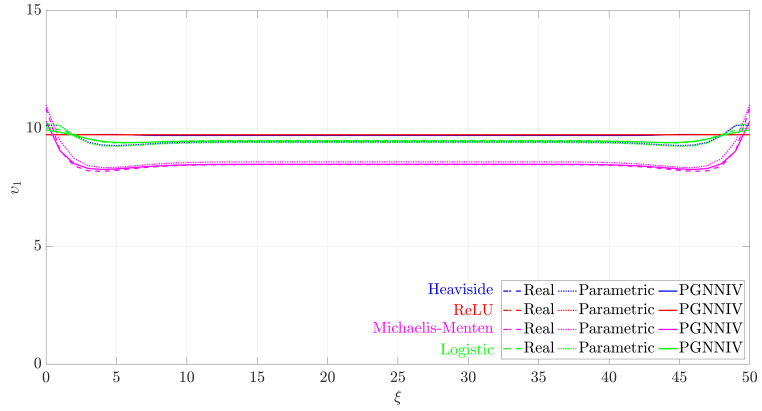
It is important to note that the training data-set was used for simulations for $\tau \leq \tau^*$ with $\tau^* = 10$, so we explore here the prediction capacity of the network out of the region defined by the training data-set.



(a) Hypoxic configuration.



(b) Gradient configuration.



(c) Normoxic configuration.

Figure 7.5: Prediction of the cell profile at $\tau^* = 10$ for the different models tested and different experimental configurations. PGNNIV improves the prediction (when compared to the parametric model) of all cell profiles, slightly for the normoxic configuration and significantly for the gradient configuration.

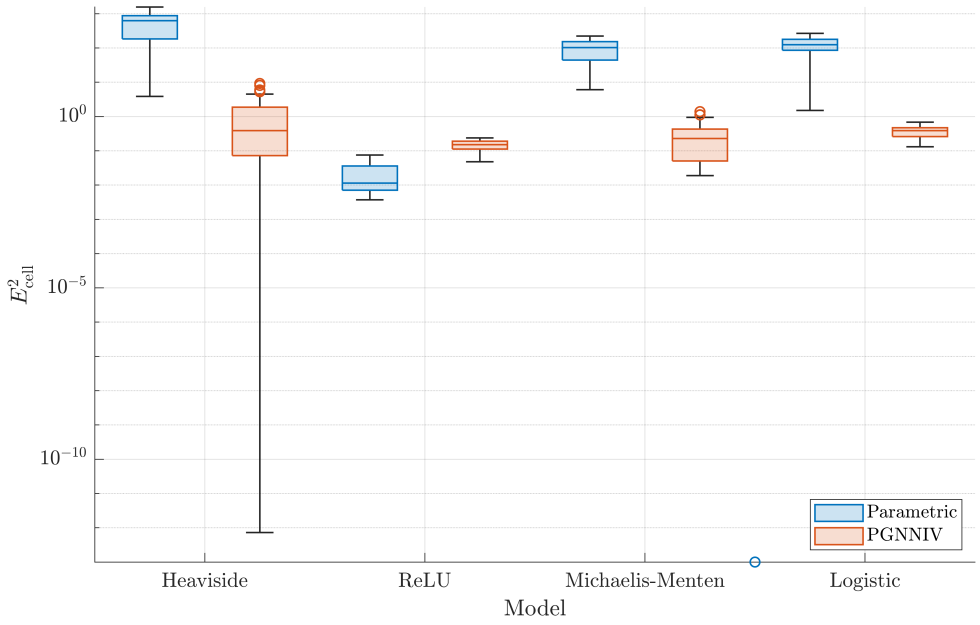


Figure 7.6: Prediction error: The non-parametric model built using the PGNNIV approach is able to correctly estimate the cell profile evolution for any arbitrary model better than specific parametric ones. The predictions are more accurate (lower error) and more precise (lower variability) along with the different tested boundary conditions.

7.4 Discussion and open possibilities

7.4.1 The present: characterisation of complex biological cell processes

Discovery of hidden cell metabolisms is a major concern in biology. Indeed, unravelling the changes of the cell behaviour when exposed to different stimuli can put us on the track of mechanisms driving the different cell signalling paths (Laplante and Sabatini, 2009; Lavrik et al., 2005). As a result, parameters as the *hypoxic threshold* (Ayensa-Jiménez et al., 2020b) are replaced by richer behaviours, as the ones illustrated in Fig. 7.7.

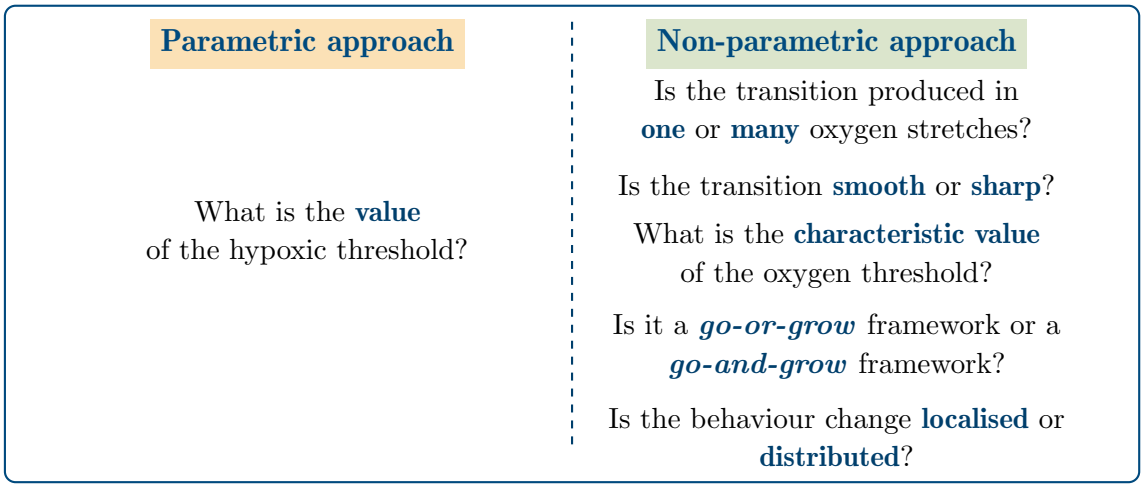


Figure 7.7: Parametric vs Non-parametric approaches: The degree of information about the cell metabolism is richer in the non-parametric approach.

Moreover, from a mechanistic perspective, different metabolic paths and schemes may be tested *in silico* using computational approaches (Kholodenko, 2006), in order to decide whether a path candidate is compatible with the metabolic changes discovered by means of the PGNNIV. Therefore, this knowledge on the macroscopic cell metabolic behaviour at the population level is important not only from an epistemic point of view, for modelling purposes, but also as a promising tool for molecular biologists, in their attempt to isolate and define the different signalling pathways, thus providing a deeper understanding of the underlying mechanisms.

Sometimes, there are some energetic constraints (the more fundamental ones are those given by the first and second principles of thermodynamics) that restrict the accessible states in a biological system (Benzinger, 1971; Dill and Bromberg, 2010; Haynie, 2001). These constraints are translated into macroscopic ones in a continuum population model. For instance, one possible constraint is the former hypothesis that $\Pi_{\text{gr}} + \Pi_{\text{go}} = 1$. However, this is a special case of the more general constraint $G(\Pi_{\text{gr}}, \Pi_{\text{go}}) = 0$, which could be founded on an energetic argument about the resources available for the cell to grow or proliferate. All these extra constraints may be incorporated in the PGNNIV framework in a direct and straightforward fashion either by expressing some relational equations between variables explicitly, or by adding appropriate penalty terms in the loss function obliging to fulfil a

mathematical constraint, such as $p\|G(\Pi_{\text{gr}}, \Pi_{\text{go}})\|^2$, with p a penalty parameter, as discussed in Chapter 5.

A last remark is that PGNNIV, as any method inspired in ANN, can be used as a universal approximator of the hidden interaction mechanisms between different cell populations, thus incorporating the ingredients of population sociology in systems biology (Ganesh et al., 2020). For instance, if many cell populations are considered, $C_1, \ldots, C_n$, one may establish many *ad-hoc* interrelation mechanisms, $\lambda = \lambda(C_1, \ldots, C_n)$, where λ is any model functional parameter, describing for instance migration or proliferation. The crosstalk between different cell populations has been demonstrated to be important in many cellular processes such as those presented here (Chen et al., 2020a; Oliveira et al., 2017). Of course, this interrelation will be properly learned if:

- The known physics and biology of the system are well enough described in terms of specific mathematical equations. That is, all known mechanisms are explicitly stated, but only them. This enables the PGNNIV to concentrate its unravelling power in the unknown interrelations.
- The available data is large enough to capture the prescribed dependency (as in the problem presented in this work). This is commonly difficult in many experimental sciences, particularly in biology, but new tools and trends such as microfluidics are promising in this regard.

As in any ML approach, care must be taken when interpreting the results and deriving conclusions, as the learning methods suffer from overfitting. A suitable strategy (train and test approaches, cross-validation, validation trials...) is therefore crucial for drawing generalisable conclusions.

7.4.2 The future: towards *in silico* personalised medicine

This work presents a method for going from cell expression at the tissue level, that is, the formation of cellular structures such as pseudopalisades in GBM invasion, to cell behaviour at the population level in response to the ambient stimuli. In one sense, it presents a link between clinicians and molecular biologists. A tumour biopsy may be extracted from one patient, cultivated and monitored in microfluidic devices where it may be exposed to different stimuli. The microfluidic devices have demonstrated to be capable of *in silico* reproducing histological features such as necrotic cores and pseudopalisades (Ayuso et al., 2016, 2017), which can be captured using image and video techniques. PGNNIV is a tool able to infer, from the culture response to several stimuli, the intrinsic model of the cell reaction to such stimuli. This ability to integrate the knowledge of the response to different stimuli is a particular capability of PGNNIV, which puts them one step ahead of other methods such as BINNs in the metabolic unravelling.

Once the intrinsic model is learned, its generalisation capability is much wider than the one offered by the histological features, as the latter is the response to very specific conditions (in mathematical terms, to very specific boundary conditions). In a sense, this strategy is the same as parametric fitting, except for the fact that there

is no need for making any assumption about the model functional structure, provided that we know the internal variables we want to relate. This last issue is not minor, but is, indeed, the main objective of biology research: to make scientific conclusions about the effect and association of chemical factors with biological response.

This extra generalisation capability, when combined with appropriate mathematical models, offers new possibilities in *in silico* drug and treatment design. Once the cell response to the different stimuli is unravelled, we may variate the different stimuli levels and the different conditions. Again, in mathematical terms, this is represented by boundary conditions, initial conditions and source terms of the associated PDEs. Consequently, we can evaluate, again from a tissue perspective, more than from a cell population point of view, the general histological features that happen in the cell virtual tissue, which is called a Virtual Digital Twin. These histological features are responsible of key factors in cancer progression such as vessel occlusion (Weis et al., 2004), intravasation, extravasation and metastasis (Chiang et al., 2016; Weis et al., 2004), necrosis and activation of inflammatory response and/or angiogenesis (Carmeliet and Jain, 2000; Greten and Grivennikov, 2019), in particular for GBM (Rong et al., 2006). Therefore, *in silico* tests will accelerate the design of new drugs and therapies as they allow to test the hypotheses in a more flexible, faster and cheaper way.

It is important to note that all the here described sequence of steps relies on one single patient-specific biopsy, thus turning all this approach fully patient-specific, grounding this approach within the global framework of personalised *in silico* medicine (Cirillo and Valencia, 2019). From a specific patient, we infer specific histological and cellular features and, therefore, the different *in silico* trials are adapted to their particular disease, running away from parametric models, whose universality is corseted by their particular functional form.

The whole process is illustrated in Fig. 7.8, where all the steps are summarised both in conceptual terms and using the particular example illustrated in this work. To conclude, PGNNIV is a tool for transferring the knowledge on the tissue response to knowledge on the cell gene expression. This knowledge is the one that enables to work on the cell expression level and to generalise both to arbitrary conditions and individual patients.

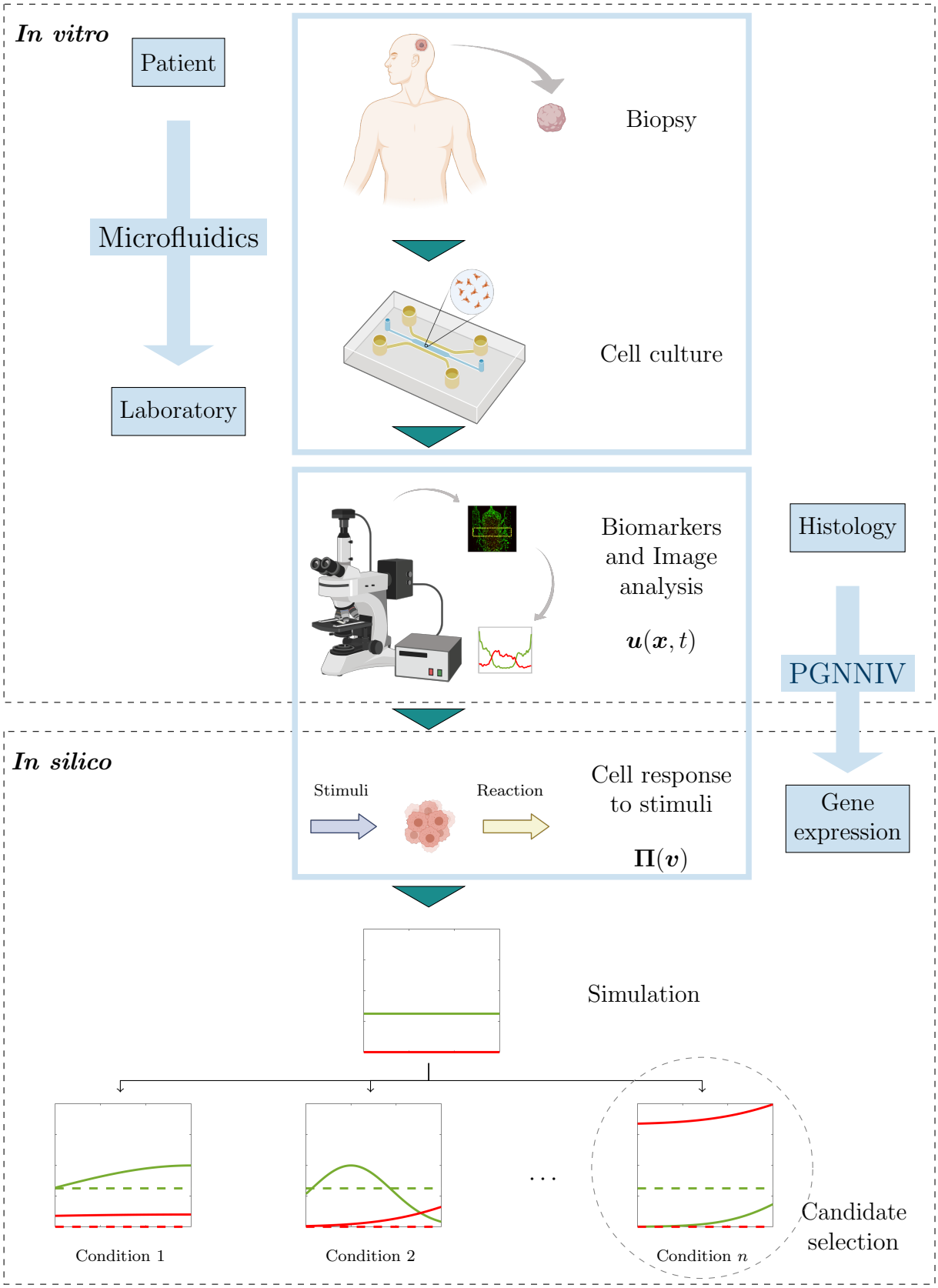


Figure 7.8: Summary of the described framework. Starting from the clinical patient, we recreate histological features associated with the tumoural tissue in the microfluidic devices, extract the cell response to stimuli using PGNNIV and use it to evaluate different therapies and drug candidates. Created with BioRender.com.

7.5 Conclusions

The emerging technology of microfluidics has brought to the field of cell culture not only the possibilities of biotechnology research in more realistic biomimetic environments but also the chance of incorporating data-intensive tools such as artificial intelligence (AI) and ML. The term coined for this framework is Intelligent Microfluidics.

Here, we have illustrated how PGNNIV may be a valuable tool to infer the cell response at the population level from the cell response at the tumour microenvironment level in response to external stimuli. Unlike other physically-informed data science (PIDS) methods, in PGNNIV the physics does not *constrain*, but only *guides* the network learning capacity, as the measured data may be supplied to endow the network with explanatory capacity. PGNNIV have demonstrated their unravelling capacity for different benchmark tested models, thus allowing us to work in model-free and non-parametric frameworks. Indeed, PGNNIV based simulations outperform the explanatory ability of any parametric model, except, at most, if the selected model belongs to the selected parametric family, which is, in practice, a strong and unrealistic assumption.

In addition, this explanatory ability is directly translated into an improvement of the prediction for different ambient conditions. The predictive ability does not only improve the one associated with classical parametric fitting but is also independent of the underlying model, thus making unnecessary the assumption of extra hypotheses, for example about cell metabolism. This improvement is achieved regardless of the environmental conditions, even if these ones are not used during the training process. In a certain sense, the ability of PGNNIV to predict out of the range of the training data-set is exploited here at its best.

The flexibility of PGNNIV allows that any information about biological systems may be incorporated totally or partially into the computations. This includes cell-cell interactions or cell-substrate interaction among other cues. In an opposite way, it is possible to focus the learning power on any relationship between fields of interest (biological, chemical, mechanical...) that is intended to be learned or quantified, both for theoretical (learn about the molecular metabolic processes) or for experimental (describe the main features of the process) purposes.

The presented methodology let us glimpse some steps in the direction of personalised medicine, as it is model-free, it allows to work with tissues extracted from different patients without the need of the specification of any particular model that would ruin out the method generalisation ability. Once characterised, the tumoural population may be *in silico* subjected to different stimuli and conditions, corresponding to different exploratory treatments. The response may be analysed from a clinician point of view, that is, at the tissue level, in order to evaluate the treatment success or failure. This strategy of *in silico* test-evaluation is quick and cheap and can strongly reduce animal experimentation, therefore facilitating research in areas such as biotechnology and biomedical engineering.

8. Mathematical analysis of the Glioblastoma evolution model

Contents

8.1 Introduction

As we have seen in previous chapters, when modelling the enormous variety and complexity of the phenomena that interact in the tumour micro-environment (TME), the resulting differential equations are highly non-linear and strongly coupled (Bearer et al., 2009; Eils and Kriete, 2013; Kitano, 2002). The numerical resolution of the equations, especially in the era of high performance computing, has helped a lot in the simulation of scenarios, the recreation of "what if" conditions and the study of effects and hypotheses individually, something that is often impossible to do with *in vivo* and *in vitro* models (Byrne et al., 2006; Katt et al., 2016). This new paradigm has established what is known as *in silico* experiments, which are more and more used in the early stages of designing drugs and therapies against tumours.

Nevertheless, the complexity of the equations to be solved often makes the use of numerical simulations impractical to address these problems, due to the high

computational cost, especially in the resolution of inverse problems, the design of experiments, model structural analysis or uncertainty quantification (UQ).

Although many modern techniques as reduced order model (ROM) and metamodels using artificial intelligence (AI) have been developed in recent years (Pérez-Aliacar et al., 2021), the existence of analytical solutions, although approximate, provides key information to understand problems and allow initial predictions of quantity of interests (QoIs) such as travelling wave fronts, equilibrium or ranges of variation of the solution and sensitivities, among others. Indeed, some works in the last years have focused on the use of these techniques for analysing glioblastoma (GBM) progression (Gerlee and Nelander, 2016; Pérez-García et al., 2011; Stepien et al., 2018).

In this chapter, we explore the dynamics of cell populations under gradients of a chemotactic agent. This general class of problems allows treating a wide variety of situations related to the evolution of tumours. In particular, the evolution of GBM, the most common and most aggressive primary brain tumour, is very much controlled by the local oxygen concentration and gradient. A mathematical description of the evolution of GBM was presented in Chapter 6, while in Chapter 7 we described a physically-informed data science (PIDS) method for accurately characterising the *go-or-grow* transition switch, using the functions Π_{go} and Π_{gr}.

The analysis of the system of partial differential equations (PDEs) that describes this problem makes it possible to quantify some histopathological characteristics, such as the spread of pseudopalisades, the population level near the vessel, or the response of the population to oscillatory stimuli. This knowledge can be used for the design of experiments, to speed up the characterisation processes of cell populations or to validate or rule out possible models.

First, we describe the structure of the mathematical problem considered and the general assumptions and hypotheses about the different involved phenomena. We derive some features about the solution field. For instance, we unveil the existence of migratory structures and a transition zone. We are able to estimate the wave front evolution and the shape of the solution profile far from the transition zone. Also, we compute an analytical solution for some specific cases, associated with particular relevant experimental situations. Next, we apply the general results to the GBM evolution problem, which has been analysed in the previous chapters, and now is faced from a purely mathematical point of view.

8.2 The general problem

Our aim in this chapter is to study the solutions of a broad class of problems that are related to the evolution of a cell culture under a chemotactic agent, when the concentration of the agent can be computed or measured. The equation to be analysed is:

$$u_t + (\alpha(x,t)u)_x = Du_{xx} + \beta(x,t)u(1-u), \tag{8.1}$$

with no flux boundary conditions:

$$Du_x - \alpha(x,t)u\big|_{x=0} = 0, \tag{8.2a}$$
$$Du_x - \alpha(x,t)u\big|_{x=L} = 0, \tag{8.2b}$$

and initial conditions:

$$u(x, t = 0) = u_0. \tag{8.3}$$

In the precedent equation, u is the field variable, usually related to a cell population concentration, α and β are functions related to the migrative and proliferative activity of the cell population and $D \in \mathbb{R}^+$ is a small parameter related to the cell random pedesis. Eq. (8.1) may be seen as the study of a cell population under heterogeneous chemotaxis and growth (such as when it depends on the concentration of an external agent). In later sections, we will apply the analysis to particular situations related to GBM populations.

8.2.1 Computation of the general solution for small diffusion

The main hypothesis, which is usually true for biological problems, is that $D \ll 1$, that is, random pedesis is neglected when compared to growth and agent-driven migration. In that case, Eq. (8.1) may be approximated by:

$$u_t + (\alpha(x, t)u)_x = \beta(x, t)u(1 - u), \tag{8.4}$$

The previous equation is a first-order hyperbolic PDE that may be faced using the method of characteristics. The method is briefly explained next.

8.2.1.1 Brief overview of the method of characteristics

The method of characteristics (Zachmanoglou and Thoe, 1986) is a technique for solving PDEs. It is commonly used to solve first-order equations, as the one given by Eq. (8.4), although it may be used for any hyperbolic PDE. The idea of the method is to transform a PDE into a family of ordinary differential equations (ODEs) along which the solution can be integrated from some initial data given on a suitable hypersurface. Even if the integration is not fully possible, this analysis may shed some light on the mathematical structure and nature of the solutions and help gain some qualitative insight (Debnath, 2011).

This method is applied to solve Eq. (8.4), which may be rewritten as:

$$P(t, x)u_t + Q(t, x)u_x = R(t, x, u), \tag{8.5}$$

with $P(t, x) = 1$, $Q(t, x) = \alpha(x, t)$ and $R(t, x, u) = u\,(\beta(x, t)(1 - u) - \alpha_x(x, t))$. We can look for the solution $u = f(t, x)$ as the one given by the implicit surface $\Sigma(x, t, u) := u - f(t, x) = 0$, which we will call the solution surface. The idea of the solving method is to generate Σ. The normal vector to the surface is given by $\boldsymbol{n} = \boldsymbol{\nabla}\Sigma = (-f_t, -f_x, 1)$. Defining the vector field $\boldsymbol{t} = (P, Q, R)$, then we have, according to Eq. (8.5):

$$\boldsymbol{t} \cdot \boldsymbol{n} = -Pf_t - Qf_x + R = -Pu_t - Qu_x + R = 0. \tag{8.6}$$

Therefore, our aim is to look for integral curves of $\boldsymbol{t}$, parametrised in terms of a given parameter τ. These are the so-called characteristic curves. As we have one

characteristic curve for each point of the initial condition, $u_0(x)$, we will need two parameters in order to fully describe parametrically the surface Σ:

$$t = t(\tau, s), \tag{8.7a}$$
$$x = x(\tau, s), \tag{8.7b}$$
$$u = u(\tau, s). \tag{8.7c}$$

The equations of the characteristic curves are, therefore:

$$\frac{dt}{d\tau} = P(t, x), \tag{8.8a}$$
$$\frac{dx}{d\tau} = Q(t, x), \tag{8.8b}$$
$$\frac{du}{d\tau} = R(t, x, u). \tag{8.8c}$$

It is possible to prove (Zachmanoglou and Thoe, 1986), at least under some regularity conditions[24], that the characteristic equations (Eqs. 8.8) define a family of curves Γ that lies entirely on Σ if they start from a point in Σ. Besides, the characteristic curves can only meet at critical points (where P and Q are both equal to 0).

8.2.1.2 Solution to the problem using the method of characteristics

If we know the initial data $u(0, x) = u_0(x)$, we need to define the initial value $(t(0), x(0), u(0))$ in order to ensure that each characteristic line belongs to the solution surface Σ, that is, for $t = 0$, $\tau = 0$ and $x = s$, we have $(t(0), x(0), u(0)) = (0, s, u_0(s))$. Also, there is another family of characteristic lines, emerging from the (t, x) points where $x = 0$ and $t > 0$. These points are related to the no flux boundary condition, Eq. (8.2). Assuming that $\alpha(0, t) \neq 0$, we conclude that $u = 0$. Therefore, for this family, $t = s$, $x = 0$ and $u = 0$. In other words, $(t(0), x(0), u(0)) = (s, 0, 0)$. At $t = 0$ and $x = 0$ there is an emerging singular characteristic that splits the domain in two regions. The geometric interpretation of the method of characteristics is shown in Fig. 8.1, which shows the projection of the characteristic curves onto the plane (t, x).

We have, therefore, two families of characteristic curves. For the first one:

$$\frac{dt}{d\tau} = 1, \quad t(0) = 0, \tag{8.9a}$$
$$\frac{dx}{d\tau} = \alpha(x, t), \quad x(0) = s, \tag{8.9b}$$
$$\frac{du}{d\tau} = u\left(\beta(x, t)(1 - u) - k\alpha_x(x, t)\right), \quad u(0) = u_0, \tag{8.9c}$$

and for the second:

$$\frac{dt}{d\tau} = 1, \quad t(0) = s, \tag{8.10a}$$
$$\frac{dx}{d\tau} = \alpha(x, t), \quad x(0) = 0, \tag{8.10b}$$
$$\frac{du}{d\tau} = u\left(\beta(x, t)(1 - u) - k\alpha_x(x, t)\right), \quad u(0) = 0. \tag{8.10c}$$

[24] We need to assume that $P(t, x)$ and $Q(t, x)$ are Lipschitz continuous in t and x and $R(y, x, u)$ is continuous and Lipschitz continuous in u.

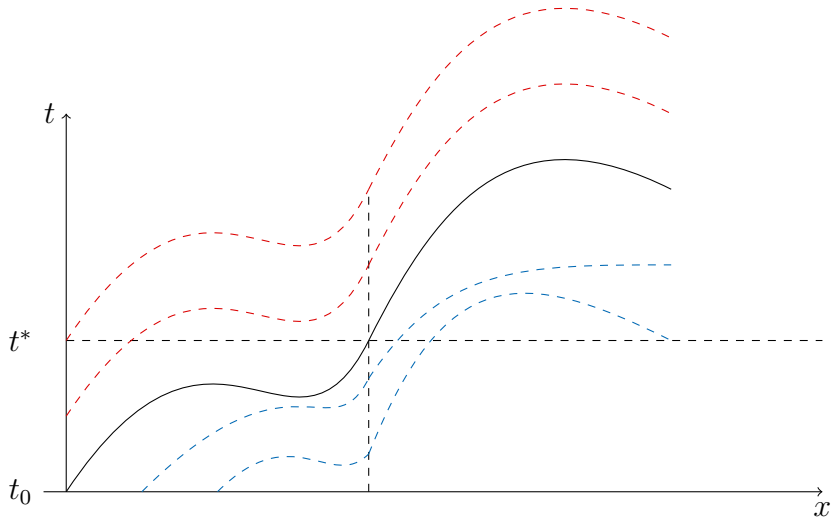


Figure 8.1: Projection of the characteristic curves. The two families of characteristic curves are shown in blue and red.

The second family of characteristic curves, given by Eqs. (8.10), has the trivial solution $u(x,t) = 0$. However, the solution to the first family, given by Eqs. (8.9), is more complex. We cannot do much more without extra assumptions, so we shall assume that α is separable in space and time, that is, we write $\alpha(x,t) = f(x)g(t)$. Then, Eqs. (8.9a) and (8.9b) may be integrated to obtain:

$$t = \tau, \tag{8.11}$$

$$\int_s^x \frac{dz}{f(z)} = \int_0^\tau g(u)\,du. \tag{8.12}$$

Eq. (8.12), although a little bit convoluted, is tractable for a broad family of solutions, as we will see later, and allows to express $x = F(t,s)$. This is not a minor observation, as $x^* = F(t,0)$ is the location of the transition at time t, which is very informative about the general trend of the solution. With respect to Eq. (8.9c), we proceed using the change of variable $v = 1/u$, we obtain the ODE in terms of the $\tau = t$ variable:

$$v'(\tau) + (\beta(x(\tau;s),\tau) - f'(x(\tau;s))g(\tau))\,v = \beta(x(\tau;s),\tau). \tag{8.13}$$

This equation is of the form $v' + p(\tau)v = q(\tau)$ for $p(\tau) = \beta(x(\tau;s),\tau) - f'(x(\tau;s))g(t)$ and $q(\tau) = \beta(x(\tau;s),\tau)$, so a general expression exists even if it is expressed by quadrature:

$$v(\tau,s) = \exp\left(-\int_0^\tau p(u,s)\,du\right)\left[\frac{1}{u_0} + \int_0^\tau q(u,s)\exp\left(\int_0^v p(v,s)\,dv\right)\,du\right]. \tag{8.14}$$

Recapping, suppose that Eq. (8.12) may be inverted to obtain $x = F(t,s)$ and $s = G(t,x)$. Then, plugging this expression into Eq. (8.14), we obtain a general expression for v and therefore for $u = u(x,t)$:

$$u(x,t) = \frac{u_0 \exp\left(\int_0^t p(u,s)\,du\right)}{1 + u_0\int_0^t q(u,s)\exp\left(\int_0^u p(v,s)\,dv\right)\,du}, \tag{8.15}$$

where:

$$\begin{aligned} p(u,s) &= \beta(F(u;s),u) - f'(F(u;s))g(u), \\ q(v,s) &= \beta(F(v;s),v), \\ s &= G(t,x). \end{aligned}$$

Consequently, it is enough to build a function that computes (8.15) for a given f, g, β, F and G. An example is given next using Matlab language:

```
1
2     function u = CompSol(x,t,u0,G,F,fp,g,BETA)
3
4
5     %%%% u = CompSol(x,t,u0,G,F,fp,g,BETA)
6     %
7     %    Function that computes the solution u at a time t
8     %    and a point x given the initial value u0.
9     %
10    %    INPUT:
11    %    - x: Space coordinate.
12    %    - t: Time coordinate.
13    %    - G: Inverse function relating the space x and parameter s.
14    %    - F: Function relating the space x and parameter s.
15    %    - fp: Derivative of the f function w.r.t. x.
16    %    - g: Function incorporating the time dependence in alpha.
17    %    - BETA: Beta function.
18    %
19    %    OUTPUT:
20    %    - u: Solution field.
21    %
22
23    N = 1e2; % For integration
24
25    % Parameter s
26    tau = t;
27    s = G(tau,x);
28    fun_P = @(tau) BETA(F(tau,s),tau) - fp(F(tau,s))g(tau);
29    fun_Q = @(tau) BETA(F(tau,s),tau);
30
31
32    uu = linspace(0,t,N);
33    FUN_Q = zeros(size(uu));
34    AUXILIAR = zeros(size(uu));
35    for i=1:length(uu)
36        FUN_Q(i) = fun_Q(uu(i));
37        AUXILIAR(i) = auxiliar(uu(i));
38    end
39    yy = FUN_Q.*exp(AUXILIAR);
40    T1 = trapz(uu,yy);
41    T2 =  exp(auxiliar(tau));
42
43    u = u0*T2/(1+T1*u0);
44
45    function Fu = auxiliar(u)
46        vv = linspace(0,u,N);
47        yy = fun_P(vv);
```

```
48          Fu = trapz(vv,yy);
49      end
50
51      end
```

List of Code Listings 8.1: Matlab function for the computation of the solution field.

8.2.2 Some particular cases of interest

The solution given by Eq. (8.15) is general up to the physical ($D \ll 1$) and mathematical (α separable) conditions specified. However, the evaluation of this equation involves one problematic step, namely the inversion of the characteristic equation given by Eq. (8.12) to obtain both functions F and G. The rest of the processes involving the solution computation are not difficult, except for cases where the integrals may have singularities. However, we are going to show that, for some relevant cases, it is possible to obtain a *semi-analytical* solution for u[25].

For instance, we may consider the following different cases:

- Cell culture is under a constant chemical gradient and the chemotaxis coefficient depends linearly on the value of the chemotactic agent itself. This is, for instance, the case of the GBM model discussed in Chapters 6 and 7 assuming $\Pi_{\text{gr}} \simeq \Pi_{\text{go}} \simeq 1$. For that case, α is constant. In that case we have:

$$\alpha(x,t) = ax + b. \tag{8.16}$$

- The chemotaxis coefficient does not depend on the value of the chemotactic agent itself and the chemical source is localised. In that case, the chemical profile at the stationary state is exponential and we have:

$$\alpha(x,t) = a\exp(-\lambda x). \tag{8.17}$$

- Cell culture is under an oscillating chemical gradient and the chemotaxis coefficient does not depend on the value of the chemotactic agent itself. Therefore:

$$\alpha(x,t) = (ax + b)\cos(\omega t). \tag{8.18}$$

8.2.2.1 Linear chemotaxis correction

We first consider a function of the type:

$$\alpha(x,t) = ax + b. \tag{8.19}$$

Then, we have $f(x) = ax + b$ and $g(t) = 1$. Eq. (8.12) writes:

$$\ln\left(\frac{ax+b}{as+b}\right) = a\tau, \tag{8.20}$$

[25] With the word *semi-analitical* we mean that the solution may be expressed by means of quadratures

so:

$$F(t,s) = \frac{1}{a}\left[(as+b)e^{at} - b\right],$$
$$G(t,x) = \frac{1}{a}\left[(ax+b)e^{-at} - b\right].$$

Hence, the transition is located at:

$$x^*(t) = \frac{b}{a}\left(1 - e^{at}\right). \tag{8.22}$$

Fig. 8.2 shows a comparison between the numerical results with $D = 1 \times 10^{-2}$ and the analytical solution for three different expressions of $\alpha(x)$, with $\beta(x,t) = 1$. In Fig. 8.3, the transition $x^* = x^*(t)$ and the value estimated for the value of the concentration at the front $u^* = u(x = x^*, t)$ are shown.

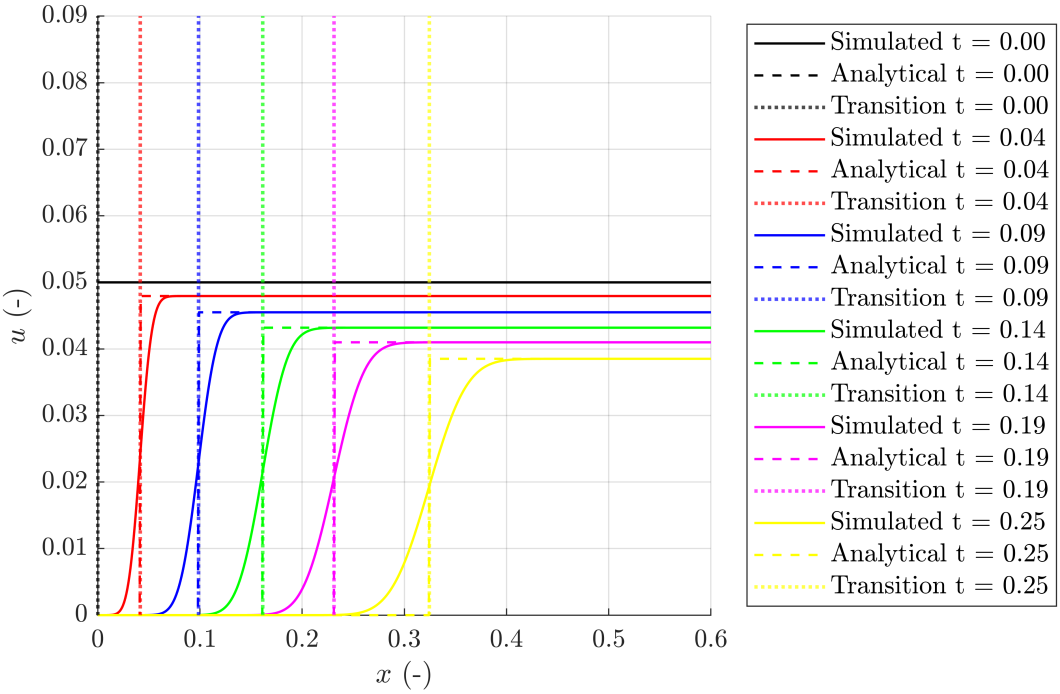


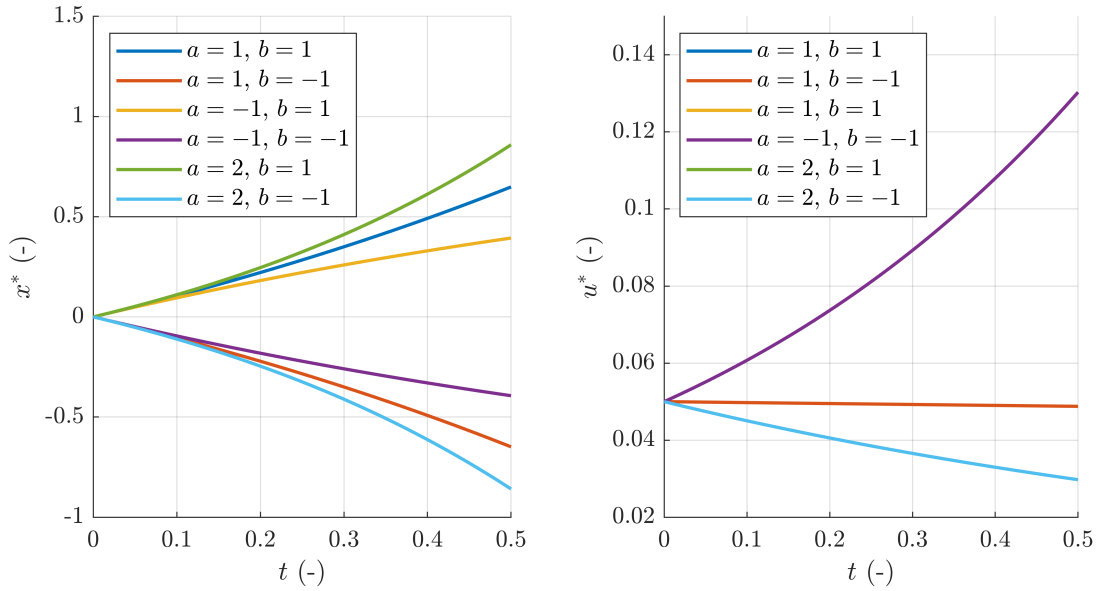
Figure 8.2: Comparison of numerical and analytical solutions for α linear. The analytical and simulated profiles are compared at different times, considering $a = 2$ and $b = 1$.

8.2.2.2 Polynomial chemotaxis correction

Now, we consider a function of the type:

$$\alpha(x,t) = ax^2 + bx + c. \tag{8.23}$$

Then, we have $f(x) = ax^2 + bx + c$ and $g(t) = 1$.



(a) Transition location $x^* = x^*(t)$.

(b) Cell concentration at $u^* = u^*(t)$.

Figure 8.3: Location of the transition and cell concentration at the transition for α linear. The evolution of the cell concentration and of the transition depends on the values of a and b.

- If $\Delta = b^2 - 4ac = 0$, Eq. (8.12) writes:

$$\frac{1}{2as+b} - \frac{1}{2ax+b} = \frac{1}{2}\tau, \tag{8.24}$$

so:

$$F(t,s) = \frac{4as + 2abst + b^2 t}{4a - 2bat - 4a^2 ts}, \tag{8.25a}$$

$$G(t,x) = \frac{4ax - 2abtx - b^2 t}{4a^2 tx + 2abt + 4a}. \tag{8.25b}$$

The transition is located at:

$$x^*(t) = \frac{b^2 t}{4a - 2bat}. \tag{8.26}$$

- If $\Delta = b^2 - 4ac < 0$, Eq. (8.12) writes:

$$\left(\arctan\left(\frac{2ax+b}{\sqrt{\Delta}}\right) - \arctan\left(\frac{2as+b}{\sqrt{\Delta}}\right)\right) = \frac{1}{2}\sqrt{\Delta}\tau, \tag{8.27}$$

so:

$$F(t,s) = \frac{1}{2a}\left[\sqrt{-\Delta}\tan\left(\frac{1}{2}\sqrt{-\Delta}t + \arctan\frac{2as+b}{\sqrt{-\Delta}}\right) - b\right], \tag{8.28a}$$

$$G(t,x) = \frac{1}{2a}\left[\sqrt{-\Delta}\tan\left(-\frac{1}{2}\sqrt{-\Delta}t + \arctan\frac{2ax+b}{\sqrt{-\Delta}}\right) - b\right]. \tag{8.28b}$$

The transition is located at:

$$x^*(t) = \frac{1}{2a}\left[\sqrt{-\Delta}\tan\left(\frac{1}{2}\sqrt{-\Delta}t + \arctan\frac{b}{\sqrt{-\Delta}}\right) - b\right]. \tag{8.29}$$

- If $\Delta = b^2 - 4ac > 0$, Eq. (8.12) writes:

$$\ln\left(\frac{2ax+b-\sqrt{\Delta}}{2ax+b+\sqrt{\Delta}}\right) - \ln\left(\frac{2as+b-\sqrt{\Delta}}{2as+b+\sqrt{\Delta}}\right) = \sqrt{\Delta}\tau, \tag{8.30}$$

so:

$$F(t,s) = \frac{1}{2a}\left[\frac{\gamma^+(2as+\gamma^-)\exp(\frac{1}{2}\sqrt{\Delta}t) - \gamma^-(2as+\gamma^+)\exp(-\frac{1}{2}\sqrt{\Delta}t)}{\gamma^+\exp(-\frac{1}{2}\sqrt{\Delta}t) - \gamma^-\exp(\frac{1}{2}\sqrt{\Delta}t)}\right], \tag{8.31a}$$

$$G(t,x) = \frac{1}{2a}\left[\frac{\gamma^+(2as+\gamma^-)\exp(-\frac{1}{2}\sqrt{\Delta}t) - \gamma^-(2as+\gamma^+)\exp(\frac{1}{2}\sqrt{\Delta}t)}{(\gamma^+)\exp(\frac{1}{2}\sqrt{\Delta}t) - \gamma^-)\exp(-\frac{1}{2}\sqrt{\Delta}t)}\right], \tag{8.31b}$$

where we have defined $\gamma^+ = b+\sqrt{\Delta}$ and $\gamma^- = b-\sqrt{\Delta}$. The transition is located at:

$$x^*(t) = 2c\left[\frac{\exp(\frac{1}{2}\sqrt{\Delta}t) - \exp(-\frac{1}{2}\sqrt{\Delta}t)}{\gamma^+\exp(\frac{1}{2}\sqrt{\Delta}t) - \gamma^-\exp(-\frac{1}{2}\sqrt{\Delta}t)}\right]. \tag{8.32}$$

Fig. 8.4 shows a comparison between the numerical results with $D = 1\times10^{-2}$ and the analytical solution for three different expressions of $\alpha(x)$ and $\beta(x,t) = 1$. In Fig. 8.5, the transition $x^* = x^*(t)$ and the value estimated for the concentration at the front $u^* = u(x = x^*, t)$ are shown.

8.2.2.3 Exponential chemotaxis correction

We consider now a function of the type:

$$\alpha(x,t) = a\exp(-\lambda x). \tag{8.33}$$

Then, we have $f(x) = a\exp(-\lambda x)$ and $g(t) = 1$. Eq. (8.12) writes now:

$$\exp(\lambda x) - \exp(\lambda s) = a\lambda\tau, \tag{8.34}$$

so:

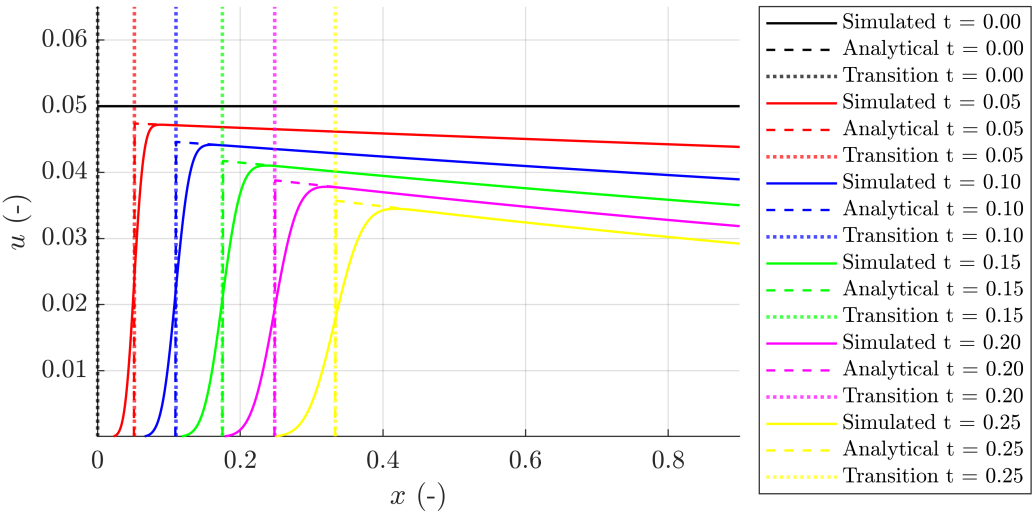
$$F(t,s) = \frac{1}{\lambda}\ln(a\lambda t + e^{\lambda s}), \tag{8.35a}$$

$$G(t,x) = \frac{1}{\lambda}\ln(-a\lambda t + e^{\lambda x}). \tag{8.35b}$$

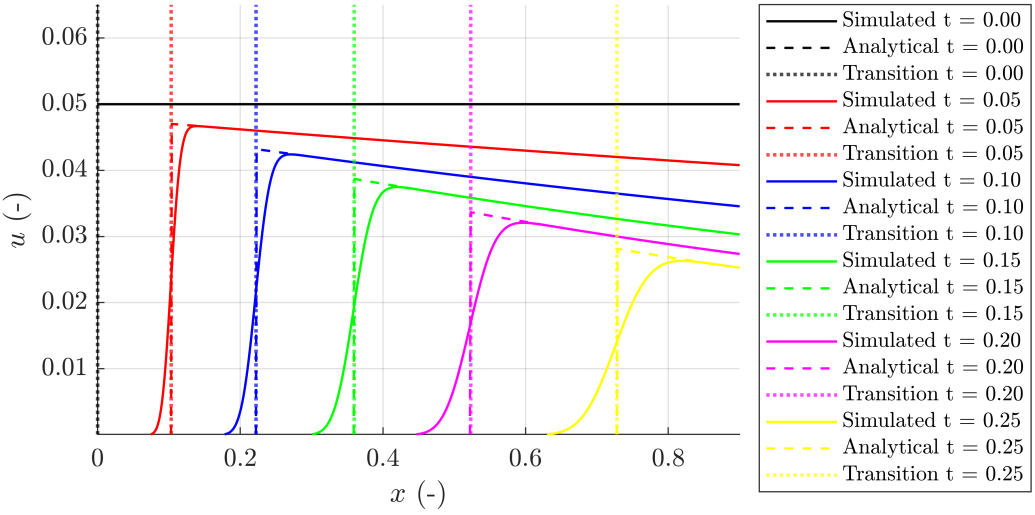
The transition is located at:

$$x^*(t) = \frac{1}{\lambda}\ln(1+\lambda a t). \tag{8.36}$$

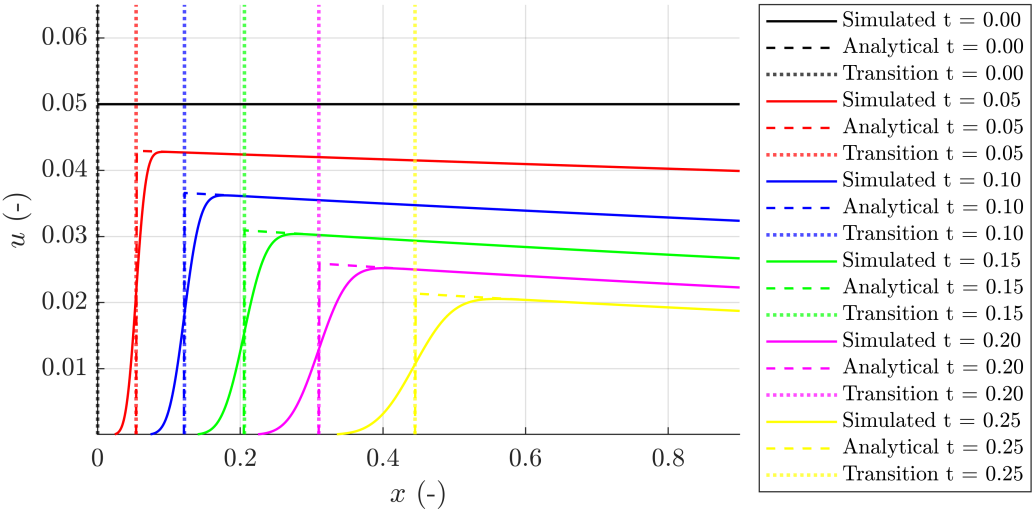
Fig. 8.6 shows the comparison between the numerical results with $D = 1\times10^{-3}$ and the analytical solution and $\beta(x,t) = 1$. Fig. 8.7 shows the transition $x^* = x^*(t)$ and the value estimated for the concentration at the front $u^* = u(x = x^*, t)$.



(a) $a = 1$, $b = 2$, $c = 1$ ($\Delta < 0$).

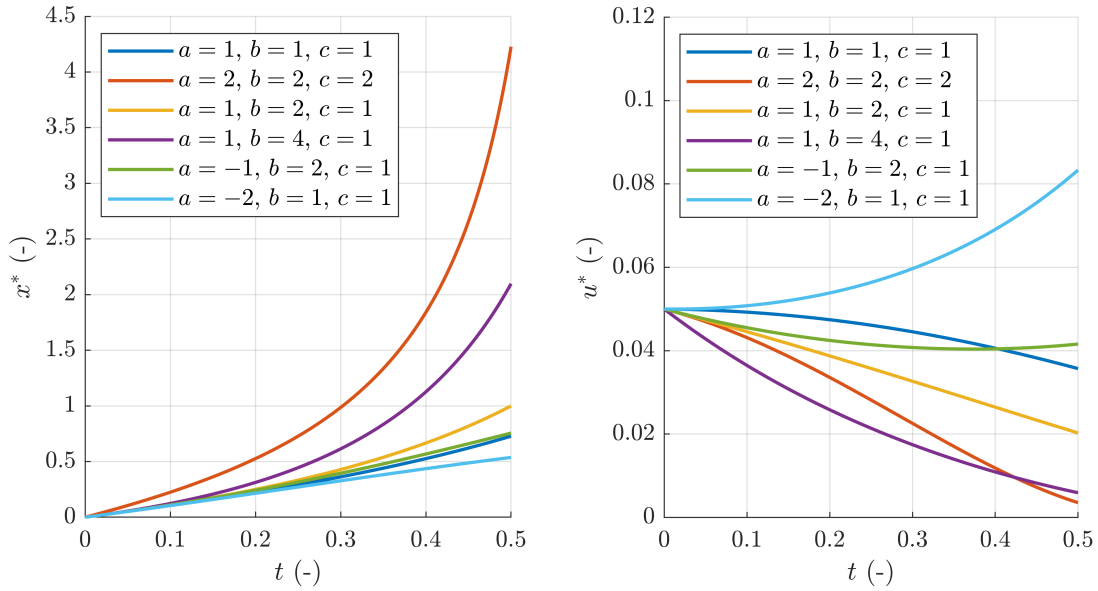


(b) $a = 2$, $b = 2$, $c = 2$ ($\Delta = 0$).



(c) $a = 1$, $b = 4$, $c = 1$ ($\Delta > 0$).

Figure 8.4: Comparison of numerical and analytical solutions for α polynomic. The analytical and simulated profiles at different times are compared for three different α expressions.



(a) Transition location $x^* = x^*(t)$.

(b) Cell concentration at $u^* = u^*(t)$.

Figure 8.5: Location of the transition and cell concentration at the transition for α polynomic. The evolution of the cell concentration and of the transition depends strongly on the values of a, b and c.

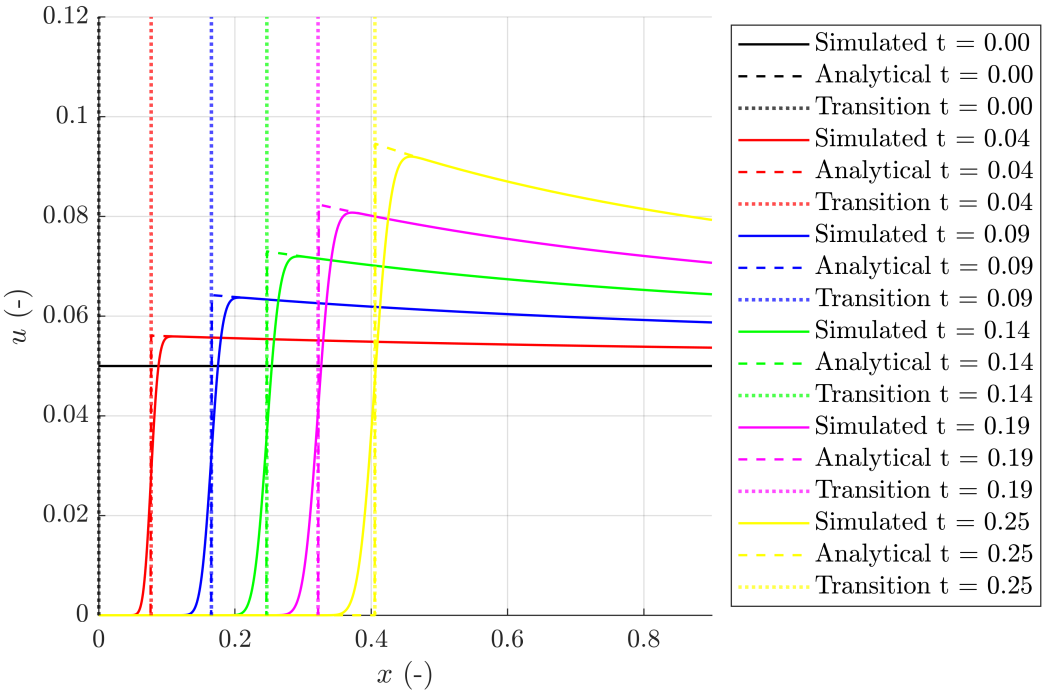
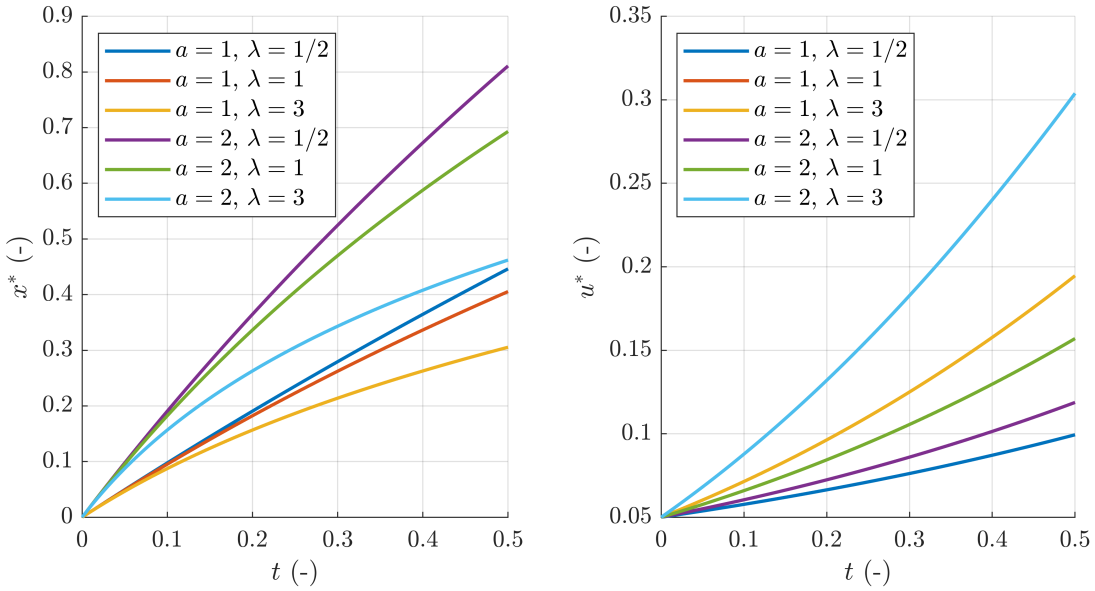


Figure 8.6: Comparison of numerical and analytical solutions for α exponential. The analytical and simulated profiles are compared at different times, considering $a = 2$ and $\lambda = 1$.



(a) Transition location $x^* = x^*(t)$.

(b) Cell concentration at $u^* = u^*(t)$.

Figure 8.7: Location of the transition and cell concentration at the transition for α exponential. The evolution of the cell concentration and of the transition depends on the values of a and λ.

8.2.2.4 Oscillating linear chemotaxis correction

Now, we consider a function of the type:

$$\alpha(x,t) = (ax+b)\cos(\omega t). \tag{8.37}$$

Then, we have $f(x) = ax + b$ and $g(t) = \cos(\omega t)$. Eq. (8.12) writes now:

$$\ln(ax+b) - \ln(as+b) = \frac{a}{\omega}\sin(\omega t), \tag{8.38}$$

so:

$$\begin{aligned} F(t,s) &= \frac{(as+b)\exp\left(\frac{a}{\omega}\sin(\omega t)\right) - b}{a}, \\ G(t,x) &= \frac{(ax+b)\exp\left(-\frac{a}{\omega}\sin(\omega t)\right) - b}{a}. \end{aligned} \tag{8.39a}$$

The transition is located at:

$$x^*(t) = \frac{b}{a}\left[\exp\left(\frac{a}{\omega}\sin(\omega t)\right) - 1\right]. \tag{8.40}$$

Note that, if in Eqs. (8.39) and (8.40) we use that $\sin(x) \sim x$ for $x \sim 0$, we recover Eqs. (8.21) and (8.22). Fig. 8.8 shows the comparison between the numerical results with $D = 1 \times 10^{-2}$ and the analytical solution, with $\beta(x,t) = 1$. Fig. 8.9 shows the transition $x^* = x^*(t)$ and the value estimated for the concentration at the front $u^* = u(x = x^*, t)$.

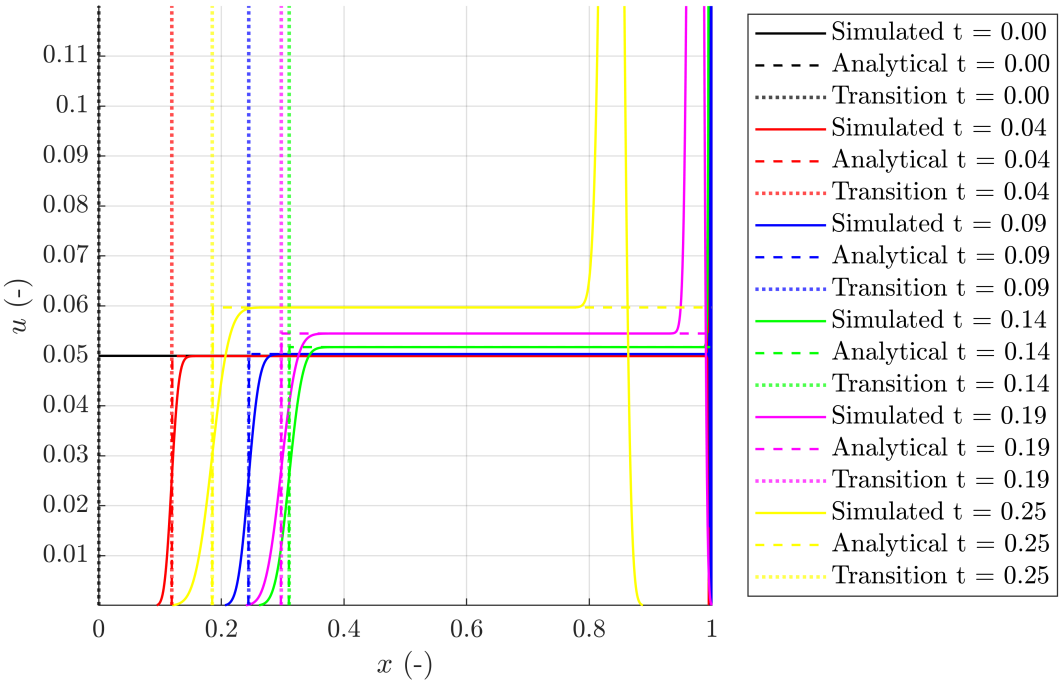
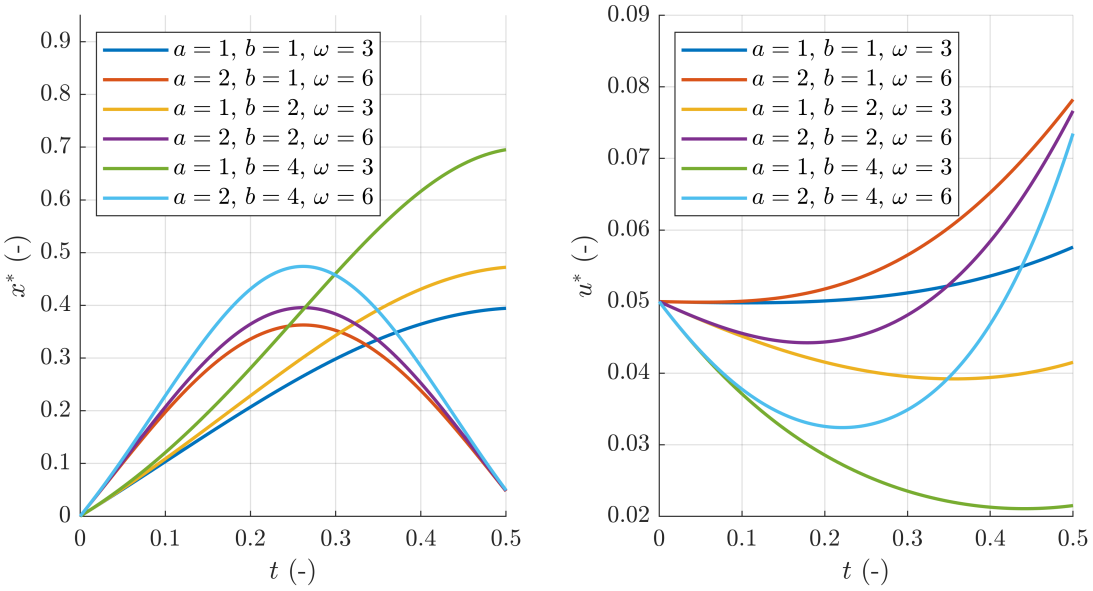


Figure 8.8: Comparison of numerical and analytical solutions for α oscillating gradient. The analytical and simulated profiles at different times are compared, considering $a = 1$, $b = 3$, $\omega = 10$. Note that only the wave propagating from left to right is taken into account, although the propagating wave from right to left may be derived analogously.



(a) Transition location $x^* = x^*(t)$.

(b) Cell concentration at $u^* = u^*(t)$.

Figure 8.9: Location of the transition and cell concentration at the transition for α oscillating gradient. The evolution of the cell concentration and of the transition depends on the values of a and b and ω.

The oscillating gradient case is very interesting as it corresponds to the case where the oxygenation feed between the two channels at the microfluidic device switches, so we shall explore it in more detail. We have obtained the general solution:

$$u(x,t) = \frac{u_0 \exp\left(\int_0^t p(u,s)\,du\right)}{1 + u_0 \int_0^t q(u,s) \exp\left(\int_0^u p(v,s)\,dv\right)\,du}, \tag{8.41}$$

where, for the oscillating gradient case:

$$\begin{aligned} p(u,s) &= \beta(F(u;s),u) - a\cos(\omega u), \\ q(v,s) &= \beta(F(v;s),v), \\ s &= G(t,x), \end{aligned}$$

and F and G are given by the expressions Eqs. (8.39). For some specific regimes, it is possible to derive approximate solutions to Eq. (8.43), that is, more manageable expressions than Eq. (8.41), which is expressed by a quadrature and therefore may be somewhat complex, for example, in inverse problems.

We observe that, for this case, Eq. (8.13) is:

$$v' + (\beta(x(t;s),t) - a\cos(\omega t))\,v = \beta(x(t;s),t), \quad v(0) = v^*. \tag{8.42}$$

The variable u is next obtained using the fact that $u = 1/v$. A slight simplification is to consider $\beta(x,t)$, that is, the growth term is independent of x and t, so Eq. (8.42) writes now:

$$v' + (\beta - a\cos(\omega t))\,v = \beta, \quad v(0) = v^*. \tag{8.43}$$

Next, we explore four different regimes:

- Slow variations of the gradients, $\omega \ll 1$.
- Fast variations of the gradients, $\omega \gg 1$.
- Dominant chemotaxis, $\beta \ll a$.
- Dominant growth, $a \ll \beta$.

Slow variations of the gradients, $\omega \ll 1$.

Let $T = \omega t$ and let us assume that $v(t) = f(\omega t) + w$. Then, Eq. (8.43) becomes:

$$\omega f'(\omega t) + \frac{dw}{dt} + (\beta - a\cos(\omega t))\,f(\omega t) + (\beta - a\cos(\omega t))\,u = \beta. \tag{8.44}$$

Besides, let $\omega f'(\omega t) + (\beta - a\cos(\omega t))\,f(\omega t) = \beta$, so we have:

$$\omega f'(T) + (\beta - a\cos(T))\,f(T) = \beta. \tag{8.45}$$

Since $\omega \ll 1$, we can approximate the solution of Eq. (8.45) to:

$$f(T) = \frac{\beta}{\beta - a\cos(T)} + \mathcal{O}(\omega). \tag{8.46}$$

Then, Eq. (8.44) writes:

$$\frac{dw}{dt} + (\beta - a\cos(\omega t))\, w = 0, \tag{8.47}$$

and the initial condition is:

$$w^* = w(t=0) = v(t=0) - f(0) = v^* - \frac{\beta}{\beta - a}. \tag{8.48}$$

As we have a slow modulation of the frequency/decay rate, we invoke the Wentzel-Kramers-Brillouin-Jeffreys (WKBJ) method (Olver, 1997). In terms of $T = \omega t$, Eq. (8.47) becomes:

$$\omega\frac{dw}{dT} + (\beta - a\cos(T))\, w = 0, \quad w(0) = w^*. \tag{8.49}$$

The WKBJ approximation is expressed here as:

$$w = p\exp\left(\frac{\phi(T)}{\omega}\right) J(T), \quad J(T) = J_0 + \omega J_1 + \mathcal{O}(\omega^2). \tag{8.50}$$

Plugging Eq. (8.50) into Eq. (8.47) and operating we obtain:

$$p\exp\left(\frac{\phi(T)}{\omega}\right)\left[\omega\left(\frac{\dot\phi}{\omega}J + \dot J\right) + (\beta - a\cos T)J\right] = 0. \tag{8.51}$$

Therefore, $\omega\left(\frac{\dot\phi}{\omega}J + \dot J\right) + (\beta - a\cos T)J = 0$, so:

$$\dot\phi\,(J_0 + \omega J_1 + \ldots) + \omega\left(\dot J_0 + \omega\dot J_1 + \ldots\right) + (\beta - a\cos T)\,(J_0 + \omega J_1 + \ldots) = 0. \tag{8.52}$$

The $\mathcal{O}(1)$ corresponding equation is:

$$J_0\left(\dot\phi + (\beta - a\cos T)\right) = 0, \tag{8.53}$$

and solving it for ϕ:

$$\phi(T) = \phi^* - \beta T + a\sin T. \tag{8.54}$$

The $\mathcal{O}(\omega)$ corresponding equation is:

$$J_1\left(\dot\phi + (\beta - a\cos T)\right) + \dot J_0 = 0, \tag{8.55}$$

so, as $\dot\phi + (\beta - a\cos T) = 0$, we obtain $J_0 = J_0^*$ (constant).

Consequently, Eq. (8.50) becomes:

$$w = K\exp\left(\frac{-\beta T + a\sin T}{\omega}\right), \tag{8.56}$$

for $K = p\exp(\frac{\phi^*}{\omega})J_0$ constant. Using that $T = \omega t$ and the initial value $w(0) = w^*$, we obtain the approximation:

$$w = \left(v^* - \frac{\beta}{\beta - a}\right) \exp\left(-\beta t + \frac{a}{\omega}\sin(\omega t)\right). \tag{8.57}$$

Finally, as $u = w + f(\omega t)$:

$$v \sim \frac{\beta}{\beta - a\cos(\omega t)} + \left(v^* - \frac{\beta}{\beta - a}\right) \exp\left(-\beta t + \frac{a}{\omega}\sin(\omega t)\right). \tag{8.58}$$

Fig. 8.11 shows the comparison between the approximation computed using the WKBJ method and the numerical solution obtained using a standard Runge-Kutta solver.

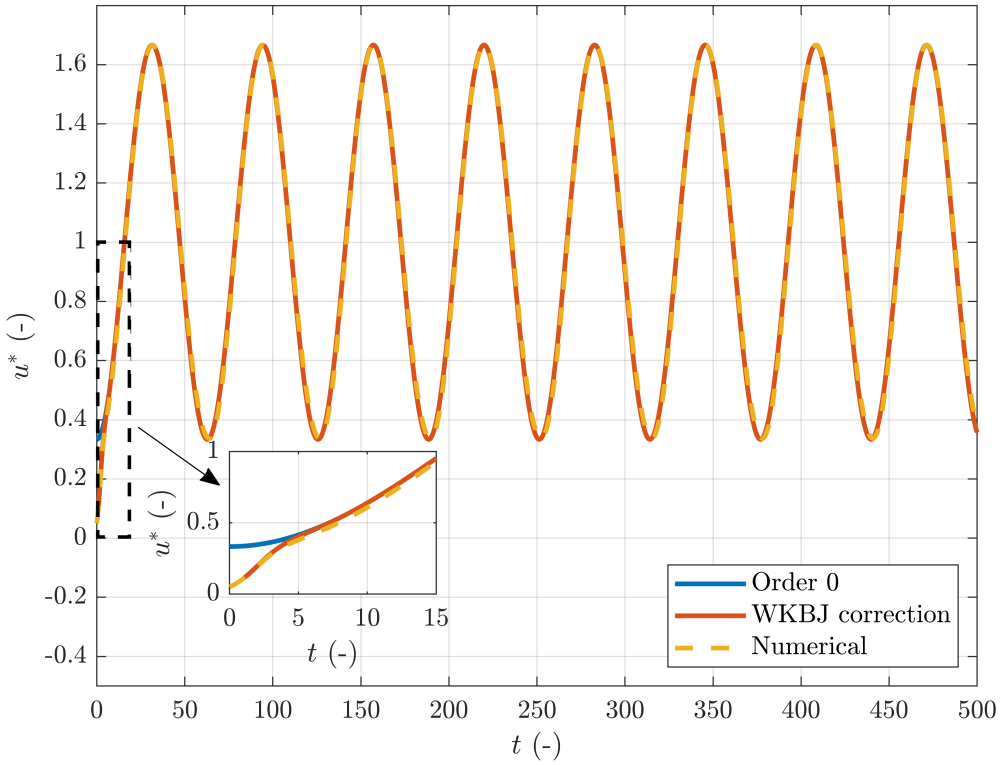


Figure 8.10: Solution for $\omega \ll 1$. The numerical solution is compared with the analytical solution obtained using WKBJ approximation, when $a = 2$, $\beta = 3$ and $\omega = 0.1$.

Fast variations of the gradients, $\omega \gg 1$.

We solve now the problem using the method of multiple scales. Let us assume that $v(t) = v(T_1, T_2)$, where $T_1 = t$ and $T_2 = \omega t$. Then:

$$\frac{dv}{dt} = \frac{\partial v}{\partial T_1}\frac{\partial T_1}{\partial t} + \frac{\partial v}{\partial T_2}\frac{\partial T_2}{\partial t} = \frac{\partial v}{\partial T_1} + \omega\frac{\partial v}{\partial T_2}. \tag{8.59}$$

If $\varepsilon = \frac{1}{\omega} \ll 1$, Eq. (8.43) writes:

$$\varepsilon \left(\frac{\partial v}{\partial T_1} + \frac{\partial v}{\partial T_2} \right) + \varepsilon \left(\beta - a \cos(T_2) \right) v = \varepsilon \beta. \tag{8.60}$$

If we use an asymptotic expansion of v, $v = v_0 + \varepsilon v_1$ we obtain:

$$\varepsilon \left(\frac{\partial v_0}{\partial T_1} + \varepsilon \frac{\partial v_1}{\partial T_1} \right) + \frac{\partial v_0}{\partial T_2} + \varepsilon \frac{\partial v_1}{\partial T_2} + \varepsilon \left(\beta - a \cos(T_2) \right) (v_0 + \varepsilon v_1) + \mathcal{O}(\varepsilon^2) = \varepsilon \beta. \tag{8.61}$$

Solving the equation obtained collecting the $\mathcal{O}(1)$ terms, we arrive to:

$$v_0 = f(T_1); \quad f(0) = v^*. \tag{8.62}$$

Now, for the equation obtained collecting the $\mathcal{O}(\varepsilon)$ terms, we arrive to:

$$f'(T_1) + \frac{\partial v_1}{\partial T_2} + (\beta - a \cos(T_2)) f(T_1) = \beta. \tag{8.63}$$

As v_1 is a periodic correction, integrating Eq. (8.63) in $[0; 2\pi]$ we obtain:

$$2\pi f'(T_1) + 2\pi \beta f(T_1) = 2\pi \beta, \tag{8.64}$$

so, solving for f:

$$f(T_1) = v^* e^{-\beta T_1} + (1 - e^{-\beta T_1}). \tag{8.65}$$

The leading order approximation is:

$$v(t) \sim v^* e^{-\beta t} + (1 - e^{-\beta t}). \tag{8.66}$$

Now, for the $\mathcal{O}(\varepsilon)$ equation we have:

$$\frac{\partial v_1}{\partial T_2} = a \cos(T_2) v_0(T_1). \tag{8.67}$$

Since $v_1(T_1, T_2 = 0) = 0$,

$$v_1(T_1, T_2) = a \sin(T_2) v_0(T_1). \tag{8.68}$$

and therefore the first order correction is, $v_0 + \varepsilon v_1$, that is:

$$v(t) \sim \left(v^* e^{-\beta t} + (1 - e^{-\beta t}) \right) \left(1 + \frac{a}{\omega} \sin(\omega t) \right). \tag{8.69}$$

Fig. 8.11 shows the comparison between the leading order approximation (order 0), the first order correction (order 1) and the numerical solution obtained using a standard Runge-Kutta solver.

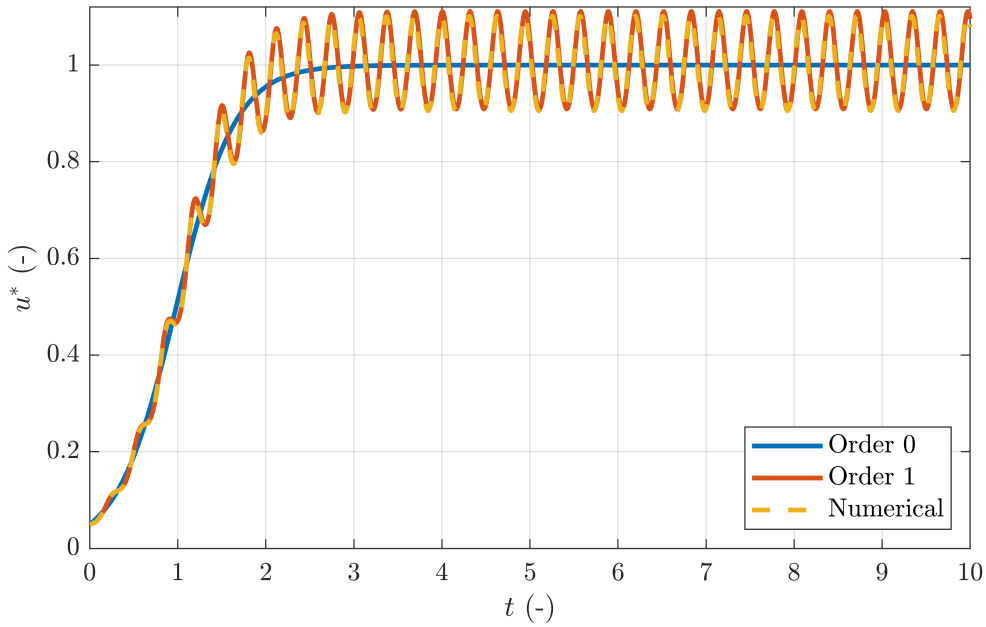


Figure 8.11: Solution for $\omega \gg 1$. The numerical solution is compared with the analytical solution obtained using the method of multiple scales (order 0 and order 1 terms), when $a = 2$, $\beta = 3$ and $\omega = 20$.

Dominant chemotaxis, $\beta \ll a$.

We solve now the problem using the standard asymptotic expansion method. We set $\varepsilon = \frac{\beta}{a} \ll 1$. Then, Eq. (8.43) writes:

$$v' + a\left(\varepsilon - \cos(\omega t)\right) v = a\varepsilon. \tag{8.70}$$

The leading order solution is obtained immediately as it is the solution to the homogeneous linear differential equation:

$$v_0' - a\cos(\omega t) v_0 = 0. \tag{8.71}$$

The solution becomes:

$$v_0 = v^* e^{\frac{1}{\omega} a \sin(\omega t)}. \tag{8.72}$$

Then, the leading order approximation is:

$$v \sim v^* e^{\frac{1}{\omega} a \sin(\omega t)}. \tag{8.73}$$

Fig. 8.12 shows the comparison between the leading order approximation (order 0) and the numerical solution obtained using a standard Runge-Kutta solver. Note that this solution is only valid for t small. When $t \gg 1$ extra correction terms are needed. However, these extra terms do not have a straightforward analytical expression.

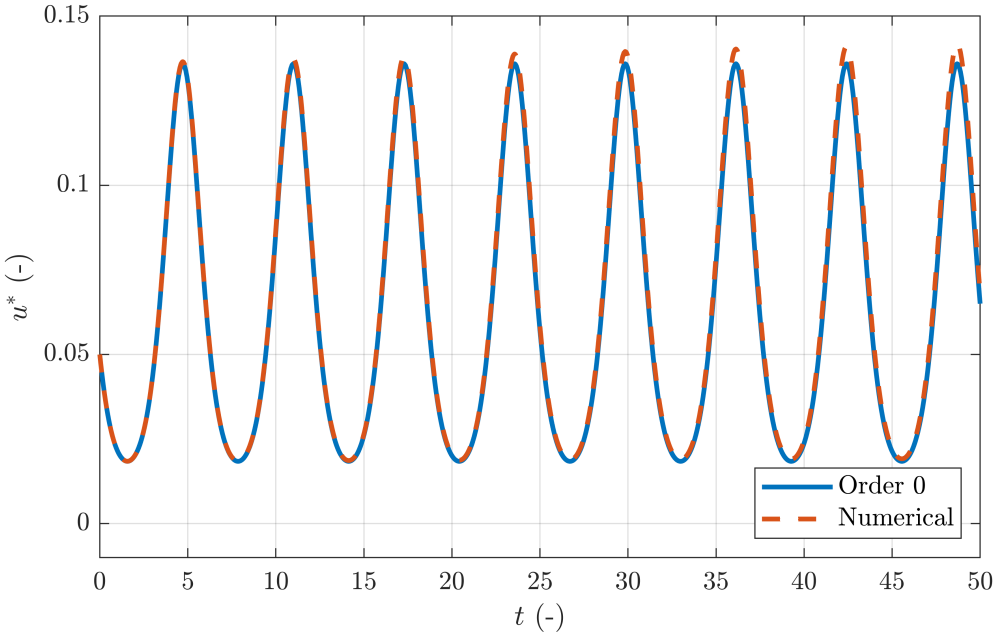


Figure 8.12: Solution for $\beta \ll a$. The numerical solution is compared with the leading order term of the asymptotic expansion, when $a = 1$, $\beta = 0.001$ and $\omega = 1$.

Dominant growth, $a \ll \beta$.

Again, we solve the problem using the standard asymptotic expansion method. Now, we set $\varepsilon = \frac{A}{\beta} \ll 1$. Then, Eq. (8.43) writes:

$$v' + \beta\left(1 - \varepsilon\cos(\omega t)\right)v = \beta. \tag{8.74}$$

The leading order solution is obtained immediately as it is the solution to the inhomogeneous linear differential equation:

$$v_0' + \beta v_0 = \beta. \tag{8.75}$$

The solution is:

$$v_0 = v^* e^{-\beta t} + \left(1 - e^{-\beta t}\right). \tag{8.76}$$

So, the leading order approximation is:

$$v \sim v^* e^{-\beta t} + \left(1 - e^{-\beta t}\right). \tag{8.77}$$

With respect to the correction term, an asymptotic expansion of v of the form $v = v_0 + \varepsilon v_1$ leads to the inhomogeneous first order differential equation:

$$v_1' + \beta v_1 = \beta\cos(\omega t)v_0. \tag{8.78}$$

The solution to this ODE with initial condition $v_1(0) = 0$ is:

$$v_1(t) = \frac{\beta\left[(v^* - 1)\gamma^2\sin(\omega t) + \omega^2 e^{\beta t}\sin(\omega t) + \omega\beta e^{\beta t}\cos(\omega t)\right]}{\omega\gamma^2 e^{\beta t}} - \frac{\beta^2}{\gamma^2}e^{-\beta t}, \tag{8.79}$$

where we have defined $\gamma^2 = \beta^2 + \omega^2$. Hence the first order correction is $v_0 + \varepsilon v_1$, which yields:

$$v(t) \sim 1 + \frac{(v^* - 1)\gamma^2\omega - A\left[\left(\gamma^2 v^* - \gamma + \omega^2 e^{\beta t}\right)\sin(\omega t) + \omega\beta e^{\beta t}\cos(\omega t) - \beta\omega\right]}{\omega\gamma^2 e^{\beta t}}. \tag{8.80}$$

Fig. 8.13 shows the comparison between the leading order approximation (order 0), the first order approximation (order 1) and the numerical solution obtained using a standard Runge-Kutta solver.

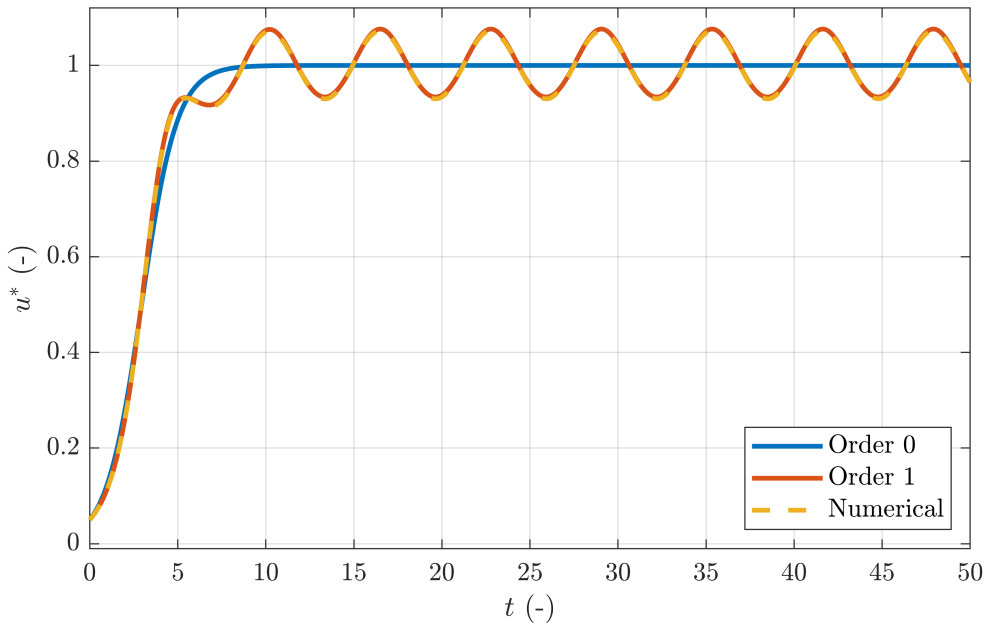


Figure 8.13: Solution for $a \ll \beta$. The numerical solution is compared with the leading order term of the asymptotic expansion, when $a = 0.01$, $\beta = 1$ and $\omega = 1$.

8.3 Application: glioblastoma culture under hypoxic conditions

Let us now consider the parametric model of GBM evolution presented in Chapters 6 and 7 that we reproduce here with a slightly different notation:

$$\frac{\partial C_n}{\partial T} = \frac{\partial}{\partial X}\left(D_1 \frac{\partial C_n}{\partial X} - \chi \Pi_{\text{go}}(O_2) F_{\text{go}}(C_n, C_d) C_n \frac{\partial O_2}{\partial X}\right) + \\ + \alpha_1 \Pi_{\text{gr}}(O_2) F_{\text{gr}}(C_n, C_d) C_n - \alpha_{12} \Pi_{\text{d}}(O_2) C_n, \tag{8.81a}$$

$$\frac{\partial C_d}{\partial T} = \alpha_{12} F_{\text{d}}(O_2) C_n, \tag{8.81b}$$

$$\frac{\partial O_2}{\partial T} = \frac{\partial}{\partial X}\left(D_2 \frac{\partial O_2}{\partial X}\right) - \alpha_2 \Pi_{\text{c}}(O_2) C_n. \tag{8.81c}$$

where C_n and C_d are respectively the alive and dead cell concentrations, F_{go} and F_{gr} are functions accounting for the saturation effect on the migration and growth, Π_{d} is the function accounting for the effect of the oxygen in cell death, Π_{gr} and Π_{gr} are the *go-or-grow* correction functions and Π_{c} is the nonlinear correction of the oxygen consumption. The boundary conditions considered are:

$$\left.\frac{\partial f_n}{\partial x}\right|_{x=0} = 0, \tag{8.82a}$$

$$\left.\frac{\partial f_n}{\partial x}\right|_{x=L} = 0, \tag{8.82b}$$

$$O_2(x=0,t) = \mathrm{O}_2^{\mathrm{L}}, \tag{8.82c}$$

$$O_2(x=L,t) = \mathrm{O}_2^{\mathrm{R}}, \tag{8.82d}$$

with $\mathrm{O}_2^{\mathrm{L}} = 0$ and $\mathrm{O}_2^{\mathrm{R}} = \mathrm{O}_2^{*}$ the oxygen at the left and right channel and $f_n = D_1 \frac{\partial C_n}{\partial X} - \chi F_{\text{go}}(C_n, C_d) \Pi_{\text{go}}(O_2) C_n \frac{\partial O_2}{\partial X}$ the alive cell flow.

In order to make the problem affordable from the analytical point of view, we neglect the dead cell population, which is equivalent to consider $\alpha_{12} \simeq 0$. Besides, we make the following assumptions about the nonlinear correction functions:

- **Saturation:** We assume $F_{\text{go}}(C_n, C_d) = 1$ and $F_{\text{gr}}(C_n, C_d) = 1 - \frac{C_n}{c_{\text{sat}}}$.
- **Oxygen consumption:** We consider the Michaelis-Menten consumption model so $\Pi_{\text{c}}(O_2) = \frac{O_2}{O_2 + k_{\text{m}}}$.

Therefore, the full model here analysed is:

$$\frac{\partial C_n}{\partial T} = \frac{\partial}{\partial X}\left(D_1 \frac{\partial C_n}{\partial X} - \chi \Pi_{\text{go}}(O_2) C_n \frac{\partial O_2}{\partial X}\right) + \alpha_1 \Pi_{\text{gr}}(O_2) C_n \left(1 - \frac{C_n}{c_{\text{sat}}}\right), \tag{8.83a}$$

$$\frac{\partial O_2}{\partial T} = \frac{\partial}{\partial X}\left(D_2 \frac{\partial O_2}{\partial X}\right) - \alpha_2 \left(\frac{O_2}{O_2 + k_{\text{m}}}\right) C_n. \tag{8.83b}$$

Note that the two functions related with the *go-or-grow* behaviour, Π_{go} and Π_{go}, may have been assumed as known, determined by standard parametric fitting, or using the data-driven (DD) non-parametric procedures described in Chapter 7.

8.3.1 Dimensionless model

In order to evaluate the relevance of the different phenomena, we define the dimensionless variables:

$$C_n = c_{\text{sat}} u, \tag{8.84a}$$
$$O_2 = \mathrm{O}_2^* v, \tag{8.84b}$$
$$X = Lx, \tag{8.84c}$$
$$T = \frac{t}{\alpha_1}, \tag{8.84d}$$

so Eqs. (8.83) become:

$$\frac{\partial u}{\partial t} = \frac{\partial}{\partial x}\left(\Pi_1 \frac{\partial u}{\partial x} - \Pi_2 \Pi_{\text{go}}(v) u \frac{\partial v}{\partial x}\right) + \Pi_{\text{gr}}(v) u\,(1-u)\,, \tag{8.85a}$$
$$\frac{\partial v}{\partial t} = \frac{\partial}{\partial x}\left(\Pi_3 \frac{\partial v}{\partial x}\right) - \Pi_4 \left(\frac{v}{v+\Pi_5}\right) u, \tag{8.85b}$$

where:

$$\Pi_1 = \frac{D_1}{\alpha_1 L^2}, \tag{8.86a}$$
$$\Pi_2 = \frac{\chi \mathrm{O}_2^*}{\alpha_1 L^2}, \tag{8.86b}$$
$$\Pi_3 = \frac{D_2}{\alpha_1 L^2}, \tag{8.86c}$$
$$\Pi_4 = \frac{\alpha_2 c_{\text{sat}}}{\alpha_1 \mathrm{O}_2^*}, \tag{8.86d}$$
$$\Pi_5 = \frac{k_{\text{m}}}{\mathrm{O}_2^*}. \tag{8.86e}$$

The new boundary conditions are:

$$\left.\frac{\partial f}{\partial x}\right|_{x=0} = 0, \tag{8.87a}$$
$$\left.\frac{\partial f}{\partial x}\right|_{x=L} = 0, \tag{8.87b}$$
$$v(x=0,t) = 0, \tag{8.87c}$$
$$v(x=1,t) = 1, \tag{8.87d}$$

where now $f = \Pi_1 \frac{\partial u}{\partial x} - \Pi_2 \Pi_{\text{go}}(v) u \frac{\partial v}{\partial x}$. In what follows, we will use u_t and u_x as a shortcut for $\frac{\partial u}{\partial t}$ and $\frac{\partial u}{\partial x}$.

8.3.2 Parameter reduction

Our starting point is the model described in Eqs. (8.85), which may be reformulated as:

$$u_t = \Pi_1 u_{xx} - \Pi_2 \left(\Pi_{\text{go}}(v) v_x u\right)_x + \Pi_{\text{gr}}(v) u\,(1-u)\,, \tag{8.88a}$$
$$v_t = \Pi_3 v_{xx} - \Pi_4 \frac{v}{v+\Pi_5} u. \tag{8.88b}$$

The parameters used for the GBM evolution are the ones presented in Chapter 6, except for the cell pedesis coefficient, which has been considered to be of the order $D_1 = 1 \times 10^{-11}$ instead of $D_1 = 5 \times 10^{-10}$ according to the values reported in the literature[26] (see the exhaustive analysis done in Chapter 6, Section 6.4.3.1). Also, we have conserved one single significant figure, due to the analysis that follows. For completeness, the parameters are listed in Table 8.1.

Parameter	Value
D_1	1×10^{-11} cm $\cdot$ s^{-1}
D_2	1×10^{-5} cm $\cdot$ s^{-1}
c_s	5×10^{7} cell $\cdot$ mL^{-1}
α_1	1×10^{-6} s^{-1}
α_2	2×10^{-9} mmHg $\cdot$ mL $\cdot$ cell^{-1} $\cdot$ s^{-1}
L	2×10^{-1} cm
$O_2{}^*$	7 mmHg
k_{m}	3 mmHg
χ	1×10^{-8} cm^2 $\cdot$ mmHg^{-1} $\cdot$ s^{-1}

Table 8.1: Physical parameters of the GBM model. The values are reported with one significant digit.

From the parameters in Table 8.1, we can obtain the following dimensionless parameters:

$$\Pi_1 \sim 2 \times 10^{-4}, \tag{8.89a}$$
$$\Pi_2 \sim 1, \tag{8.89b}$$
$$\Pi_3 \sim 2 \times 10^{2}, \tag{8.89c}$$
$$\Pi_4 \sim 10^{3}, \tag{8.89d}$$
$$\Pi_5 \sim 4 \times 10^{-1}. \tag{8.89e}$$

Therefore, we may consider $\Pi_1 \ll 1$ and $\Pi_3, \Pi_4 \gg 1$, and Eqs. (8.88) become:

$$u_t + k\left(\Pi_{\mathrm{go}}(v) v_x u\right)_x = \Pi_{\mathrm{gr}}(v) u\left(1-u\right), \tag{8.90a}$$
$$D v_{xx} = \frac{v}{v+v_0} u, \tag{8.90b}$$

where $k = \Pi_2$, $D = \frac{\Pi_3}{\Pi_4}$ and $v_0 = \Pi_5$.

The boundary conditions are the ones given by Eqs. (8.87), except for the fact that, now, as $\Pi_2 \ll 1$ we neglect the diffusion term and $f = \Pi_{\mathrm{go}}(v) v_x u$.

Eqs. (8.90) are strongly coupled and highly nonlinear, so they represent a hard problem to be analytically addressed. In what follows we shall use two different approaches:

[26] This assumption is justified from the scientific point of view, and will improve substantially the accuracy of the asymptotic analysis at the low diffusion regime.

1. **Strong assumption.** We assume that the cell concentration is small, that is, $u \sim 0$ so that Eq. (8.90b) may be approximated by $v_{xx} = 0$ and hence the oxygen profile is easily integrated to $v(x) = x$. This is a strong assumption but will lead to analytical solutions, valid for spaces and times where this approximation is reasonable.
2. **Slightly more general assumption.** We assume that the oxygen profile is known, for instance, if it can be measured experimentally or kept fixed as external data.

In both cases, observing that it is possible to write $k\Pi_{\text{go}}(v)v_x = \alpha(x,t)$ and $\Pi_{\text{gr}}(v) = \beta(x,t)$, the equation governing the cell profile evolution is:

$$u_t + (\alpha(x,t)u)_x = \beta(x,t)u(1-u), \tag{8.91}$$

with appropriate functions α and β.

The problem corresponding to the second approach has been solved for a general class of functions α and β. Next, we shall obtain an approximation of the solution for low cell concentrations.

8.3.3 Solution for low cell concentration

Let us first assume that $u \simeq 0$, so that we can approximate Eqs. (8.90b) by:

$$v_{xx} = 0. \tag{8.92}$$

As the boundary conditions are $v(x=0) = 0$ and $v(x=1) = 1$, the solution is $v(x) = x$, that is, a constant oxygen gradient. We assume that the two *go-or-grow* functions are the ones specified in Chapter 6, that is:

$$\Pi_{\text{go}}(O_2) = \begin{cases} 1 & \text{if} \quad O_2 \leq 0 \\ 1 - \frac{O_2}{O_2^H} & \text{if} \quad 0 \leq O_2 \leq O_2^H \\ 0 & \text{if} \quad O_2 > O_2^H \end{cases}, \tag{8.93a}$$

$$\Pi_{\text{gr}}(O_2) = \begin{cases} 0 & \text{if} \quad O_2 \leq 0 \\ \frac{O_2}{O_2^H} & \text{if} \quad 0 \leq O_2 \leq O_2^H \\ 1 & \text{if} \quad O_2 > O_2^H \end{cases}. \tag{8.93b}$$

Also, as at the whole domain $O_2 \leq {\text{O}_2}^{\text{H}} = 7$ mmHg, the two *go-or-grow* functions, in terms of the dimensionless variable v are:

$$\Pi_{\text{go}}(v) = 1 - v, \tag{8.94a}$$

$$\Pi_{\text{gr}}(v) = v. \tag{8.94b}$$

As we have seen, it is possible to express the (dimensionless) oxygen profile as $v(x) = x$, resulting in:

$$\alpha(x,t) = k\Pi_{\text{go}}(v)v_x = k(1-x), \tag{8.95a}$$

$$\beta(x,t) = \Pi_{\text{go}}(v) = x, \tag{8.95b}$$

while Eq. (8.91) may be writen as:

$$u_t + k(1-x)u_x = u\left(x - ux + 1\right), \tag{8.96}$$

that is, for the method of characteristics, $Q(t,x) = k(1-x)$ and $R(t,x,u) = u\left(x - ux + k\right)$. Also, the cell flow is given by:

$$f = -k(1-x)u. \tag{8.97}$$

8.3.3.1 Computation of the solution

Let us suppose a constant initial condition, $u_0(x) = u_0$. In this problem, we are interested in the characteristic equations given by Eqs. (8.8). The problem is the one corresponding to an oscillating gradient with $\alpha(x) = (ax + b)\cos(\omega t)$ with $a = -k$, $b = k$ and $\omega = 0$ and $\beta(x, t) = x$. Therefore, from Eqs. (8.21) we obtain:

$$\begin{aligned} F(t, s) &= 1 - (1 - s)e^{-kt}, \\ G(t, x) &= 1 - (1 - x)e^{kt}. \end{aligned} \tag{8.98}$$

The transition is located at:

$$x^*(t) = F(t, 0) = 1 - e^{-kt}. \tag{8.99}$$

The solution at the right domain has already been derived and is given by Eq. (8.41), where for the particular case in hands:

$$\begin{aligned} p(u, s) &= 1 - (1 - s)e^{-ku} + k, \\ q(v, s) &= 1 - (1 - s)e^{-kv}, \\ s &= 1 - (1 - x)e^{kt}. \end{aligned}$$

Note that $q(u, s) + k = p(u, s)$. The solution given by Eq. (8.41) may be expressed as:

$$v(t) = \exp\left(-\int_0^t p(u)\,du\right)\left[\frac{1}{u_0} + \int_0^t q(u)\exp\left(\int_0^u p(v)\,dv\right)\,du\right]. \tag{8.100}$$

We have:

$$\begin{aligned} &\int_0^t q(u)\exp\left(\int_0^u p(v)\,dv\right)\,du \\ &= \int_0^t (p(u) - k)\exp\left(\int_0^u p(v)\,dv\right)\,du, \\ &= \int_0^t p(u)\exp\left(\int_0^u p(v)\,dv\right)\,du - k\int_0^t \exp\left(\int_0^u p(v)\,dv\right)\,du, \\ &= \int_0^t \frac{d}{du}\left[\exp\left(\int_0^u p(v)\,dv\right)\right]\,du - k\int_0^t \exp\left(\int_0^u p(v)\,dv\right)\,du, \\ &= \left[\exp\left(\int_0^u p(v)\,dv\right)\right]_0^t - k\int_0^t \exp\left(\int_0^u p(v)\,dv\right)\,du, \\ &= \exp\left(\int_0^t p(u)\,du\right) - 1 - k\int_0^t \exp\left(\int_0^u p(v)\,dv\right)\,du. \end{aligned}$$

Defining $a(t, s) := \int_0^t p(u)\,du$ and $b(t, s) := \int_0^t \exp\left(\int_0^u p(v)\,dv\right)\,du$:

$$v(t, s) = \frac{1}{u_0}\exp(-a(t, s)) + 1 - \exp(-a(t, s)) - kb(t, s)\exp(-a(t, s)). \tag{8.101}$$

and, therefore:

$$u(t,s) = \frac{u_0 \exp(a(t,s))}{1 + u_0 \left[\exp(a(t,s)) - 1 - kb(t,s)\right]}. \tag{8.102}$$

Now, we compute the integrals of $a(t,s)$ and $b(t,s)$:

- **Computation of $a(t,s)$:**

$$\int_0^t p(u)\,du = (1+k)t + \frac{1}{k}(1-s)\left(e^{-kt} - 1\right). \tag{8.103}$$

- **Computation of $b(t,s)$:**

$$\int_0^t \exp\left(\int_0^u p(v)\,dv\right) du = \int_0^t \exp\left((1+k)u + \frac{1}{k}(1-s)\left(e^{-ku} - 1\right)\right) du \tag{8.104}$$

$$= e^{-\frac{1}{k}(1-s)} \int_0^t \exp\left((1+k)u + \frac{1}{k}(1-s)e^{-ku}\right) du \tag{8.105}$$

$$= e^{-\frac{1}{k}(1-s)} \int_1^{e^t} w^k \exp\left(\frac{1-s}{k} w^{-k}\right) dw. \tag{8.106}$$

By defining $A(s,t;k) = \int_1^{e^t} w^k \exp\left(\frac{1-s}{k} w^{-k}\right) dw$, we have;

$$b(t,s) = e^{-\frac{1}{k}(1-s)} A(s,t;k). \tag{8.107}$$

Now, we have $1 - s = (1-x)\exp(kt)$ so:

$$u(x,t) = \frac{u_0 \exp(a(x,t))}{1 + u_0 \left[\exp(a(x,t)) - 1 - kb(x,t)\right]}, \tag{8.108}$$

where:

$$a(x,t) = (1+k)t + \frac{1-x}{k}\left(1 - e^{kt}\right), \tag{8.109}$$

$$b(x,t) = \exp\left(-\frac{1}{k}(1-x)e^{kt}\right) B(t,x;k), \tag{8.110}$$

with $B(x,t;k) = \int_1^{e^t} w^k \exp\left(\frac{e^{kt}}{k}(1-x)w^{-k}\right) dw$.

Finally:

$$u(x,t) = \begin{cases} 0, & \text{if} \quad x < 1 - e^{-kt} \\ \frac{u_0 \exp(a(x,t))}{1 + u_0 [\exp(a(x,t)) - 1 - kb(x,t)]}, & \text{if} \quad x > 1 - e^{-kt} \end{cases} \tag{8.111}$$

8.3.3.2 Numerical comparisons

We now explore the agreement between the numerical solution of Eqs. (8.83) and the analytical expression given by Eq. (8.111). For reproducibility purposes, the exact value of the parameters is shown in Table 8.2. The results are shown in Fig. (8.14) for different values of the initial conditions, corresponding to $u_0 = 10^{-4}, 10^{-3}, 10^{-2}$ (recall that $u = C_n/c_{\text{sat}}$, so these values correspond to $0.01\%, 0.1\%$ and 1% of the cell saturation capacity of the device).

Parameter	Value
D_1	1×10^{-11} cm $\cdot$ s^{-1}
D_2	1×10^{-5} cm $\cdot$ s^{-1}
c_{sat}	5×10^{7} cell $\cdot$ mL^{-1}
α_1	1.43×10^{-6} s^{-1}
α_2	1×10^{-9} mmHg $\cdot$ mL $\cdot$ cell^{-1} $\cdot$ s^{-1}
L	2×10^{-1} cm
$\mathrm{O_2}^*$	7 mmHg
k_{m}	2.5 mmHg
χ	7.5×10^{-9} cm^2 $\cdot$ mmHg^{-1} $\cdot$ s^{-1}

Table 8.2: Physical parameters of the GBM model used for the numerical simulations. These are the values used in the whole model for obtaining the dimensionless cell profiles in Fig. 8.14.

As it is observed, there is a perfect match between the theoretical and the numerical solution for low cell concentrations, whereas the analytical estimation is not so accurate when increasing the value of the initial condition u_0, as expected.

The agreement between the analytical solution and the numerical one justifies the use of the analytical solution for some relevant biological estimations. For instance, it is possible to estimate the time for the cell propagating front to reach the oxygenated boundary. From Eq. (8.99), the time T for the cell front to reach $x = 1/2$ is:

$$t^* = \frac{\ln 2}{k}, \tag{8.112}$$

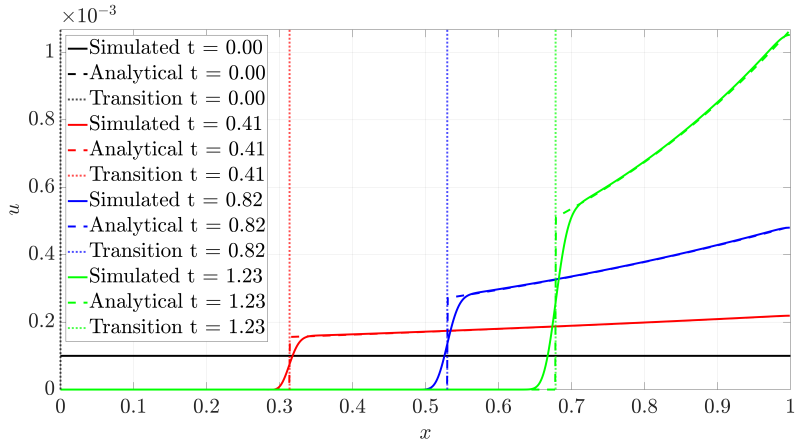
and, using dimensional variables, the time for the cell front to reach the middle $L/2$ of the chamber is:

$$T^* = \frac{L^2 \ln 2}{\chi \mathrm{O_2}^*}. \tag{8.113}$$

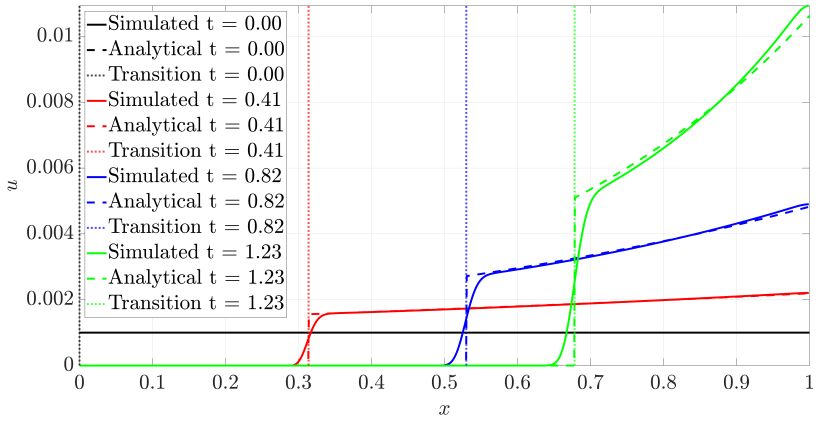
Using the values of Table 8.2, this corresponds to a value of:

$$T = 6\,\text{days}, \tag{8.114}$$

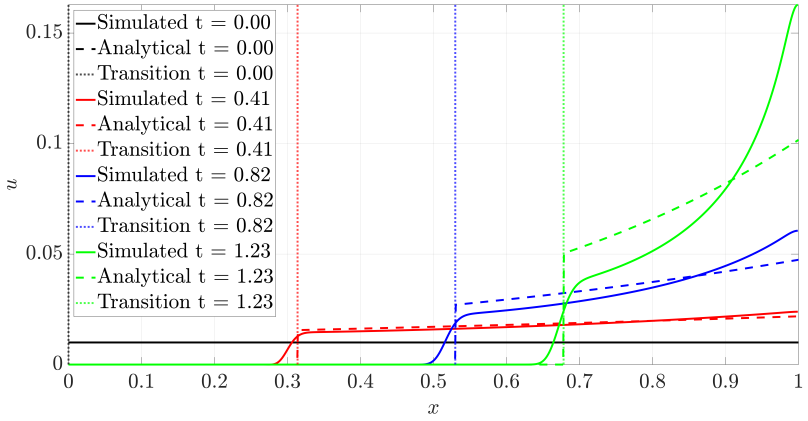
This last value is indeed of the order of the time reported in experiments to observe the migrating structures, as is described in Chapter 6, even if many simplifications have been considered. The expression given by Eq. (8.113) is very useful from the experimental point of view, as it allows the design of experiments, provided we are under the main hypothesis of this study, that is, $\mathrm{O_2}^* \leq \mathrm{O_2^H}$, $\alpha_{12} \simeq 0$ and the initial cell profile $C_n(x, t = 0) = C_n^0$ satisfies $C_n^0 \ll c_{\text{sat}}$.



(a) $u_0 = 1 \times 10^{-4}$.



(b) $u_0 = 1 \times 10^{-3}$.



(c) $u_0 = 1 \times 10^{-2}$.

Figure 8.14: Comparison of numerical and analytical solutions. The theoretical and simulated profiles at different times are compared for three different initial cell profiles. The agreement between the numerical and analytical solutions is excellent for low cell concentrations, whereas it gets worse when the number of initial cells seeded increases. The estimation of the transition between the two solution zones, that is, the front of the cell profile is also good.

8.4 Discussion and conclusions

The method of characteristics has been demonstrated to be a valuable tool for analysing one-dimensional cell cultures under heterogeneous growth and chemotaxis. We have analysed a class of problems that may be represented by the equation:

$$\frac{\partial u}{\partial t} = D\frac{\partial^2 u}{\partial x^2} - \frac{}{\partial x}\left(\alpha(x,t)u_x\right) + \beta(x,t)u(1-u), \tag{8.115}$$

under the hypothesis $D \ll 1$, $\alpha \sim \beta \sim 1$. We have shown that, this is the evolution equation of cell cultures in one-dimensional microfluidic devices, once formulated in a dimensionless form.

The dynamics of the system is characterised by means of the two functions α and β, associated with the migratory and proliferative activity of cells, respectively. The nature and property of these two functions have an impact on global features such as the speed of the propagating structures (or travelling waves), the value of the solution at the wave front or far from the wave front, the existence of periodic solutions or the existence of equilibrium points.

We have been able to derive semi-analytical solutions under the assumption of the α separability. This hypothesis, even if somewhat restrictive, is very common in practical situation, particularly in a broad range of problems related to the diffusion of a chemical species. If it is not the case, one may always evaluate the separability by doing a proper orthogonal decomposition (POD) of the function $\alpha(x,t)$, that is, to compute the singular value decomposition (SVD) of the matix $A_{ij} = \alpha(x_i, t_j)$, where x_i and t_j are spatial points and time frames and to take the dominant mode, as a first approximation. Finally, and in the worst case scenario, the system of ODEs given by Eqs. (8.9) and (8.9) may be solved numerically, avoiding the discretisation step and thus alleviating the computational requirements.

As a final remark, it is important to note that the semi-analytical solutions derived are not only important for making predictions about some culture features, such as the propagation of pseudopalisades, but also as a reference solution for detecting departures from the model or wrong assumption hypotheses. Indeed, in this work we have paid special attention to oscillatory solutions. The comparison of this solution trends with real experimental data can shed some light on some hidden phenomena. For instance, we exposed in Chapter 6 how a possible model improvement is to include the effect of the history on the current state of the cell, that is, epigenetics changes (Jaenisch and Bird, 2003), such as resistance or adaptative mechanisms. The comparison between the obtained reference solution under oscillating gradients, that is periodic for many situations, and the actual measurement cell concentrations, would help us to detect the existence of epigenetic changes. This would help us to propose new epigenetic mechanisms, or a least, to prove using statistical tools the existence of such hidden mechanisms.

Conclusions and outcomes

9. Conclusions

9.1 Concluding remarks

9.1.1 Summary of the dissertation

Surely, one of the possible answers that a researcher can give to the question "what is your goal?" would be to find a cure for cancer. It would be pretentious to say that progress has been made in this thesis for such purposes, but I have settled some very humble steps in that direction, to understand and fight cancer. I have focused on one particular cancer, glioblastoma (GBM), the most common and aggressive among primary brain tumours.

Throughout this dissertation, I have proposed new physically-informed data science (PIDS) methods and I have adapted some existing ones to more general frameworks. The first method proposed extended the work by Kirchdoerfer and Ortiz (2016a) to deal with noisy data, in what was called the reliability-based data-driven (RBDD) solver[27]. I derived some mathematical and statistical properties, performed some illustrative numerical tests and applied the method to concrete modelling for data coming from different tests.

Next, I extended the method to cope with incomplete data. Standard statistical imputation methods were not suitable for PIDS methods as they do not conserve the physical structure of the data. By establishing a novel unsupervised completion technique, I have transformed imputation weaknesses, such as their biased character and uncertainty reduction, into their strengths. Indeed, we do want the physics to reduce uncertainty and bias the data towards more physically sound data. I have demonstrated that this method improves standard Data Science predictions and is able to reduce systematic biases. Additionally, I have applied it to biomechanical problems, both as an instrumental procedure for running data-driven (DD) simulations and as a tool for accelerating multiscale computations.

[27] In parallel, Kirchdoerfer and Ortiz (2017) have proposed other solutions for dealing with noisy data-sets.

After these two contributions for DD problems, I changed the approach to the use of artificial neural networks (ANNs). Despite the huge amount of works in the last years merging partial differential equations (PDEs) and ANNs, as seen in Chapter 2, the singularity of the present work is that it is able to work with non-measurable variables, circumventing the need for data at the state space that may be considered only under very specific hypotheses, as explained in Chapter 3. This difficulty has been overcome only very recently, in very successful papers (Flaschel et al., 2021; Huang et al., 2020). I have demonstrated that our method has both predictive and explanatory capacity, and that it improves ANNs in aspects such as data needs, noise filtering and extrapolation capacity.

Next, I moved to the application of the proposed methods. First, I have built an initial parametric model for analysing and simulating cell cultures, and in particular GBM, in microfluidic devices. I have proposed a mathematical model composed of a system of PDEs modelling the evolution of different cell populations and species. Although there are other existing models in the literature, (Ayuso et al., 2017; Martínez-González et al., 2012), I have demonstrated to be able to predict the outcome of experiments under different experimental configurations for the first time. Despite its predictive power, I have demonstrated, using a novel stochastic approach for analysing parametric models, that the model presented some lacks in relation to its explanatory power, as it failed to perfectly characterise one of the main processes leading to the fast invasion of brain tissues: the *go-or-grow* switch.

As a response to this problem, I have used the new PIDS methods proposed in combination with the parametric model, but relaxing the rigidity of standard parametric fitting and replacing it with the flexibility of neural networks. I have been able to unravel, from a non-parametric perspective, the *go-or-grow* behaviour, reducing the errors both for the model explanations and for the model predictions, when compared to standard parametric fitting. Also, I can make predictions for different boundary conditions, something that other methods, such as biologically-informed neural networkss (BINNs) (Lagergren et al., 2020), cannot afford. I suggest that this approach lays the first stones towards personalised medicine.

Finally, I closed the dissertation with a mathematical analysis of cell cultures under heterogeneous migration and proliferation mechanisms. From the computational point of view, the derivation of global features such as migrating structures or trends of the solution is much less expensive and may offer insights about GBM evolution without the need for detailed field computations. I have demonstrated to be able to predict accurately the formation of pseudopalisades, and to accurately estimate their speed. Also, I have derived many analytical solutions for interesting cases, that may be used as a ground solution for novelty detection, as might be the one derived from epigenetic considerations.

9.1.2 Specific remarks

Throughout this dissertation, I have used different approaches to analyse cell cultures in microfluidic devices, ranging from conventional parametric models, to modern approaches under the scope of PIDS methods. The degree of knowledge about the cell culture evolution process and the data availability has been exploited in different levels, as illustrated in Fig. 9.1:

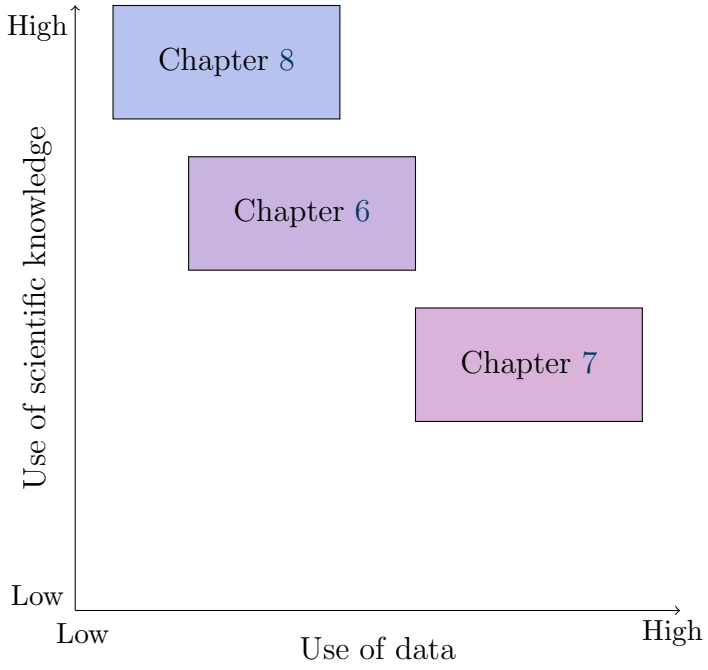


Figure 9.1: Situation of the work under the prism of Physically-Informed Data Science methods. The approaches proposed in each chapter use in a different manner the data availability and the existing knowledge.

1. In Chapter 6, I used this knowledge to define a set of PDEs modelling GBM evolution, reducing the ignorance about the model to the value of some model parameters.
2. In Chapter 7, I relaxed this knowledge by allowing a non-parametric dependence for the *go-or-grow* switch, at the cost of needing more data for characterising accurately this transition.
3. In Chapter 8, the model is assumed as perfectly defined and is reduced to the value of some dimensionless parameters, that are assumed as known, so no extra data is needed in order to make predictions.

Chapters 3 and 4 may be viewed as more instrumental ones. The method and techniques proposed there have not been applied to the GBM problem. However, this is because when facing the GBM parametric model limitations, I detected that a crucial point would be to work under a framework able to incorporate non-measurable variables, as are the two nonlinear corrections related to the *go-or-grow* behaviour, Π_{go} and Π_{gr}, as explained in the discussion of Chapter 6. The only way to incorporate this information using a PIDS was to slightly change the approach. The use of ANN methods and machinery was of course another argument for justifying this decision.

9.1.3 General remarks and outlook

One of the main objectives of this thesis was to develop new PIDS strategies without losing contact with real bioengineering problems. This is the reason for, whenever possible, using the methods and strategies proposed in real bioengineering problems

and with real experimental data. Therefore, and despite the strong methodological character of this dissertation, we have applied the method in at least three problems using real data:

1. The RBDD solver used for concrete modelling was used in combination with real measurements from uniaxial tests.
2. A DD solver has been used in combination with the filling data procedures described for the analysis of cortical bone mechanics and structure in multiscale computations. The data used for this work were obtained from real digital image correlation (DIC) tests.
3. The parametric model for GBM evolution in microfluidic devices was fitted using real data obtained from microfluidic devices, from previous and ongoing experiments in the TMELAB group[28].

Actually, the broad literature about PIDS is mainly focused on the methods, and usually uses *in silico* data for validating the approaches, as it is also frequently done thoughout this dissertation. There are, indeed, very few works that incorporate real data to computations in the PIDS workflow, using data acquisition techniques (see, for instance, Badías et al. (2020); Moya et al. (2020a)). Here, this has been possible thanks to the use of microfluidic platforms[29] in combination with specific biomarkers, thus opening the door to the use of machine learning (ML) techniques in cell culture monitoring (Galan et al., 2020). There is no other work, at least up to my knowledge, that combines these three fundamental pillars: ML and artificial intelligence (AI) tools, real bioengineering or biomedical data obtained from a recent technology and the incorporation of the physics in a consistent manner.

To be fair, I must specify that the final goal to be achieved is the use of the methods presented in Chapter 5 and particularised to the GBM evolution problem in Chapter 7 to real data obtained from cell cultures in microfludic devices. However, this requires experimental control and automation image adcquisition techniques that are not yet available (but will be soon, hopefully) in our lab capacities. We should highlight two main limitations:

- The capacity of measuring quantitatively and accurately the oxygen concentration at the devices, extensively, and not only in specific locations.
- The capacity of capturing and processing the cell evolution (or, rather, the biomarkers signals) in real time and without affecting the culture conditions. Indeed, some biomarkers used for fluorescence signalling change the cell state and can be used only at the end of the experiment[30].

In any case, the DD methods developed, particularly the physically-consistent imputation method for incomplete data, would help us in this direction.

[28] Link to the group page: https://tmelab.unizar.es/

[29] During the exposition, I have also used real data coming from DIC tests, but I put them aside for a moment as they are not the core of this dissertation.

[30] As a curiosity, this is a paradigmatic case where the measurement/observation of a certain variable affects the experiment, without the need of invoking quantum mechanics and go to atomic length scales.

9.2 Thesis contributions

The fundamental contribution of this thesis is the development of PIDS specific tools for the analysis of the GBM evolution in microfluidic devices. This is the first work that has addressed this goal with at least the following ingredients, in a single framework:

- The approach combines the physical knowledge about a system with an intensive use of experimental data, obtained from singular platforms as the microfluidic devices. The methods here presented are therefore adapted for a new paradigm in biological sciences and bioengineering, which had not been previously exploited due to the lack of such data generation techniques.
- The focus on the *go-or-grow* switch in hypoxic cells has required a unique approach, making use of the concept of internal variables. Before this thesis, there were no methods able to deal with internal non-measurable variables in a robust and flexible way for analysing PDEs, such as the ones modelling GBM evolution.

Next, we describe other technical contributions related in a way or another to the former, belonging to the two main directions of the thesis: the methodological one, related to the design of new PIDS tools, and the applied one, related to GBM evolution models. This enumeration does not coincide strictly with the two parts of the thesis exposition.

9.2.1 Methodological contributions

From the methodological point of view, the following contributions may be enumerated:

1. I have formulated PIDS in a general and abstract framework, where each of the discussed methods has its own representation.
2. I have adapted some existing PIDS methods to "real world" problems with imperfect data:
    - I have extended the DD solver to the RBDD solver, which is able to incorporate the data uncertainty and to run appropriately in contexts with noisy data.
    - I have defined a new PIDS statistical imputation method. It allows to fill gaps in data-sets in a physically-consistent manner and may be combined with DD schemes.
3. I have established a new family of PIDS methods, the so-called physically-guided neural networks with internal variabless (PGNNIVs), that are able to incorporate, with flexibility, the problem physics (universal laws, constitutive equations or both) under the framework of field theories, which are formulated using internal non-measurable state variables.

4. I have built a computational framework for simulating cell cultures in microfluidic devices. This framework includes:
   - A mathematical model for modelling cell growth, death and differentiation, cell random pedesis, chemotaxis, mechanotaxis, thermotaxis and electrotaxis and tissue remodelling, as well as the associated nonlinearities.
   - A 3D computer software, based on finite element method (FEM), able to simulate all the aforementioned phenomena.
   - A 1D software, based on an already existing Matlab package, for simulating GBM progression in microfluidic devices under the one-dimensional hypothesis in competitive times.
5. I have proposed an original stochastic framework based on *copulae* to study non-explicit parametric models (such as the ones that are the result of numerical simulation), able to incorporate parameter uncertainty and structural dependence. This method is also able to detect model limitations in relation with the available data and the design of new experiments or experimental campaigns.
6. I have exploited the method of characteristics as a suitable method for analysing cell cultures in microfluidics under the one-dimensional assumption, and to be able to analyse some features such as the formation of migrating structures (pseudopalisades).

9.2.2 Applications

From the point of view of results and applications, we may enumerate the following contributions:

1. I have applied the RBDD solver to different problems, some of them with real world data, such as concrete modelling and bone biomechanics.
2. I have demonstrated that the filling data procedure may be combined with the DIC method for accelerating the multiscale computations, in the context of tissue engineering.
3. I have demonstrated that PGNNIVs improve standard ANN methods with respect to data needs, noise filtering, convergence rate and extrapolation capacity. What is more, they are the only ones that have explanatory capacity (ability to unravel constitutive equations), in addition to predictive capacity (ability to predict the state of the system at a point that is not part of the data-set).
4. I have defined a parametric model able to recreate the evolution of GBM under hypoxic conditions in microfluidic devices. This model is a parametric system of PDEs, that includes 11 model parameters. This model has demonstrated to be able to reproduce the experimental results of different experimental configurations, such as the single and double pseudopalisade (one or two oxygenated channels respectively) formation and the necrotic core formation (high cell concentration).
5. I have detected the GBM evolution model limitations, particularly with regard to the *go-or-grow* metabolic switch. This has been possible thanks to the presented stochastic approach.

6. I have applied PGNNIV to the GBM evolution problem, being able to capture different *go-or-grow* switch state model candidates, in a non-parametric way and outperforming standard parametric fitting both in predictive and explanatory performance, and for different boundary conditions.
7. I have estimated accurately the formation of pseudopalisades and the time for them to reach a distant oxygenated structure, the lateral chamber, that plays the role of an oxygenated vessel.

9.3 Scientific and technological outcomes

Here, I summarise the scientific and technological outcomes of this thesis. A detailed list of publications, works published and congress participations may be found later.

9.3.1 Software developed

During the thesis, I have developed many scientific codes and software packages.

1. A FEM based software for simulating cell cultures in microfluidic devices, implemented from scratch in Matlab and accounting for nonlinearities, coupling between the different field variables and tissue remodelling. This software has been validated for a broad class of problems.
2. Some software packages and routines, in Matlab, related to the RBDD solver and the missing data strategy. Some of them are used for solving the one-dimensional problem exposed in Chapter 3 and the 3D problem of concrete under the assumptions established. Also, a code for solving the filling data procedure in the linear case was developed. Finally, a code for the data completion technique for the biomechanical macroscopic problem and a routine for the matching scales procedures at the multiscale approach.
3. Some software packages and routines for the analysis of parametric models using *copulae*, including model analysis, probabilistic predictions and experimental utility computations. These codes have been developed in collaboration with another Ph.D. student (Marina Pérez Aliacar).
4. Some software packages and routines, in TensorFlow@Python, for solving the different PGNNIV problems exposed. These include the hydraulic prediction and characterisation problem, the diffusion equation (heterogeneous and nonlinear), the GBM evolution problem, and some codes related to solid mechanics that are at the core of future published work.

9.3.2 Validated models

As a by-product of this thesis, we can declare a parametric model for GBM evolution in microfluidic devices, consisting of a system of PDEs with 11 model parameters. This model has been fitted to experimental data and validated using different experimental configurations.

9.3.3 Summary of the published work

During the thesis, I have published in scientific journals and with scientific publishers.

1. Five (5) journal articles as the first author in first quartile journals.
2. Two (2) journal articles as participating author, in first and second quartile journals, that are applications of the methodology developed in this Thesis and in which I had a leading role.
3. One (1) book chapter, as the first author, from Elsevier publisher.

In addition, I have the following papers in progress:

1. One (1) journal article as the first author under review in a first quartile journal *PLoS Computational Biology*.
2. One (1) journal article as the first author available in a public repository (*arXiv*) pending to be submitted.
3. One (1) journal article in preparation for *Bulletin of Mathematical Biology*.
4. One (1) journal article planned on the application of PGNNIV to solid mechanics.

Finally, I have participated in many national and international conferences (as the first author and speaker):

1. One (1) co-chair of a special session in an international conference.
2. Nine (9) oral communications in international conferences.
3. Two (2) poster communications in international conferences.
4. One (1) chair of a parallel session in national conferences.
5. Seven (7) oral communications in local or national events.
6. Four (4) poster communications in local or national events.

9.4 Thesis conclusions

In summary, the main conclusions of this dissertation are the following:

Thesis main conclusions

- It is possible to adapt the methods of data-driven simulation-based engineering and sciences (DDSBES) to contexts with data that are noisy or incomplete.
- For physical problems with internal and non-measurable variables, PGNNIVs have demonstrated to be a PIDS method with both predictive and explanatory capacity.
- GBM evolution in microfluidic devices can be modelled by means of a parametric model. This model is able to recreate culture evolution under different experimental configurations, but fails to completely explain the *go-or-grow* switch.
- The use of PGNNIV in combination with the GBM evolution problem is able to alleviate the parametric model lacks and to explain general *go-or-grow* metabolic switches.

Other conclusions of each chapter are:

- **Chapter 3:**
  - DD solvers are more robust than regression or interpolation techniques in nonlinear contexts.
  - The RBDD solver is robust in contexts with data uncertainty, allows for uncertainty propagation and may be applied to data from different experiments.

  Chapter 4:
  - A novel unsupervised learning technique presented for data completion improves some statistical imputation techniques and may correct non-physical biases of the data.
  - This technique can be used in multiscale contexts to accelerate the computations without losing the access to microstructural data.

  Chapter 5:
  - PGNNIV outperform classical ANNs in aspects such as convergence speed, data needs, noise filtering capacity and extrapolation capacity.
  - PGNNIV may incorporate the most common features of linear and nonlinear operators in Continuum Physics.

  Chapter 6:
  - A general computational framework is possible for modelling cell cultures in microfluidic devices.

– Common biological and biomedical parametric models are complex and need a nonlinear statistical approach for analysing the spread and correlation of the model parameters.

Chapter 7:

– PGNNIVs applied to the GBM evolution problem outperform standard parametric fitting in both predictive and explanatory capacity.
– As PGNNIV work correctly for variable boundary conditions, the presented approach let us glimpse the first steps towards personalised medicine.

Chapter 8:

– It is possible to recreate GBM histological features such as migrating pseudopalisades and vessel occlusion without the need for numerical simulations.

9.5 Future work

Paraphrasing the engineer Fred Brooks[31]: "Dissertations are not finished, they are abandoned", so there is still room for improvement and future investigations. In order to clarify the exposition, I include the different future work lines associated with the different chapters of this dissertation, and after that, I present some more general aspects.

9.5.1 Future lines corresponding to specific tasks

Chapter 3

In Chapter 3, I proposed a new RBDD solver able to incorporate the uncertainty of the data into the computations. There are many possible future directions:

- One possible work was to extend the RBDD methodology to more complex situations, such as dynamics, inelasticity, fracture mechanics... However, and simultaneously to the development of this thesis, M. Ortiz and co-workers have covered each and every one of these aspects (see Chapter 2 for the details).
- Another future line is to develop sampling strategies based on the RBDD error estimation. As the RBDD solver provides an error estimate (the final distance between the physical manifold and the closest point belonging to the data-set), it is possible to use it for evaluating data coverage and therefore sample new data points, that will, in turn, feed the solver and improve its performance.
- As the RBDD solver (like the DD one) strongly depends on the data sampled at the state space, it is very important to lay out complete and high-quality databases. This is not an easy task since, as illustrated for the concrete application, the sample coverage increases strongly with the material complexity, and if no reliable hypotheses are assumed (isotropy, symmetries, homogeneity), the problem becomes intractable from the DD point of view. Building lower dimension manifolds from the sample data could be an interesting solution to be explored.

[31] Frederick P. Brook is Professor Emeritus at the University of North Carolina of Chapel Hill, https://cs.unc.edu/person/frederick-p-brooks-jr/

Chapter 4

In Chapter 4, I proposed a new statistical imputation aware of the problem physics. This method is sensitive to the data structure and external physical constraints. Some possible future lines are:

- I have presented a filling data procedure that solves the problem in one stroke for data with linear structure and I have presented a heuristic algorithm for nonlinear structures, based on the combination of linear projections and the use of the exponential map for returning to the nonlinear manifolds. This algorithm is expensive and its performance is context dependent, as it relies on some desirable properties such as smoothness, regularity and convexity of the functions and manifolds involved. There is an obvious need for establishing sound mathematical results for sufficient convergence conditions, error estimates and convergence rate under specific conditions.
- The proposed filling data procedure assumes that all the physical equations of the system are known. For instance, when applied to multiscale problems, it assumes a certain stress-strain relation at the microscale, as only strains are measurable at this level (using DIC techniques). Stresses at the microscopic level have to be derived from a given constitutive equation, if desired. A future work direction would be to relax this assumption and to adapt the DD procedures to this demanding situation.

Chapter 5

In Chapter 5, I defined the concept of PGNNIV, which has demonstrated to be very useful as a PIDS method able to work with internal non-measurable variables. As it has been seen on the state of the art review in 2, combining ANNs with physical information is a glowing hot spot in research nowadays (some examples are physics-informed neural networks (PINNs), structure preserving neural networks (SPNNs), thermodynamic-aware ANNs, among many others). Some possible directions of future work are:

- The adaptation of the methodology to continuum problems has demonstrated to be both effective and challenging. The discretisation of the problem has to be carefully performed, as it has been demonstrated for the diffusion equation, where increasing the number of nodes does not improve the accuracy of the predictions. In that sense, there is here also a need for theoretical mathematical works exploring convergence sufficient or necessary conditions. In a way, the results about ANNs as universal approximators have to be reframed under the scope of PDEs. The theoretical results obtained for PINNs can be adapted for establishing analogous results for PGNNIVs.
- Another challenging problem of continuum PGNNIV is the curse of dimensionality. As they are formulated, PGNNIV have as many output variables as degrees of freedom has the problem. For instance, for the prediction of a displacement field in a 3D problem with a mesh of $100 \times 100 \times 100$ nodes, we need a network having $N = 3 \times 10^6$ outputs, what is unapproachable from the computational

point of view. Fortunately, we are more interested in capturing global features than independent single output values, so this can be bypassed by the definition of appropriate quantity of interests (QoIs). Another approach to this problem, is to use spectral techniques. Defining the outputs of the PGNNIV as the spectral coefficients of the solution would turn the problem in a more manageable one and would circumvent the curse of dimensionality.

- Finally, variable or irregular domain geometries are also a challenging problem, due to the intrinsic structure nature of PGNNIV. A variable geometry problem has been solved for the hydraulic problem in Chapter 5, but the task is harder when the geometry is associated with a mesh, and no simple parametrisation is possible. Besides, standard interpolations fail to predict accurately the value of a given field when no Euclidean structure is present on the data (as it is the case for structured meshes represented by pixels and voxels). Indeed, the interpolation has to be performed at the latent reduced space and next the results have to be projected into the original space. As it has been presented in the state of the art of Chapter 2, there is considerable progress in that direction, in what is called geometric deep learning (DL) with the appearance in particular of graph neural networks.

Chapter 6

In Chapter 6, I have defined a computational framework for analysing cell culture evolution in microfluidic devices and I have particularised it to GBM. Also, I have fitted a parametric model from experimental data, and using a stochastic approach, I have performed sensitivity and structural analyses. In addition to the intrinsic limitations of the model, that have been surpassed in Chapter 7, I present here some possible future lines:

- From the modelling point of view, I have detected some limitations to reproduce experimental results. First, I have detected a mismatch between predicted and experimental results close to the lateral channels and an overestimation of alive cell population at the centre of the chamber for high cell concentration experiments. These errors could be explored by including the oxygen flow obstruction effect that may be the result of high density cultures. This may be done by including a nonlinear correction at the flux term associated with the oxygen evolution. However, for making these extra modelling tasks, it would surely be necessary to include raw measurements of the oxygen level, something that is not so straightforward from the experimental point of view. Another model improvement would be to include the effect of the history on the cell metabolic behaviour. The study of the (reversible or not) changes suffered by a cell due to the action of the external stimuli, changing the cell gene expression (phenotype) and not affecting a change in the genetic content of the ADN is known as epigenetics (Jaenisch and Bird, 2003) and is a hot spot of current research. To include epigenetics in the GBM model will correct some of the discrepancies found in short term measurements and could explain complex mechanisms such as cell conditioning to hypoxia, cell adaptation and cell resistance to therapies.

- From the computational point of view, there is an intrinsic limitation of the presented framework due to a multiscale physical problem. The characteristic time of oxygen diffusion is much lower than the characteristic time of cell pedesis. Indeed, the ratio between the two time scales is $\rho = \frac{\tau_{O_2}}{\tau_{\text{cell}}} = \frac{D_n}{D_{O_2}} = 5 \times 10^{-5}$. The computer software, based on a FEM implementation needs a lot of iterations to solve the overall problem, especially for the first time steps where the oxygen level has not reached a stationary profile. This monolithic approach is expensive and makes the problem unaffordable due to the computational cost. Possible solutions are to adopt a staggered approach, where the oxygen profile is assumed as stationary or to build a response surface relating the low time scale and the high time scale.
- Another problem related to computational cost is the Bayesian approach of design of experiments (DoE). As the measurement space in microfluidic devices is large (the cell concentration at each measurement point) the utility computations are very expensive as they require the computation of multiple integrals in high dimensional spaces. The use of suitable integration methods as Markov chain Montecarlo method (MCMCM) would alleviate these computational requirements.

Chapter 7

In Chapter 7, I have applied the concept of PGNNIV to the problem of GBM evolution, in particular, to the study of the *go-or-grow* behaviour.

- I have proposed a method for unravelling in a non-parametric fashion the *go-or-grow* switch behaviour. As the method is intrinsically non-parametric, we need to define new tools for uncertainty incorporation. Probabilistic Bayesian neural networks (BNNs) are certainly a tool that should be explored in order to establish a way for incorporating this uncertainty to the fitted network parameters and to propagate it to the state equation unveiling. Another possible approach is to use a statistical second order characterisation of the output layer (for instance using Mahalanobis distance) and to propagate quadratic statistics to the internal layers of the network. This work is indeed in process.
- Of course, this method has to be used with real experimental data, rather than *in silico* generated data. The approach followed was justified by the fact that our aim was to evaluate whether the method is able to unravel the *go-or-grow* switch for many different functional dependencies. But it is for unravelling the real cell culture behaviour when this method will really add value to scientific research. Once the non-parametric dependence is learned, a cross-over between the derived model and many available candidates associated with different metabolic pathways is possible, thus unveiling the real mechanistic and molecular phenomena driving this behaviour change, and shedding light on the biological processes involved in GBM evolution.
- As suggested earlier, the GBM evolution model may be enriched by taking into account epigenetics and, therefore, the history dependence of cells. This, under the scope of PGNNIV, may be faced by using recurrent neural network (RNN),

that were precisely designed for dealing with historical data and time series analysis.

Chapter 8

In Chapter 8, I have applied the method of characteristics for deriving some global features about the solution. Although important features, such as travelling waves, having their counterpart in histological features (pseudopalisades), were derived, there is still room for improvement.

- I have derived the outer region solution using the method of characteristics, but the solution at the transition region (inner solution) and at the boundaries (boundary layer) may also be explored using conventional mathematical techniques (asymptotic theory). However, this is not an easy task as the strong nonlinearities of the problem complicate the use of mathematical tools for deriving analytical solutions.
- The approach followed in this chapter was focused on one-dimensional problems. An evident future work possibility is to extend the results to 2D and 3D ones (such as the one of the GBM progression in the brain tissue). This direction will tremendously enrich the analysis, as some features such as domain heterogeneity and its effect on the shape of the travelling front surfaces might be explored.

9.5.2 Future lines corresponding to general tasks

In addition to the specific future lines associated with the different works at each chapter, other more general future lines are:

- To use the built GBM evolution model to explore therapeutic treatment strategies and drug design. For instance, we can explore how the action over oxygenated tissues and oxygenation structures (such as blood vessels), or the use of drugs promoting localised hypoxia, may affect GBM evolution. This is a very ambitious goal but there is no doubt that this is one of the final goals of biomedical engineering.
- To incorporate to the GBM evolution model the effect of the substrate mechanics. Even if this has been considered in the general framework established in Chapter 6, there are two important things that have to be pointed out. First, I have considered a two-way mechanical coupling but with simple assumptions: some mechanical parameters influence the cell progression (by affecting the different coefficients or by defining mechanical gradients, such as stiffness gradients or pressure gradients), which in turn may modify the value of these parameters, that are indeed treated as spatial fields (mechanical remodelling). However, this mechanical remodelling is set *a priori* using *ad-hoc* remodelling relations, for instance, how the cell concentrations affect the tissue stresses. A more appropriate analysis of the problem would be to consider also the mechanical problem and to solve it completely. Indeed, what cells really do is produce substances or proteins, such as collagen fibers, affecting the extracellular matrix

(ECM) structure and therefore changing its stiffness. This stiffness change would be transferred (at the same strain level) to a change of the stress state, which would be obtained by completely solving the mechanical problem. This is also applicable to more complex phenomena involving viscous effects, or using poroelastic or poroviscoelastic approaches.

- The use of the PGNNIV framework has been applied here to illustrative, although conceptually simple problems. One future line is to apply this framework to more complex problems, especially from the area of computational mechanics (CM), and in particular to the elastic and inelastic problem in finite strains, where constitutive modelling has also been a bottleneck due to the experimental limitations. Although the last decades have been very fruitful in constitutive modelling of inelastic solids and soft tissues, this modelling step may be by-passed thanks to the use of PGNNIV, avoiding an *a priori* selection.

10. Conclusiones

10.1 Observaciones finales

10.1.1 Resumen de la disertación

Al preguntar a un investigador cuál es su objetivo, seguramente una de las respuestas posibles sea "encontrar una cura para el cáncer". Sería pretencioso decir que en esta tesis se ha avanzado hacia ese fin, pero sí he dado algunos humildes pasos en la dirección de entender mejor el cáncer y luchar contra él. En concreto, el trabajo se centra en un tipo de cáncer, el glioblastoma (GBM), el más común y letal de entre los tumores primarios que afectan al sistema nervioso central.

A lo largo de la disertación, he propuesto nuevos métodos de Ciencia de Datos Informada por la Física (PIDS por sus siglas en inglés) y he adaptado algunos métodos existentes a marcos más generales. El primer método propuesto parte del trabajo de Kirchdoerfer and Ortiz (2016a) y lo amplía para trabajar con datos con ruido, en lo que se ha llamado *solver* RBDD[32]. He deducido algunas propiedades matemáticas y estadísticas, realizado simulaciones numéricas ilustrativas y, finalmente, el método ha sido aplicado al modelado del hormigón, a partir de datos experimentales provenientes de distintos ensayos.

Seguidamente, extendí el método para poder tratar con datos incompletos. Los métodos estándar de imputación estadística no son apropiados para los métodos de PIDS, ya que no conservan la estructura física de los datos. Al establecer una nueva técnica de completado no supervisada, las debilidades de la imputación estadística, como su carácter sesgado y la reducción de la incertidumbre, se transforman en fortalezas. Así, queremos que la física reduzca la incertidumbre y el sesgo de los datos, dando lugar a datos físicamente más robustos. He demostrado que este método mejora las predicciones estándar de la Ciencia de Datos y que es capaz de reducir sesgos sistemáticos. Además, lo he aplicado a problemas biomecánicos, tanto de forma

[32]En paralelo, Kirchdoerfer and Ortiz (2017) han propuesto otras soluciones para tratar con conjuntos de datos que tienen ruido.

instrumental para realizar simulaciones guiadas por datos como como herramienta para acelerar cálculos en simulaciones multiescala.

Tras estas dos contribuciones, enfocadas a problemas guiados por datos (DD por sus siglas en inglés), reenfoqué el trabajo hacia las Redes Neuronales Artificiales (ANN por sus siglas en inglés). Se ha visto en el Capítulo 2 que, en los ultimos años, se han desarrollado multitud de trabajos fusionando ANNs y Ecuaciones en Derivadas Parciales (PDE por sus siglas en inglés). Frente a ellos, la singularidad de este trabajo es que es capaz de trabajar con variables no medibles, eliminando así la necesidad de datos en el espacio de estados que solo pueden ser considerados bajo ciertas hipótesis muy concretas, como se ha explicado en el Capítulo 2. Esta dificultad ha sido superada muy recientemente, en artículos de mucho éxito (Flaschel et al., 2021; Huang et al., 2020). Hemos demostrado que nuestro método tiene capacidad tanto predictiva como explicativa, y que mejora las ANNs en aspectos como la cantidad de datos que requieren, el filtrado de ruido y la capacidad de extrapolación.

Seguidamente, pasé a la aplicación de los métodos propuestos. En primer lugar, he definido un modelo paramétrico para analizar y simular cultivos celulares, en particular de GBM, en dispositivos microfluídicos. He propuesto un modelo matemático compuesto por un sistema de PDEs capaz de modelar la evolución de distintas poblaciones celulares y especies químicas. Aunque hay otros modelos existentes en la literatura (Ayuso et al., 2017; Martínez-González et al., 2012), el modelo aquí propuesto es capaz de predecir, por primera vez, el resultado de experimentos en distintas condiciones experimentales. A pesar de este poder predictivo, he demostrado, utilizando un nuevo marco estadístico para analizar modelos paramétricos, que el modelo tiene algunas carencias en cuanto a su poder explicativo, ya que no es capaz de caracterizar del todo uno de los principales procesos que resulta en la rápida invasión de los tejidos cerebrales por parte del GBM: el *go-or-grow*[33].

En respuesta a este problema, he utilizado el nuevo método de PIDS propuesto (lo que he llamado Redes Neuronales Guiadas por la Física con Variables Internas o PGNNIV por sus siglas en inglés) con el modelo paramétrico, relajando la rigidez del ajuste paramétrico y reemplazándola por la flexibilidad de las redes neuronales. He sido capaz de desentrañar, desde un punto de vista no paramétrico, el comportamiento *go-or-grow*, reduciendo los errores obtenidos para la predicción y explicación del modelo respecto de los ajustes paramétricos tradicionales. Además, se pueden hacer predicciones para distintas condiciones de contorno, algo que otros métodos como las Redes Neuronales Guíadas por la Biología (BINNs por sus siglás en inglés) no son capaces de conseguir (Lagergren et al., 2020). Así, sugiero que el enfoque propuesto da los primeros pasos hacia la medicina personalizada.

Finalmente, cierro la disertación con un análisis matemático de cultivos celulares sujetos a mecanismos de migración y proliferación heterogéneos. Desde el punto de vista computacional, la derivación de características globales, como estructuras migratorias, o de tendencias en la solución, es mucho menos costosa y puede ofrecer

[33]Este paradigma se caracteriza por considerar dos comportamientos diferenciados para las células de GBM. Por un lado, las células muestran un comportamiento fundamentalmente migratorio cuando se encuentran privadas de oxígeno (*go*), mientras que cuando tienen condiciones de oxígeno suficiente su comportamiento es proliferativo (*grow*).

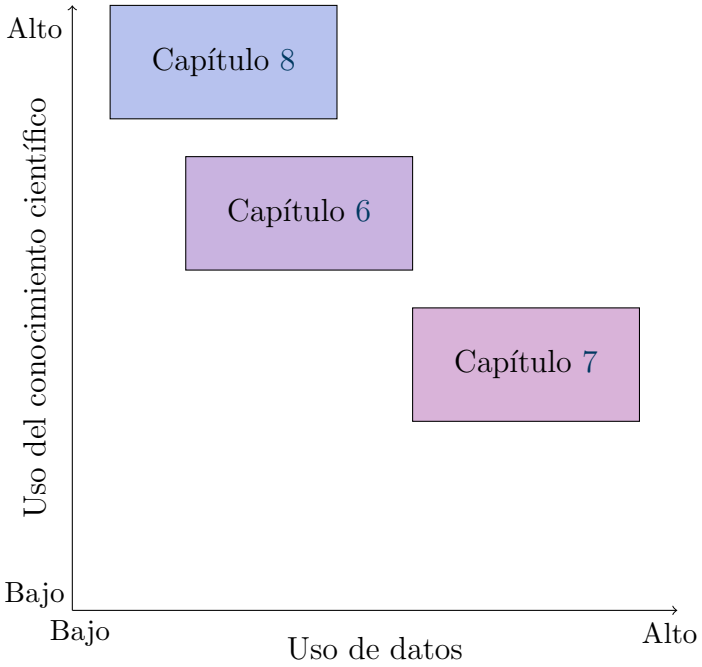


Figure 10.1: Situación del trabajo bajo el prisma de la Ciencia de Datos Guiada por la Física. Las aproximaciones propuestas en cada capítulo utilizan de una forma distinta los datos disponibles y el conocimiento existente.

información sobre la evolución del GBM sin la necesidad de calcular los campos completos. Asi, se ha predicho la formación de pseudoempalizadas, así como de estimar su velocidad. Además, he derivado varias soluciones analíticas para casos interesantes, que pueden ser usadas como soluciones de base para detectar novedades, como por ejemplo la consideración de la epigenética.

10.1.2 Observaciones específicas

A lo largo de esta disertación he usado diferentes aproximaciones para analizar cultivos celulares en dispositivos microfluídicos, que van desde modelos paramétricos hasta enfoques más modernos al amparo de los métodos PIDS. El grado de conocimiento sobre el proceso de evolución del cultivo celular, así como la disponibilidad de datos, se han explotado en diferentes niveles, como se ilustra en la Figura 10.1:

1. En el Capítulo 6, se ha usado este conocimiento para definir un sistema de PDEs que define la evolución del GBM, reduciendo el desconocimiento del modelo al valor de sus parámetros.
2. En el Capítulo 7, se ha relajado el grado de conocimiento permitiendo que las dependencias que conforman el *go-or-grow* sean no paramétricas, a costa de necesitar más datos para caracterizar con precisión estas transiciones.
3. En el Capítulo 8, se supone que el modelo está perfectamente definido y se reduce al valor de algunos parámetros adimensionales que también se suponen conocidos, así que no se requieren datos adicionales para hacer predicciones.

Los Capítulos 3 y 4 son, de alguna forma, los más instrumentales. Los métodos y técnicas ahí propuestos no se han aplicado al problema del GBM. Esto se debe a que, al estudiar las limitaciones del modelo paramétrico de GBM, detecté que es crucial trabajar en un marco capaz de incorporar variables no medibles, como las dos correcciones que comprenden el comportamiento *go-or-grow*, Π_{go} y Π_{gr}, como se ha expuesto en la discusión del Capítulo 6. La única forma de incorporar esta información utilizando PIDS era cambiar ligeramente el enfoque. El uso de métodos y técnicas basados en ANNs fue por supuesto otro argumento que justifica esta decisión.

10.1.3 Comentarios generales y perspectivas

Uno de los principales objetivos de esta tesis era desarrollar nuevas estrategias de PIDS sin perder el contacto con los problemas reales de bioingenería. Esta es la razón que nos ha llevado a, cuando fuera posible, utilizar los datos y estrategias propios de los problemas de bioingeniería, así como a utilizar datos experimentales reales. En consecuencia, y a pesar del marcado carácter metodológico de esta disertación, los métodos se han aplicado hasta en tres ocasiones a datos reales:

1. El *solver* RBDD para el modelado de hormigón se utilizó con resultados reales de ensayos uniaxiales.
2. Un *solver* guiado por datos se ha utilizado, combinado con los procedimientos descritos para completar datos, para el análisis mecánico y estructural del huesco cortical en un problema multiescala. Los datos utilizados se obtuvieron a partir de correlación digital de imágenes (DIC por sus siglas en inglés).
3. El modelo parámetrico de evolución del GBM en dispositivos microfluídicos se ajustó utilizando datos realez obtenidos de dispositivos microfluídicos, tanto de experimentos en curso como realizados anteriormente en el grupo TMELAB[34].

La extensa literatura sobre PIDS está principalmente centrada en los métodos, y habitualmente utiliza datos *in silico* para validar las aproximaciones propuestas, como también se hace frecuentemente a lo largo de esta disertación. En efecto, hay muy pocos trabajos que incorporen datos reales a los calculos en PIDS utilizando técnicas de adquisición de datos (consultar, por ejemplo, Badías et al. (2020); Moya et al. (2020a)). En este trabajo, la adquisición de datos ha sido posible gracias a la utilización de la tecnología microfluídica[35] combinada con biomarcadores específicos, abriendo la puerta al uso de técnicas de Aprendizaje Automático (ML por sus siglas en inglés) para la monitorización de cultivo celular (Galan et al., 2020). Hasta donde llega mi conocimiento, no hay otro trabajo que combine estos tres pilares fundamentales: herramientas de ML o de Inteligencia Artificial (AI por sus siglas en inglés), datos reales en el área de la biomedicina o bioingeniería obtenidos a partir de una tecnología reciente y novedosa, y la incorporación de la física de una forma consistente.

[34]Enlace a la página web del grupo: https://tmelab.unizar.es/

[35]Durante la tesis, también se han utilizado datos obtenidos de ensayos DIC, pero no los consideramos en este momento ya que no son el núcleo de la tesis.

Para ser justo, debo especificar que el objetivo último que se desea alcanzar es la utilización de los métodos presentados en el Capítulo 5 y particularizados para el problema de evolución del GBM en el Capítulo 7 a datos reales obtenidos de cultivos celulares en dispositivos microfluídicos. Sin embargo, esto requiere de un control experimental y técnicas automatizadas de adquisición de imágenes que aún no están disponibles (pero se espera que lo estén pronto) en nuestro laboratorio. Habrían de señalarse dos limitaciones principales:

- La capacidad de medir cuantitativamente y con precisión la concentración de oxígeno en los dispositivos, extensivamente y no solo en puntos específicos de los mismos.
- La capacidad de capturar y procesar la evolución celular (o, mejor dicho, las señales de los biomarcadores) en tiempo real y sin afectar a las condiciones del cultivo. Algunos marcadores utilizados para la señalización por fluorescencia cambian el estado de las células y por tanto solo se pueden utilizar al final del experimento[36].

En cualquier caso, los métodos guiados por datos desarrollados, particularmente el método de imputación para datos incompletos consistente con la física, pueden ser de ayuda en este sentido.

10.2 Contribuciones de la tesis

La contribución fundamental de la tesis es el desarrollo de herramientas específicas de PIDS para el análisis de la evolución del GBM en dispositivos microfluídicos. Este es el primer trabajo que ha abordado esta tarea con, al menos, los siguientes ingredientes, en un único marco:

- El enfoque combina el conocimiento físico sobre un sistema con un uso intensivo de datos experimentales, obtenido de plataformas singulares como los dispositivos microfluídicos. Los métodos que se presentan aquí están por tanto adaptados para un nuevo paradigma en las ciencias biológicas y la bioingeniería, los cuales no han sido explotados previamente debido a la falta de técnicas de generación de datos.
- Poner el foco en el comportamiento *go-or-grow* en células hipóxicas ha requerido de una aproximación única, utilizando el concepto de variables internas. Antes de esta tesis, no existían métodos capaces de tratar con variables internas no medibles de forma robusta y flexible, para analizar PDEs, como las utilizadas para modelar la evolución del GBM.

A continuación, se describen otras contribuciones técnicas relacionadas de alguna forma con lo anterior, perteneciendo a las dos direcciones principales de esta tesis: la metodológica, relacionada con el diseño de nuevas herramientas de PIDS, y la aplicada, relacionada con modelos de evolución del GBM. Esta enumeración no coincide estrictamente con las dos partes en las que se divide la exposición de la tesis.

[36] Como curiosidad, este es un caso paradigmático donde la medida/observación de una determinada variable afecta al experimento, sin necesidad de hablar de mecánica cuántica y acudir hasta la escala de longitud de los átomos.

10.2.1 Contribuciones metodológicas

Desde el punto de vista metodológico, se pueden enumerar las siguientes contribuciones:

1. He formulado un marco general y abstracto para PIDS, donde se pueden situar cada uno de los métodos mencionados.
2. He adaptado algunos de los métodos de PIDS existentes para problemas del "mundo real" con datos imperfectos:
   - He extendido los *solvers* guiados por datos dando lugar al *solver* RBDD, capaz de incorporar la incertidumbre en los datos y de trabajar adecuadamente en contextos con datos con ruido.
   - He definido un nuevo método PIDS de imputación estadística. Este método permite completar huecos en conjuntos de datos de forma consistente con la física y se puede combinar con otros esquemas guiados por datos.
3. He establecido una nueva familia de métodos PIDS, las llamadas PGNNIVs, capaces de incorporar, de forma flexible, la física del problema (leyes universales, ecuaciones constitutivas, o ambas) bajo el prisma de las teorías de campo, que se formulan utilizando variables de estado no medibles.
4. He construido un marco computacional para simular cultivos celulares en dispositivos microfluídicos. Este marco incluye:
   - Un modelo matemático para modelar el crecimiento, muerte y diferenciación celular, así como el movimiento aleatorio de las células, la quimiotaxis, mecanotaxis, termotaxis y electrotaxis, la remodelación celular y las no linealidades asociadas a todos estos fenómenos.
   - Un *software* tridimensional, basado en el Método de los Elementos Finitos (FEM por sus siglas en inglés), capaz de simular todos los fenómenos mencionados en el punto anterior.
   - Un *software* unidimensional, basado en una librería existente en Matlab, para simular la progresión del GBM en dispositivos microfluídicos bajo la hipótesis unidimensional en tiempos de simulación razonables.
5. He propuesto una metodología estocástica original, basada en cópulas, para estudiar modelos paramétricos no explícitos (como los que son resultados de simulación numérica), capaces de incorporar la incertidumbre de los parámetros y su dependencia estructural. Este método también es capaz de detectar limitaciones del modelo relacionadas con los datos disponibles, así como de orientar en el diseño de nuevos experimentos y capañas experimentales.
6. He explotado el método de características como un método apropiado para analizar cultivos celulares en dispositivos microfluídicos bajo la aproximación unidimensional, consiguiendo analizar algunos rasgos del modelo como la formación de estructuras migratorias (pseudoempalizadas).

10.2.2 Aplicaciones

Desde el punto de vista de los resultados y aplicaciones, podemos enumerar las siguientes contribuciones:

1. He aplicado el *solver* RBDD a diferentes problemas, incluyendo algunos con datos reales, como el modelado del hormigón y la biomecánica del hueso.
2. He demostrado que el procedimiento de llenado de huecos puede combinarse con el métido DIC para acelerar los cálculos multiescala, en el contexto de la ingeniería de tejidos.
3. He demostrado que las PGNNIVs mejoran los métodos basados en ANNs estándar respecto a los datos requeridos, el filtrado de ruido, la convergencia y la capacidad de extrapolación. Es más, son los únicos que tienen capacidad explicativa (capacidad de descifrar ecuaciones constitutivas), además de capacidad predictiva (capacidad de predecir el estado del sistema en un punto que no forma parte del conjunto de datos).
4. He definido un modelo paramétrico capaz de recrear la evolución del GBM en condiciones hipóxicas en dispositivos microfluídicos. Este modelo paramétrico es un sistema de PDEs, que incluye 11 parámetros. El modelo es capaz de reproducir resultados experimentales en distintas configuraciones, como formación de pseudoempalizadas o doble pseudoempalizadas, (con uno y dos canales oxigenados respectivamente) y la formación de núcleos necróticos (con altas concentraciones celulares).
5. He detectado las limitaciones del modelo propuesto para la evolución del GBM, en particular en relación con el cambio metabólico en el *go-or-grow*. Esto ha sido posible gracias a la metodología estocástica presentada.
6. He aplicado las PGNNIV al problema de evolución del GBM, siendo capaz de capturar distintos modelos de transiciones candidatos para el *go-or-grow*, de forma no paramétrica y mejorando los ajustes paramétricos tradicionales tanto en capacidad predictiva como explicativa, además de hacerlo para distintas condiciones de contorno.
7. He estimado con precisión la formación de pseudoempalizadas y el tiempo que les cuesta alcanzar una estructura oxigenada distante, el canal lateral, que hace el papel de vaso oxigenado.

10.3 Resultados científicos y tecnológicos

Resumimos aquí los resultados científicos y tecnológicos de esta tesis. La lista detallada de publicaciones, trabajos publicados y participaciones en congresos se puede encontrar más adelante.

10.3.1 *Software* desarrollado

Durante la tesis, he desarrollado varios códigos científicos y paquetes de *software*:

1. Un software basado en FEM para la simulación de cultivos celulares en dispositivos microfluídicos, implementado desde cero en Matlab y que tiene en cuenta las no linealidades, el acoplamiento entre las diferentes variables de campo y la remodelación de tejidos. Este software ha sido validado para una amplia clase de problemas.
2. Varios paquetes de software y rutinas, en Matlab, relacionados con el *solver* RBDD y el tratamiento de datos incompletos. Algunos de ellos se utilizan para resolver el problema unidimensional expuesto en el Capítulo 3 y el problema 3D del hormigón bajo las hipótesis establecidas. Además, se ha desarrollado un código para resolver el procedimiento de llenado de huecos en el caso lineal. Finalmente, un código para la técnica de completado de datos para el problema biomecánico macroscópico y una rutina para los procedimientos de emparejamiento de escalas en el enfoque multiescala.
3. Distintos paquetes de *software* y rutinas para el análisis de modelos paramétricos usando cópulas, incluyendo análisis de modelos, predicciones probabilísticas y cálculos de utilidad experimental. Estos códigos han sido desarrollados en colaboración con otra estudiante de doctorado (Marina Pérez Aliacar).
4. Algunos paquetes de software y rutinas, en TensorFlow@Python, para resolver los diferentes problemas de PGNNIV expuestos. Estos incluyen el problema de predicción y caracterización hidráulica, la ecuación de difusión (heterogénea y no lineal), el problema de evolución de GBM y varios códigos relacionados con la mecánica de sólidos que constituirán el núcleo de futuras publicaciones.

10.3.2 Modelos validados

Como subproducto de esta tesis, se puede considerar un modelo paramétrico para la evolución de GBM en dispositivos microfluídicos, que consiste en un sistema de PDEs con 11 parámetros. Este modelo ha sido ajustado a datos experimentales y validado usando diferentes configuraciones experimentales.

10.3.3 Resumen de publicaciones

Durante la tesis, he publicado en revistas y editoriales científicas.

1. Cinco (5) artículos de revista como primer autor en revistas científicas en el primer quartil.
2. Dos (2) artículos de revista como uno de los autores participante, en revistas de primer y segundo cuartil, que presentan distintas aplicaciones de los métodos desarrollados en esta tesis.
3. Un (1) capítulo de libro, como primer autor, de la editorial Elsevier.

Además, tenemos los siguientes artículos en proceso:

1. Un (1) artículo de revista como primer autor en revisión en una revista del primer cuartil *PLoS Computational Biology*.

2. Un (1) artículo de revista como primer autor disponible en un repositorio público (*arXiv*), pendiente de envío.
3. Un (1) artículo de revista en preparación para *Bulletin of Mathematical Biology*.
4. Un (1) artículo de revista planificado sobre la aplicación de PGNNIV en mecánica de sólidos.

Finalmente, he participado en multitud de congresos nacionales e internacionales (como primer autor y ponente):

1. Una (1) participación como uno de los moderadores de una sesión especial en un congreso internacional.
2. Nueve (9) comunicaciones orales en congresos internacionales.
3. Dos (2) comunicaciones en formato póster en congresos internacionales.
4. Una (1) participación como moderador de una sesión paralela en un congreso nacional.
5. Siete (7) comunicaciones orales en eventos locales o nacionales.
6. Cuatro (4) comunicaciones en formato póster en congresos locales o nacionales.

10.4 Conclusiones de la tesis

En resumen, las principales conclusiones de esta disertación son las siguientes:

Principales conclusiones de la tesis

- Es posible adaptar los métodos de Simulación Basada en Datos en Ingeniería y Ciencias (DDSBES por sus siglas en inglés) a situaciones o contextos de datos con ruido o incompletos.
- Para problemas físicos con variables internas y/o no medibles, las PGNNIVs han demostrado ser un método PIDS con capacidad tanto predictiva como explicativa.
- La evolución del GBM en dispositivos microfluídicos puede modelarse con un modelo paramétrico. Este modelo es capaz de recrear la evolución de cultivos celulares en tres configuraciones experimentales diferentes, pero no es capaz de explicar completamente la transición de *go-or-grow*.
- El uso de PGNNIV combinadas con el problema de evolución del GBM es capaz de paliar las carencias del modelo paramétrico y de explicar los cambios metabólicos como el *go-or-grow*.

Otras conclusiones de cada capítulo son:

- **Capítulo 3:**
  - Los métodos de resolución guiados por datos son mucho más robustos en contextos no lineales que las técnicas de regresión o interpolación.

– El RBDD solver es robusto en contextos con incertibumbre en los datos, permite propagar dicha incertidumbre y puede ser aplicado a datos de diferentes experimentos.

Capítulo 4:

– La nueva técnica de aprendizaje no supervisado presentada para completar datos mejora algunas técnicas de imputación estadística y puede corregir sesgos no físicos de los datos.
– Esta técnica puede ser usada en contextos multiescala para acelerar los cálculos sin perder el acceso a los datos microestructurales.

Capítulo 5:

– Las PGNNIV superan a las clásicas ANNs en aspectos como la velocidad de convergencia, los datos requeridos, el filtrado del ruido y la capacidad de extrapolación.
– Las PGNNIV pueden incorporar las características más comunes de los operadores de física del continuo tanto lineales como no lineales.

Capítulo 6:

– Es posible desarrollar un marco computacional general para modelar cultivos celulares en dispositivos microfluídicos.
– Los modelos paramétricos biológicos y biomédicos son complejos y necesitan de un marco estadístico no lineal para analizar la dispersión y la correlación de los parámetros del modelo.

Capítulo 7:

– Las PGNNIVs aplicadas al problema de evolución del GBM superan a los ajustes paramétricos estándar tanto en capacidad predictiva como explicativa.
– Como las PGNNIV son capaces de trabajar correctamente con condiciones de contorno variables, el enfoque presentado es un primer paso hacia la medicina personalizada.

Capítulo 8:

– Es posible recrear las características histológicas del GBM, como las pseudoempalizadas y las oclusiones de vasos sin la necesidad de simulaciones numéricas.

10.5 Trabajo futuro

Parafraseando al ingeniero Fred Brooks[37]: "Las tesis no se acaban, se abandonan", así que hay margen de mejora para futuras investigaciones. Para clarificar la exposición

[37]Frederick P. Brook es Profesor Emérito en la Universidad de Carolina del Norte de Chapel Hill, https://cs.unc.edu/person/frederick-p-brooks-jr/

de esta sección, se incluyen las diferentes líneas futuras asociadas con los capítulos correspondientes de la disertación, y a continuación se presentan algunos aspectos más generales.

10.5.1 Líneas futuras correspondientes a tareas específicas

Capítulo 3

En el Capítulo 3, propusimos el RBDD *solver*, capaz de incorporar la incertidumbre de los datos a los cálculos. Hay varias posibles direcciones futuras:

- Un posible trabajo era extender la metodología RBDD a situaciones más complejas, como dinámica, inelasticidad, mecánica de la fractura... Sin embargo, y de forma simultánea al desarrollo de esta tesis, M. Ortiz y sus colaboradores han cubierto todos y cada uno de estos aspectos (ver el Capítulo 2 para más detalles).
- Otra línea futura es desarrollar estrategias de muestreo a partir de una estimación del error, basadas en fiabilidad y guiada por datos (RBDD). Como el *solver* RBDD proporciona una estimación del error (la distancia final entre la variedad física y el punto más cercano que pertenece al conjunto de datos), es posible usarla para evaluar el espacio cubierto por los datos y, por lo tanto, para muestrear nuevos puntos que a su vez alimentarán el *solver* y mejoren su rendimiento.
- Como el *solver* RBDD (al igual que el guiado por datos o DD) depende en gran medida de los datos muestreados en el espacio de estados, es muy importante diseñar bases de datos completas y de alta calidad. Ésta no es una tarea fácil ya que, como se ilustra para la aplicación concreta estudiada, el espacio cubierto por la muestra aumenta notablemente con la complejidad del material, y si no se suponen hipótesis fiables (como isotropía, simetrías u homogeneidad), el problema se vuelve intratable desde una perspectiva DD. Una solución interesante para explotar en el futuro sería construir variedades de menor dimensión a partir de los datos de la muestra.

Capítulo 4

En el Capítulo 4, propusimos una nueva imputación estadística teniendo en cuenta el problema físico. Este método es sensible a la estructura de los datos y a las restricciones físicas externas. Algunas posibles líneas futuras son:

- Hemos presentado una metodología de llenado de huecos que resuelve el problema en una sola iteración para datos con una estructura lineal, y hemos presentado un algoritmo heurístico para estructuras no lineales, basado en la combinación de proyecciones lineales y el uso de la aplicación exponencial para regresar a las variedades no lineales. Este algoritmo es costoso y su rendimiento depende del contexto, ya que se basa en algunas propiedades deseables de las funciones y variedades involucradas, como la suavidad, la regularidad y la convexidad. Existe una necesidad obvia de establecer resultados matemáticos sólidos para condiciones de convergencia suficientes, estimaciones de error y tasas de convergencia en condiciones específicas.

- El método de llenado de huecos propuesto supone que se conocen todas las ecuaciones físicas del sistema. Por ejemplo, cuando se aplica a un problema multiescala, supone una determinada relación tensión-deformación en la microescala, dado que únicamente las deformaciones son medibles (utilizando técnicas PIDS). Si se desean conocer las tensiones a nivel microscópico, se deben derivar a partir de una ecuación constitutiva dada. Una dirección de trabajo futuro sería relajar esta suposición y adaptar los procedimientos DD a esta situación exigente.

Capítulo 5

En el Capítulo 5, definimos el concepto de PGNNIV, que ha demostrado ser muy útil como método PIDS capaz de trabajar con variables internas no medibles. Como se ha visto en la revisión del estado del arte en el Capítulo 2, combinar ANNs con información física es un punto candente en la investigación hoy en día (algunos ejemplos son PINNs, SPNNs, ANNs conscientes de la termodinámica, entre muchos otros). Algunas posibles direcciones de trabajo futuro son:

- La adaptación de la metodología a problemas continuos ha demostrado ser a la vez efectiva y desafiante. La discretización del problema debe realizarse con cuidado, como se ha demostrado para la ecuación de difusión, donde aumentar el número de nodos no mejora necesariamente la precisión de las predicciones. En ese sentido, aquí también hay una necesidad de trabajos matemáticos teóricos que exploren las condiciones suficientes o necesarias de la convergencia. En cierto modo, los resultados sobre las ANNs como aproximadores universales deben replantearse bajo el prisma de las PDEs. Los resultados teóricos obtenidos para PINNs pueden adaptarse para establecer resultados análogos para PGNNIVs.
- Otro desafío en la aplicación de PGNNIV al continuo es la maldición de la dimensión. Tal como están formuladas, las PGNNIVs tienen tantas variables de salida como grados de libertad tenga el problema. Por ejemplo, para la predicción de un campo de desplazamientos en un problema tridimensional con una malla de $100 \times 100 \times 100$ nodos, necesitamos una red $N = 3 \times 10^6$ salidas, lo cual es inabordable desde el punto de vista computacional. Afortunadamente, estamos más interesados en capturar características y tendencias globales que en valores de salida concretos, así que esto puede ser evitado con una definición apropiada de las Cantidades De Interés (QoIs por sus siglas en inglés). Otra aproximación para este problema es utilizar técnicas espectrales. Definir las salidas de la PGNNIV como coeficientes espectrales convertiría el problema en uno más manejable evitando así la maldición de la dimensión.
- Por último, las geometrías variables o irregulares representan también un verdadero reto, debido a la estructura intrínseca de las PGNNIV. Se ha resuelto un problema de geometría variable para el problema hidráulico en el Capítulo 5, pero la tarea es más difícil cuando la geometría está asociada con una malla, y no es posible una parametrización simple. Además, las interpolaciones estándar no logran predecir con precisión el valor de un campo determinado cuando no hay una estructura euclídea presente en los datos (como es el caso de las mallas estructuradas representadas por píxeles y vóxeles). De hecho, la interpolación debe realizarse en el espacio reducido latente y, a continuación, los resultados

deben proyectarse en el espacio original. Como se ha presentado en el estado del arte del Capítulo 2, hay muchos avances en esa dirección, en lo que se llama aprendizaje profundo (DL por sus siglas en inglés) geométrico, destacando la aparición de las redes neuronales basadas en grafos.

Capítulo 6

En el Capítulo 6 se ha definido un marco computacional para analizar la evolución de cultivos celulares en dispositivos microfluídicos, que ha sido después particularizado para la evolución del GBM. Además, hemos ajustado un modelo paramétrico a partir de datos experimentales y, utilizando un enfoque estocástico, hemos realizado análisis de sensibilidad y estructurales. Además de las limitaciones intrínsecas del modelo, que han sido superadas en el Capítulo 7, aquí hay algunas posibles líneas futuras:

- Desde el punto de vista del modelado, se han detectado algunas limitaciones para reproducir los resultados experimentales. En primer lugar, hemos detectado un desajuste entre los resultados experimentales y los predichos por el modelo, en particular cerca de los canales laterales, así como una sobrestimación de la población de células vivas en el centro de la cámara en experimentos con altas concentraciones celulares. Estos errores podrían ser explorados en más detalle incluyendo el efecto de la obstrucción en el flujo de oxígeno que puede producirse en cultivos con altas concentraciones de células. Esto podría hacerse incluyendo una corrección no lineal en el término flujo, asociada con la evolución del oxígeno. En cualquier caso, para realizar estas modificaciones en el modelo, sería con seguridad necesario incluir medidas del nivel de oxígeno, algo que no es tan sencillo desde el punto de vista experimental. Otra mejora del modelo sería incluir el efecto de la historia sobre el comportamiento metabólico celular. El estudio de los cambios (reversibles o no) que sufre una célula por la acción de los estímulos externos, cambiando la expresión génica (fenotipo) pero sin afectar a la secuencia de ADN, se conoce como epigenética (Jaenisch and Bird, 2003) y es un punto muy prolífico en la investigación hoy en día. La inclusión de la epigenética en el modelo corregiría algunas de las discrepancias observadas en las medidas en el corto plazo, y podría explicar mecanismos complejos como el precondicionamiento causado por la hipoxia, la adaptación celular y la resistencia a terapias.
- Desde el punto de vista computacional, hay una limitación intrínseca del modelo debido a un problema físico de multiescala. El tiempo característico de la difusión del oxígeno es mucho menor que el tiempo característico de la pedesis celular. En concreto, la razón entre las dos escalas temporales es $\rho = \frac{\tau_{O_2}}{\tau_{\text{cell}}} = \frac{D_n}{D_{O_2}} = 5 \times 10^{-5}$. El *software* desarrollado, basado en FEM requiere de muchas iteraciones para resolver el problema completo, especialmente para los primeros pasos de tiempo en los que el nivel de oxígeno aún no ha alcanzado el régimen estacionario. Este enfoque monolítico es costoso y hace que el problema sea inabarcable debido al coste computacional. Algunas posibles soluciones incluyen adoptar un enfoque escalonado para la resolución del problema o construir una superficie de respuesta relacionando ambas escalas temporales.

- Otro problema relacionado con el coste computacional es el enfoque Bayesiano del Diseño de Experimentos (DoE por sus siglas en inglés). Como el espacio de medida en los dispositivos microfluídicos es grande (la concentración celular en cada punto de medida), los cálculos de la utilidad son muy costosos ya que requieren el cálculo de integrales múltiples en espacios de alta dimensionalidad. El uso de técnicas de integración apropiadas como el Método de Monte Carlo basado en cadenas de Markov (MCMCM) aligeraría los costes.

Capítulo 7

En el Capítulo 7 hemos aplicado el concepto de PGNNIV al problema de la evolución de GBM, en particular, al estudio del comportamiento *go-or-grow*.

- Hemos propuesto un método para desentrañar de forma no paramétrica el comportamiento *go-or-grow*. Como el método es intrínsecamente no paramétrico, es necesario definir nuevas herramientas para la incorporación de la incertidumbre. Las Redes Neuronales Bayesianas (BNNs por sus siglas en inglés) probabilísticas son sin duda una herramienta que debe explorarse para establecer una forma de incorporar esta incertidumbre a los parámetros de la red y propagarla al descubrimiento de la ecuación de estado. Otro enfoque posible sería utilizar una caracterización estadística de segundo orden de la capa de salida (por ejemplo, utilizando la distancia de Mahalanobis) y propagar estadísticas cuadráticas a las capas internas de la red. De hecho, este útlimo trabajo está en proceso.
- Por supuesto, este método está concebido para ser utilizado con datos experimentales reales, en lugar de con datos generados sintéticamente. El enfoque seguido está justificado por el hecho de que nuestro objetivo era evaluar si el método es capaz de desentrañar las transiciones del *go-or-grow* para diferentes dependencias funcionales. Sin embargo, será con su aplicación para desentrañar el comportamiento real de las células cuando este método aportará valor añadido a la investigación. Una vez que se aprende la dependencia no paramétrica, es posible establecer una transición entre el modelo propuesto y muchos candidatos asociados a distintas vías metabólicas, desentrañando así los fenómenos mecanísticos y moleculares que impulsan este cambio de comportamiento, y arrojando luz sobre los procesos biológicos implicados en la evolución del GBM.
- Como se ha mencionado anteriormente, el modelo de evolución de GBM podría enriquecerse teniendo en cuenta la epigenética y, por lo tanto, la dependencia del comportamiento con la historia de las células. Esto, en el ámbito de las PGNNIV, puede ser planteado usando Redes Neuronales Recurrentes (RNN por sus siglas en inglés), que fueron diseñadas precisamente para tratar series históricas y análisis de series temporales.

Capítulo 8

En el Capítulo 8 hemos aplicado el método de las características para derivar algunas características globales de la solución. Aunque se derivaron características importantes como las ondas viajeras, que tienen su contraparte en las características histológicas (pseudopalisades), todavía hay margen de mejora en este campo.

- Hemos obtenido la solución en la región externa utilizando el método de características, pero se podría explorar también en la región de transición (solución interior) y en los límites (capa límite), utilizando técnicas matemáticas convencionales (teoría asintótica). Sin embargo, ésta no es una tarea fácil ya que las fuertes no linealidades del problema complican el uso de herramientas matemáticas para derivar soluciones analíticas.
- El enfoque seguido en este capítulo está centrado en problemas unidimensionales. Una posibilidad evidente de trabajo futuro es extender los resultados a 2D y 3D (como el de la progresión del GBM en el tejido cerebral). Esto enriquecerá enormemente el análisis, ya que se podrían explorar algunas características como la heterogeneidad del dominio y su efecto sobre la forma de las superficies de frente de onda.

10.5.2 Líneas futuras correspondientes a tareas generales

Además de las líneas futuras específicas asociadas con los distintos capítulos, otras direcciones de trabajo más generales son:

- Utilizar el modelo de evolución del GBM para explorar nuevas estrategias de tratamiento y diseño de fármacos. Por ejemplo, se puede investigar cómo actuar sobre tejidos oxigenados y estructuras de oxigenación (como los vasos sanguíneos), o utilizar fármacos que promuevan la hipoxia localizada, puede afectar a la evolución del GBM. Esta es una meta ambiciosa, pero sin duda es uno de los objetivos últimos de la ingeniería biomédica.
- Incorporar al modelo de evolución de GBM el efecto de la mecánica del sustrato. Aunque esto se ha considerado en el marco general establecido en el Capítulo 6, hay dos cosas importantes que deben señalarse. En primer lugar, hemos considerado un acoplamiento mecánico bidireccional pero con suposiciones simples: algunos parámetros mecánicos influyen en la progresión de la célula (afectando a los diferentes coeficientes o definiendo gradientes mecánicos, como los gradientes de rigidez o los gradientes de presión), que a su vez pueden modificar el valor de estos parámetros, que son además tratados como campos espaciales (remodelación mecánica). Sin embargo, esta remodelación mecánica se establece *a priori* usando relaciones de remodelación *ad-hoc*, por ejemplo, cómo las concentraciones celulares afectan las tensiones del tejido. Un análisis más apropiado del problema sería considerar también el problema mecánico y resolverlo por completo. De hecho, lo que realmente hacen las células es producir sustancias o proteínas, como fibras de colágeno, que afectan la estructura de la Matriz Extracelular (ECM por sus siglas en inglés) y, por lo tanto, cambian su rigidez. Este cambio de rigidez se trasladaría (al mismo nivel de deformación) a un cambio de estado tensional, que se obtendría resolviendo completamente el problema mecánico. Esto también es aplicable a fenómenos más complejos que involucran efectos viscosos, o que utilizan enfoques poroelásticos o poroviscoelásticos.
- El uso del marco PGNNIV se ha aplicado aquí a problemas ilustrativos, aunque conceptualmente simples. Una línea futura es aplicar este marco a problemas más complejos, especialmente del área de la Mecánica Computacional (CM por sus

siglas en inglés), y en particular al problema elástico e inelástico en deformaciones finitas, donde el modelado constitutivo también ha sido históricamente un cuello de botella debido a las limitaciones experimentales. Aunque las últimas décadas han sido muy fructíferas en el modelado constitutivo de sólidos inelásticos y tejidos blandos, este paso de modelado puede ser evitado gracias al uso de PGNNIV, evitando una selección de modelos *a priori*.

11. Thesis Dissemination

Publications in scientific journals

Publications as 1st author

1. Ayensa-Jiménez J., Sanz-Herrera, J. A., Doweidar M. H., Doblaré M. "A new reliability-based data-driven approach for noisy experimental data with physical constraints.", *Computer Methods in Applied Mechanics and Engineering*, 2018, **IF = 4.821** (**Q1**).
2. Ayensa-Jiménez J., Sanz-Herrera, J. A., Doweidar M. H., Doblaré M. "An unsupervised data completion method for physically-based data-driven models.", *Computer Methods in Applied Mechanics and Engineering*, 2019, **IF = 5.763** (**Q1**).
3. Ayensa-Jiménez J., Doweidar M. H., Sanz-Herrera, J. A., Doblaré M. "Prediction and identification of physical systems by means of Physically-Guided Neural Networks with meaningful internal layers.", *Computer Methods in Applied Mechanics and Engineering*, 2020, **IF = 6.756** (**Q1**).
4. Ayensa-Jiménez J., Pérez-Aliacar M., Randelovic T., Oliván S., Fernández L., Sanz-Herrera, J. A., Ochoa I., Doweidar M. H., Doblaré M. "Mathematical formulation and parametric analysis of in vitro cell models in microfluidic devices: application to different stages of glioblastoma evolution.", *Scientific Reports*, 2020, **IF = 4.380** (**Q1**).
5. Ayensa-Jiménez J., Pérez-Aliacar M., Randelovic T., Sanz-Herrera J. A., Doweidar M. H., Doblaré M. "Analysis of the parametric correlation in mathematical modeling of in vitro glioblastoma evolution using copulas.", *Mathematics*, 2020, **IF = 2.258** (**Q1**).

Other works directly related to the thesis

1. Mora-Macías J., Ayensa-Jiménez J., Reina-Romo E., Doweidar M. H., Domínguez J., Doblaré M., Sanz-Herrera J. A. "A multiscale data-driven approach for bone

tissue biomechanics.", *Computer Methods in Applied Mechanics and Engineering*, 2020, **IF = 6.756** (**Q1**).

2. Sanz-Herrera J. A., Mora-Macías J., Ayensa-Jiménez J., Reina-Romo E., Doweidar M. H., Domínguez J., Doblaré M. "Data-Driven Computational Simulation in Bone Mechanics.", *Annals of Biomedical Engineering*, 2021, **IF = 3.934 (2020)** (**Q2**).

In peer review process

1. Ayensa-Jiménez J., Sanz-Herrera J. A., Doweidar M. H., Doblaré M. "Understanding glioblastoma invasion using the unravelling power of physically-guided neural networks with internal variables.", *PLoS Computational Biology*.

In public repositories

1. Ayensa-Jiménez J., Sanz-Herrera J. A., Doweidar M. H., Doblaré M. "On the application of Physically-Guided Neural Networks with Internal Variables to Continuum Problems.", *arXiv*.

In preparation

1. Ayensa-Jiménez J., Doweidar M. H., Doblaré M., Gaffney E. A. "Cell dynamics in microfluidic devices under heterogeneous chemotaxis-growth conditions: a mathematical study.", *Bulletin of Mathematical Biology*.

Book chapters

1. Ayensa-Jiménez J., Doweidar M. H., Randelovic T., Fernández L., Oliván S., Ochoa I., Doblaré M. "On the Simulation of Organ-on-Chip Cell Processes: Application to an In Vitro Model of Glioblastoma Evolution." *Advances in Biomechanics and Tissue Regeneration.*, 2019, Elsevier.

Presentations at international conferences

Oral communications

1. Ayensa-Jiménez J., Doweidar M. H., Doblaré M. "A new Reliability-Based Data-Driven solver for different scales uncertainty propagation." Data-Best, 2017, Nantes (France).
2. Ayensa-Jiménez J., Doweidar M. H., Doblaré M. "A nonlinear FEM chemotaxis model for interacting cellular phenotypes: application to the study of hypoxia-driven pseudopallisades in glioblastoma." International Workshop on Modelling of Nonlinear Continua, 2017, Castro Urdiales (Spain).
3. Ayensa-Jiménez J., Doweidar M. H., Doblaré M. "Facing the missing data problem in Data-Driven Simulation Based Engineering and Sciences: the need of new and adapted statistical strategies." Data-Best, 2019, Paris (France).

4. Ayensa-Jiménez J., Pérez-Aliacar M., Doweidar M. H., Doblaré M. "Interacting cellular populations in organ-on-chip devices: a framework for parameter identification and simulation in Glioblastoma Multiforme." Congress on Numerical Methods in Engineering (CMN2019), 2019, Guimarães (Portugal).
5. Ayensa-Jiménez J., Doweidar M. H., Doblaré M. "Discovering state equations using Physically-Guided Neural Networks: coupling universal physics and data information under the deep learning framework." World Congress on Computational Mechanics (WCCM2020), 2021, Paris (France).
6. Ayensa-Jiménez J., Doweidar M. H., Doblaré M. Coupling universal physics and data information with Physically-Guided Convolutional Neural Networks for both predicting and explaining continuum physics. ECCOMAS Young Investigators Conference (YIC2021), 2021, Valencia (Spain).
7. Ayensa-Jiménez J., Doweidar M. H., Doblaré M. "Unravelling cell metabolism using Physically-Guided Neural Networks. Application to the understanding of Glioblastoma invasion process." Computer Methods in Biomechanics and Biomedical Engineering (CMBBE2021), 2021, Bonn (Germany).
8. Ayensa-Jiménez J., Doweidar M. H., Doblaré M. "Unravelling cell metabolism using Physically-Guided Neural Networks. Application to the understanding of Glioblastoma invasion process." Mechanistic Machine Learning and Digital Twins for Computational Science, Engineering & Technology (MMLDT-CSET2021), 2021, San Diego (USA).
9. Ayensa-Jiménez J., Pérez-Aliacar M., Randelovic T., Doweidar M. H., Sanz-Herrera J.A., Ochoa I., Doblaré M. "Bio In silico models for understanding and predicting the evolution of glioblastoma tumor in vitro." 3rd Sino-Spanish Research and Innovation Forum, 2021, online format.

Poster communications

1. Ayensa-Jiménez J., Bernal-Lecina M., Peña Baquedano E., Doweidar M. H., Doblaré M. "A novel model for the simulation of interacting celular populations in organ-on-chip devices. Application to the study of glioblastoma hypoxia-driven pseudo-palisades." World Congress of Biomechanics (WCB2018), 2018, Dublin (Ireland).
2. Ayensa-Jiménez J., Randelovic T., Pérez-Alicar M., Doweidar M. H., Ochoa I., Doblaré M, Fernández L. "Glioblastoma on chip – combination of in vitro and in silico models in the race for a good preclinical model." European Organ-On-Chip society annual meeting (EUROoCS2020), 2020, Uppsala (Sweden).

Moderated sessions

1. Sepecial session "Mathematical modelling and simulation of tumours", Computer Methods in Biomechanics and Biomedical Engineering (CMBBE2021), 2021, Bonn (Germany).

Presentations at national conferences

Oral communications

1. Ayensa-Jiménez J., Doweidar M. H., Doblaré M. "A new reliability-based data-driven approach to simulation-based models." 4th BSC Severo Ochoa International Doctoral Symposium, 2017, Barcelona (Spain).
2. Ayensa-Jiménez J. "Diagnóstico automático del cáncer de pulmón a partir de técnicas de Deep Learning e Imágenes médicas.", 2017, Zaragoza (Spain).
3. Ayensa-Jiménez J., Chaparro-Victoria D. "Herramientas para revalorizar las imágenes médicas: marco y posibilidades." I Workshop en NanoOncología, 2018, Zaragoza (Spain).
4. Ayensa-Jiménez J., Doweidar M. H., Sanz-Herrera J.A., Doblaré M. "Influencia del microentorno en la respuesta celular mediante técnicas mixtas de simulación y autoaprendizaje. Avances y resultados provisionales." V jornada de Doctorandos del Programa de Doctorado en Ingeniería Mecánica, 2018, Zaragoza (Spain).
5. Ayensa-Jiménez J., Doweidar M. H., Doblaré M. "Unravelling cell metabolic behaviour using Physically-Guided neural Networks: Towards personalized medicine in the fight against Glioblastoma", Ciclo de seminarios IPIBA, 2021, Zaragoza (Spain).
6. Ayensa-Jiménez J., Doweidar M. H., Sanz-Herrera J.A., Doblaré M. "Redes Neuronales Guiadas por Física para aprender el metabolismo celular: aplicación al estudio de la evolución del Glioblastoma Multiforme en condiciones hipóxicas." VII jornada de Doctorandos del Programa de Doctorado en Ingeniería Mecánica, 2020, Zaragoza (Spain).
7. Ayensa-Jiménez J., Doweidar M. H., Doblaré M, Gaffney E.A. Dinámica de poblaciones celulares bajo gradientes de un agente quimiotáctico. Una aproximación analítica. X Capítulo Español de la Sociedad Europea de Biomecánica, 2021, Granada (Spain).

Poster communications

1. Ayensa-Jiménez J., Pérez-Aliacar M., Randelovic T., Oliván S., Fernández L., Sanz-Herrera, J. A., Ochoa I., Doweidar M. H., Doblaré M. "Simulación de poblaciones celulares en dispositivos Organ-On-Chip: de los modelos clásicos a las simulaciones guiadas por datos." VII jornada de jóvenes investigadores del Instituto de Ingeniería de Aragón, 2018, Zaragoza (Spain).
2. Ayensa-Jiménez J., Doweidar M. H., Doblaré M. "A mathematical and computational model for the simulation of Interacting cellular populations in mechanical environments. Application to glioblastoma cell cultures in microfluidic devices." I Colloquium of the Spanish Society of Theoretical and Applied Mechanics (STAMS2018), 2018, Madrid (Spain).
3. Ayensa-Jiménez J., Doweidar M. H., Doblaré M. "Redes neuronales guiadas con variables internas físicas: cuando la física guía a los datos en el aprendizaje profundo." IX jornada de jóvenes investigadores del Instituto de Ingeniería de Aragón, 2020, Zaragoza (Spain).

4. Ayensa-Jiménez J., Doweidar M. H., Doblaré M. "De la realidad histológica a la metabólica: desentrañando la respuesta celular a partir de la evolución de cultivos celulares utilizando redes neuronales guiadas por la física." X jornada de jóvenes investigadores del Instituto de Ingeniería de Aragón, 2021, Zaragoza (Spain).

Moderated sessions

1. Sesión paralela de Mecanobiología, X Capítulo Español de la Sociedad Europea de Biomecánica, 2021, Granada (Spain).

Research stays

- Centre: Wolfson Centre For Mathematical Biology - Mathematical Institute (Oxford University).
- Supervisor: Prof. Eamonn A. Gaffney.
- Dates: April 1 to October 31, 2021 (214 days).
- Summary: During the stay stay, the main focus was on the study of the mathematical nature of the GBM evolution model in microfluidic devices that was previously developed. In particular, the work focused on the so-called *go-or-grow* paradigm and the development of mathematical tools for analysing it. Particularly, I worked on the derivation of global properties of solutions to the system of PDEs (total amount of cells, velocity of the pseudoepalisades, cell concentration far from the moving structures), depending on the mathematical nature of the growth and chemotaxis coefficients. Also, I was trained on some standard mathematical methods for analysing PDEs such as stability analysis, asymptotic expansions and perturbative methods, Wentzel-Kramers-Brillouin-Jeffreys (WKBJ) method and the method of multiple scales. The result of this stay is reflected in Chapter 8, one congress oral communication and in a paper that is under preparation.

Scientific dissemination

It is said that the work of a scientist is not only to investigate, but also to disseminate the results of their research to the general public. In that sense, the content of this thesis has appeared in the media in several occasions during these last years:

- On December 4th, 2017, I was interviewed on the TV show "En Ruta por la Ciencia", from local TV channel "Aragón televisión". During the programme, dedicated to Big Data, I had the opportunity to explain my contribution to GBM understanding using PIDS techniques, and about ML and AI methods applied to medical image problems.
- I was given the opportunity to write an article at the local newspaper "Heraldo de Aragón", in which I summarised the overview and main results of the thesis. This article was published on December 21th, and is reproduced in Appendix C.

- I participated at the competition "Tesis en 3 minutos", which consists in explaining the contents of your thesis in front of a multidisciplinary jury. The exposition may be found here.
- I participated at the competition "#HiloTesis", which consists in explaining the contents of your thesis in using a twitter thread. The thread may be found here.
- I have regularly participated in the series of internal seminars and journal clubs organised by the groups AMB and TMELab to share the work of the different Ph.D. students.

A. Mathematical concepts and proofs

A.1 Euclidean spaces and orthogonal projections

A.1.1 Mean Square Error function

First, we introduce the mean square error (MSE) function.

Definition A.1: Mean square error function

Let $\boldsymbol{x}_j \in \mathbb{R}^n$ and $w_j \in \mathbb{R}^+$, $j = 1, \dots, N$. The MEAN SQUARE ERROR function of the points $\{\boldsymbol{x}_j\}_{j=1,\dots,N}$ with associated weights $\{w_j\}_{j=1,\dots,N}$ is the function $\mathrm{MSE} : \mathbb{R}^n \to \mathbb{R}^+$ defined by:

$$\mathrm{MSE}(\boldsymbol{x}) = \sum_{j=1}^{N} w_j \|\boldsymbol{x} - \boldsymbol{x}_j\|. \tag{A.1}$$

Next, we introduce two familiar sample estimators of the position and spread of a set of points $\{\boldsymbol{x}_j\}_{j=1,\dots,N}$.

Definition A.2: Sample mean

Given $\boldsymbol{x}_j \in \mathbb{R}^n$ and $w_j \in \mathbb{R}^+$, $j = 1, \dots, N$. The SAMPLE WEIGHTED MEAN of the points $\{\boldsymbol{x}_j\}_{j=1,\dots,N}$ with associated weights $\{w_j\}_{j=1,\dots,N}$ is:

$$\bar{\boldsymbol{x}} = \frac{\sum_{j=1}^{N} w_j \boldsymbol{x}_j}{\sum_{j=1}^{N} w_j}. \tag{A.2}$$

Definition A.3: Sample variance - covariance matrix

Given $\boldsymbol{x}_j \in \mathbb{R}^n$ and $w_j \in \mathbb{R}^+$, $j = 1, \ldots, N$. The SAMPLE VARIANCE - COVARIANCE MATRIX of the points $\{\boldsymbol{x}_j\}_{j=1,\ldots,N}$ with associated weights $\{w_j\}_{j=1,\ldots,N}$ is:

$$\boldsymbol{S} = \frac{\sum_{j=1}^{N} w_j (\boldsymbol{x}_j - \bar{\boldsymbol{x}})(\boldsymbol{x}_j - \bar{\boldsymbol{x}})^\intercal}{\sum_{j=1}^{N} w_j}. \tag{A.3}$$

A classical result from probability theory (Papoulis, 1990) states the following:

Theorem A.1: Mean, variance and MSE minimization

The function MSE is minimised when $\boldsymbol{x} = \bar{\boldsymbol{x}}$ and the value of this minimum is $\mathrm{Tr}(\boldsymbol{S})$ where $\bar{\boldsymbol{x}}$ is the weighted mean value of the data and $\boldsymbol{S}$ their variance-covariance matrix given by Eqs. (A.2) and (A.3) respectively.

A.1.2 Orthogonal projection and properties

Definition A.4: Linear projection

Let E be a vector space and $F \subset E$ a vector subspace and G a supplementary subspace of F, that is, $E = F \oplus G$. Therefore, $\forall \boldsymbol{x} \in E$, $\exists! (\boldsymbol{x}', \boldsymbol{x}'') \in F \times H$ such that:

$$\boldsymbol{x} = \boldsymbol{x}' + \boldsymbol{x}''. \tag{A.4}$$

The LINEAR PROJECTION of F onto G is the linear map $\pi : E \to E$, $\boldsymbol{x} \mapsto \boldsymbol{x}'$.

The linear projection satisfies the following properties (Galántai, 2013):

- Any linear projection is an idempotent operator, $\pi^2 = \pi$.
- We have $F = \mathrm{Im}(\pi)$, $G = \mathrm{Ker}(\pi)$, that is, $E = \mathrm{Im}(\pi) \oplus \mathrm{Ker}(\pi)$.
- Any idempotent linear endomorphism p, $p^2 = p$, is a linear projection, Indeed, it is the linear projection of $\mathrm{Im}(p)$ onto $\mathrm{Ker}(p)$.
- If π is the projection of F onto G, therefore $I - \pi$ is the projection of G onto F, where I is the identity map.

Definition A.5: Orthogonal projection

Let E be a vector space and $F \subset E$ a vector subspace and G a supplementary subspace of F, that is, $E = F \oplus G$. A linear projection π is said ORTHOGONAL if $\mathrm{Ker}(\pi) = \mathrm{Im}(\pi)^\perp$.

We have the following result, which suggests an alternative but common definition for orthogonal projections:

Proposition A.1: Equivalent definition of orthogonal projections

A projection π defined in the vector space E is an orthogonal projection if and only if π is self adjoint.

Proof. **Necessity.** Let π be an orthogonal projection. If $\boldsymbol{x}, \boldsymbol{y} \in \mathrm{Im}(\pi)$, then $\boldsymbol{x} \cdot \pi(\boldsymbol{y}) = \pi(\boldsymbol{x}) \cdot \pi(\boldsymbol{y}) = \pi(\boldsymbol{x}) \cdot \boldsymbol{y}$. If $\boldsymbol{x}, \boldsymbol{y} \in \mathrm{Ker}(\pi)$, then $\boldsymbol{x} \cdot \pi(\boldsymbol{y}) = \boldsymbol{0} = \pi(\boldsymbol{x}) \cdot \boldsymbol{y}$. The general case is obtained using that $E = \mathrm{Im}(\pi) \oplus \mathrm{Ker}(\pi)$.

Sufficiency. If p^* is the adjoint of the linear map p, then $\mathrm{Ker}(\pi^*) = \mathrm{Im}(\pi)^\perp$ so, if $p^* = p$, $\mathrm{Ker}(p) = \mathrm{Im}(p)^\perp$. □

Orthogonal projections are important as they provide useful solutions for problems involving norm minimisation. Our aim is to solve the following problem: given a vector space E, a subspace $F \subset E$ and a vector $\boldsymbol{v} \in E$, find the vector $\boldsymbol{u} \in F$, which is closest to $\boldsymbol{v}$. First, let us clarify what *closest* means. The tool to measure distance in a Banach space is the norm, so we want $\|\boldsymbol{v} - \boldsymbol{u}\|$ to be as small as possible. That is, we want to find a vector $\boldsymbol{u} \in F$ such that $\|\boldsymbol{v} - \boldsymbol{u}\| \leq \|\boldsymbol{v} - \boldsymbol{w}\|$, all $\boldsymbol{w} \in F$. We have the following very important result:

Proposition A.2: Characterisation of the orthogonal projection (I)

Let E a Hilbert space, $F \subset E$ and $\boldsymbol{v} \in E$. Then:

$$\|\boldsymbol{v} - \pi_F(\boldsymbol{v})\| \leq \|\boldsymbol{v} - \boldsymbol{u}\|, \quad \forall \boldsymbol{u} \in F. \tag{A.5}$$

Furthermore, equality holds if and only if $\boldsymbol{u} = \pi_F(\boldsymbol{v})$.

Proof. Let $\boldsymbol{u} \in E$. Then, as $\boldsymbol{v} - \pi_F(\boldsymbol{v}) \in F^\perp$ and $\pi_{\mathcal{F}}(\boldsymbol{v}) - \boldsymbol{u} \in F$, using Pythagoras theorem:

$$\begin{aligned} \|\boldsymbol{v} - \boldsymbol{u})\|^2 &= \|\boldsymbol{v} - \pi_F(\boldsymbol{v})\|^2 + \|\pi_F(\boldsymbol{v}) - \boldsymbol{u}\|, \\ &\geq \|\boldsymbol{v} - \pi_F(\boldsymbol{v})\|^2. \end{aligned}$$

Furthermore, equality holds only if $\|\pi_F(\boldsymbol{v}) - \boldsymbol{u}\| = 0$, which is equivalent to $\pi_F(\boldsymbol{v}) = \boldsymbol{u}$. □

Proposition A.3: Characterisation of the orthogonal projection (II)

Let $\boldsymbol{x}_j \in \mathbb{R}^n$, $j = 1, \cdots, N$, such as $\bar{\boldsymbol{x}}$ and $\boldsymbol{S}$ are the sample mean and variance - covariance matrix of the vectors. Let $\mathcal{V} \subset \mathbb{R}^n$ be a linear manifold, then, the solution to the constrained minimisation problem:

$$\min_{\boldsymbol{x} \in \mathcal{V}} \quad \mathrm{MSE}(\boldsymbol{x}), \tag{A.6}$$

is given by:

$$\boldsymbol{x}^* = \pi_{\mathcal{V}}(\bar{\boldsymbol{x}}). \tag{A.7}$$

Moreover, if $s^2 = \mathrm{MSE}(\boldsymbol{x}^*)$, then:

$$s^2 = \sum_{j=1}^{N} w_j(\boldsymbol{x}_j - \pi_{\mathcal{V}}(\boldsymbol{x}_j))^2 + \sum_{j=1}^{N} w_j(\pi_{\mathcal{V}}(\boldsymbol{x}_j) - \boldsymbol{x}^*)^2. \tag{A.8}$$

Proof. The proof is again based on the Pythagoras theorem. So, we can get:

$$\begin{aligned}
\mathrm{MSE}(\boldsymbol{x}) &= \sum_{j=1}^{N} w_j(\boldsymbol{x}_j - \boldsymbol{x})^2 \\
&= \sum_{j=1}^{N} w_j \left[(\boldsymbol{x}_j - \pi_{\mathcal{V}}(\boldsymbol{x}_j))^2 + (\pi_{\mathcal{V}}(\boldsymbol{x}_j) - \boldsymbol{x})^2\right] \\
&= \sum_{j=1}^{N} w_j(\boldsymbol{x}_j - \pi_{\mathcal{V}}(\boldsymbol{x}_j))^2 + \sum_{j=1}^{N} w_j(\pi_{\mathcal{V}}(\boldsymbol{x}_j) - \boldsymbol{x})^2.
\end{aligned} \tag{A.9}$$

If we analyse this last expression, the only term depending on $\boldsymbol{x}$, is $\sum_{j=1}^{N} w_j(\pi_{\mathcal{V}}(\boldsymbol{x}_j) - \boldsymbol{x})^2$, then the minimum is achieved for $\boldsymbol{x}^* = \overline{\pi_{\mathcal{V}}(\boldsymbol{x}_j)} = \pi_{\mathcal{V}}(\bar{\boldsymbol{x}})$, where, in the last expression, we have used the linearity of the projection operator. Additionally, using Eq. (A.9), and $s^2 = \mathrm{MSE}(\boldsymbol{x}^*)$, we obtain:

$$s^2 = \mathrm{MSE}(\boldsymbol{x}^*) = \sum_{j=1}^{N} w_j(\boldsymbol{x}_j - \pi_{\mathcal{V}}(\boldsymbol{x}_j))^2 + \sum_{j=1}^{N} w_j(\pi_{\mathcal{V}}(\boldsymbol{x}_j) - \boldsymbol{x}^*)^2. \tag{A.10}$$

□

A.2 Results from probability theory

A.2.1 Stochastic quadratic forms

Now introduce stochastic quadratic forms (SQFs) and their fundamental properties:

Definition A.6: Stochastic quadratic form

Let $\boldsymbol{X} = (X_1, \dots, X_n)$ be a random vector and $\boldsymbol{M}$ a symmetric positive-definite matrix.
The Stochastic Quadratic Form $Q_{\boldsymbol{M}}(\boldsymbol{X})$ is the random variable defined as:

$$Q_{\boldsymbol{M}}(\boldsymbol{X}) = \boldsymbol{X}^\intercal \boldsymbol{M} \boldsymbol{X}. \tag{A.11}$$

Lemma A.1: Moments associated with SQF

Let $Q_{\boldsymbol{A}}(\boldsymbol{X})$ and $Q_{\boldsymbol{B}}(\boldsymbol{X})$ be two SQF, and let $\{\boldsymbol{e}_k\}_{k=1,\dots,n}$ the standard basis in $\mathbb{R}^n$, then:

$$\mathbb{E}[Q_{\boldsymbol{A}}(\boldsymbol{X})] = A_{ij}\Omega_{ij}, \tag{A.12}$$

$$\mathbb{E}[Q_{\boldsymbol{A}}(\boldsymbol{X})\boldsymbol{X}] = A_{ij}\Lambda_{ijk}\boldsymbol{e}_k, \tag{A.13}$$

$$\mathbb{E}[Q_{\boldsymbol{A}}(\boldsymbol{X})Q_{\boldsymbol{B}}(\boldsymbol{X})] = A_{ij}B_{kl}\Upsilon_{ijkl}, \tag{A.14}$$

$$\mathrm{Cov}(Q_{\boldsymbol{A}}(\boldsymbol{X}), Q_{\boldsymbol{B}}(\boldsymbol{X})) = A_{ij}B_{kl}\left(\Upsilon_{ijkl}(\boldsymbol{X}) - \Omega_{ij}(\boldsymbol{X})\Omega_{kl}(\boldsymbol{X})\right), \tag{A.15}$$

where $\Omega_{ij} = \mathbb{E}[X_iX_j]$, $\Lambda_{ijk} = \mathbb{E}[X_iX_jX_k]$ and $\Upsilon_{ijkl} = \mathbb{E}[X_iX_jX_kX_l]$, are the second, third and fourth moment tensors associated with the random vector $\boldsymbol{X}$.

Proof. Using index notation $Q_{\boldsymbol{A}}(\boldsymbol{X}) = A_{ij}X_iX_j$, and therefore:

$$\begin{aligned}\mathbb{E}[Q_{\boldsymbol{A}}(\boldsymbol{X})] &= \mathbb{E}[A_{ij}X_iX_j],\\ &= A_{ij}\mathbb{E}[X_iX_j],\\ &= A_{ij}\Omega_{ij}(\boldsymbol{X}),\end{aligned}$$

$$\begin{aligned}\mathbb{E}[Q_{\boldsymbol{A}}(\boldsymbol{X})\boldsymbol{X}] &= \mathbb{E}[A_{ij}X_iX_jX_k\boldsymbol{e}_k],\\ &= A_{ij}\boldsymbol{e}_k\mathbb{E}[X_iX_jX_k],\\ &= A_{ij}\Lambda_{ijk}(\boldsymbol{X})\boldsymbol{e}_k,\end{aligned}$$

$$\begin{aligned}\mathbb{E}[Q_{\boldsymbol{A}}(\boldsymbol{X})Q_{\boldsymbol{B}}(\boldsymbol{X})] &= \mathbb{E}[A_{ij}X_iX_jB_{kl}X_kX_l],\\ &= A_{ij}B_{kl}\mathbb{E}[X_iX_jX_kX_l],\\ &= A_{ij}B_{kl}\Upsilon_{ijkl}(\boldsymbol{X}).\end{aligned}$$

Finally:

$$\begin{aligned}&\mathrm{Cov}(Q_{\boldsymbol{A}}(\boldsymbol{X}), Q_{\boldsymbol{B}}(\boldsymbol{X}))\\ &= \mathbb{E}\left[(A_{ij}X_iX_j - A_{ij}\mathbb{E}[X_iX_j])(B_{kl}X_kX_l - B_{kl}\mathbb{E}[X_kX_l])\right]\\ &= A_{ij}B_{kl}\mathbb{E}\left[X_iX_jX_kX_l - \mathbb{E}[X_iX_j]X_kX_l - X_iX_j\mathbb{E}[X_kX_l] + \mathbb{E}[X_iX_j]\mathbb{E}[X_kX_l]\right]\\ &= A_{ij}B_{kl}\left(\mathbb{E}[X_iX_jX_kX_l] - 2\mathbb{E}[X_iX_j]\mathbb{E}[X_kX_l] + \mathbb{E}[X_iX_j]\mathbb{E}[X_kX_l]\right)\\ &= A_{ij}B_{kl}\left(\Upsilon_{ijkl}(\boldsymbol{X}) - \Omega_{ij}(\boldsymbol{X})\Omega_{kl}(\boldsymbol{X})\right).\end{aligned}$$

□

Proposition A.4: Expectation of a SQF

Let $Q_{\boldsymbol{A}}(\boldsymbol{X})$ be a SQF and let $\boldsymbol{\mu} = \mathbb{E}[\boldsymbol{X}]$ be the expected value of $\boldsymbol{X}$ and $\boldsymbol{\Sigma} = \mathbf{Cov}(\boldsymbol{X})$ the variance - covariance matrix of $\boldsymbol{X}$. Then:

$$\mathbb{E}[Q_{\boldsymbol{A}}(\boldsymbol{X})] = \mathrm{Tr}(\boldsymbol{A}\boldsymbol{\Sigma}) + \boldsymbol{\mu}^\intercal \boldsymbol{A}\boldsymbol{\mu}. \tag{A.16}$$

Proof. Following Lemma A.1, $\mathbb{E}[Q_{\boldsymbol{A}}(\boldsymbol{X})] = M_{ij}\Omega_{ij}$. Furthermore, $\Omega_{ij} = \Sigma_{ij} + \mu_i\mu_j$, and then:

$$\begin{aligned}\mathbb{E}[Q_{\boldsymbol{A}}(\boldsymbol{X})] &= A_{ij}(\Sigma_{ij} + \mu_i\mu_j),\\ &= A_{ij}\Sigma_{ij} + A_{ij}\mu_i\mu_j,\\ &= \mathrm{Tr}(\boldsymbol{A}\boldsymbol{\Sigma}) + \boldsymbol{\mu}^\intercal\boldsymbol{A}\boldsymbol{\mu}.\end{aligned}$$

In the last equality, we have used $\mathrm{Tr}(\boldsymbol{A}\boldsymbol{B}) = \boldsymbol{A} : \boldsymbol{B}$. □

Proposition A.5: Variance of a SQF and covariance of two SQF

Let $Q_{\boldsymbol{A}}(\boldsymbol{X})$ and $Q_{\boldsymbol{B}}(\boldsymbol{X})$ be two SQF and let $\boldsymbol{\mu} = \mathbb{E}[\boldsymbol{X}]$ be the expected value of $\boldsymbol{X}$, $\boldsymbol{\Sigma} = \mathbf{Cov}(\boldsymbol{X})$ the variance - covariance matrix of $\boldsymbol{X}$ and $\boldsymbol{\Upsilon} = \mathbb{M}_4(\boldsymbol{X})$ the fourth order moment tensor of $\boldsymbol{X}$. Then:

$$\mathrm{Cov}(Q_{\boldsymbol{A}}(\boldsymbol{X}), Q_{\boldsymbol{B}}(\boldsymbol{X})) = \boldsymbol{A} : \boldsymbol{\Upsilon} : \boldsymbol{B} - (\mathrm{Tr}(\boldsymbol{A}\boldsymbol{\Sigma}) + \boldsymbol{\mu}^\intercal\boldsymbol{A}\boldsymbol{\mu})\,(\mathrm{Tr}(\boldsymbol{B}\boldsymbol{\Sigma}) + \boldsymbol{\mu}^\intercal\boldsymbol{B}\boldsymbol{\mu})\,. \tag{A.17}$$

In particular, if $\boldsymbol{A} = \boldsymbol{B}$:

$$\mathrm{Var}(Q_{\boldsymbol{A}}(\boldsymbol{X})) = \boldsymbol{A} : \boldsymbol{\Upsilon} : \boldsymbol{A} - (\mathrm{Tr}(\boldsymbol{A}\boldsymbol{\Sigma}) + \boldsymbol{\mu}^\intercal\boldsymbol{A}\boldsymbol{\mu})^2. \tag{A.18}$$

Proof. Following Lemma A.1, $\mathrm{Cov}(Q_{\boldsymbol{A}}(\boldsymbol{X}), Q_{\boldsymbol{A}}(\boldsymbol{X})) = A_{ij}B_{kl}\Upsilon_{ijkl} + A_{ij}B_{kl}\Omega_{ij}\Omega_{kl}$. However, $\Omega_{ij} = \Sigma_{ij} + \mu_i\mu_j$, and therefore:

$$\begin{aligned}\mathrm{Cov}(Q_{\boldsymbol{A}}(\boldsymbol{X}), Q_{\boldsymbol{A}}(\boldsymbol{X})) &= A_{ij}B_{kl}\Upsilon_{ijkl} + A_{ij}B_{kl}(\Sigma_{ij} + \mu_i\mu_j)(\Sigma_{kl} + \mu_k\mu_l),\\ &= A_{ij}B_{kl}\Upsilon_{ijkl} + (A_{ij}\Sigma_{ij} + A_{ij}\mu_i\mu_j)(B_{kl}\Sigma_{kl} + B_{kl}\mu_k\mu_l),\\ &= \boldsymbol{A} : \boldsymbol{\Upsilon} : \boldsymbol{B} - (\boldsymbol{A} : \boldsymbol{\Sigma} + \boldsymbol{\mu}^\intercal\boldsymbol{A}\boldsymbol{\mu})(\boldsymbol{B} : \boldsymbol{\Sigma} + \boldsymbol{\mu}^\intercal\boldsymbol{B}\boldsymbol{\mu}).\end{aligned}$$

The final result is obtained noting that $\mathrm{Tr}(\boldsymbol{A}\boldsymbol{B}) = \boldsymbol{A} : \boldsymbol{B}$. □

Let's now assume normality. The following result may be found in Isserlis (1918):

Proposition A.6: Fourth order moments of centered multivariate normal distribution

Let $\boldsymbol{Z} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{\Sigma})$ be an n-dimensional multivariate normally distributed random vector whose expected value is zero and whose variance - covariance matrix is $\boldsymbol{\Sigma}$.

If $\boldsymbol{\mu} = \mathbb{M}_1(\boldsymbol{Z})$, $\boldsymbol{\Omega} = \mathbb{M}_2(\boldsymbol{Z})$, $\boldsymbol{\Lambda} = \mathbb{M}_3(\boldsymbol{Z})$ and $\boldsymbol{\Upsilon} = \mathbb{M}_4(\boldsymbol{Z})$ are the first, second, third and fourth order moments, then, $\forall i, j, k, l = 1, \ldots, n$:

$$\mu_i = 0, \tag{A.19}$$

$$\Omega_{ij} = \Sigma_{ij}, \tag{A.20}$$

$$\Lambda_{ijk} = 0, \tag{A.21}$$

and

$$\Upsilon_{ijkl} = \Sigma_{ij}\Sigma_{kl} + \Sigma_{ik}\Sigma_{jl} + \Sigma_{jk}\Sigma_{il}. \tag{A.22}$$

Lemma A.2: Moments of a SQF under normality

Let $\boldsymbol{Z} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{\Sigma})$ be an n-dimensional multivariate normally distributed random vector with expected value $\boldsymbol{\mu} = \boldsymbol{0}$ and variance - covariance matrix $\boldsymbol{\Sigma}$. Then, for symmetric matrices $\boldsymbol{A}$ and $\boldsymbol{B}$:

$$\mathbb{E}[Q_{\boldsymbol{A}}(\boldsymbol{Z})] = \mathrm{Tr}(\boldsymbol{A\Sigma}) \tag{A.23}$$

$$\mathbb{E}[Q_{\boldsymbol{A}}(\boldsymbol{Z})Q_{\boldsymbol{B}}(\boldsymbol{Z})] = \mathrm{Tr}(\boldsymbol{A\Sigma})\mathrm{Tr}(\boldsymbol{B\Sigma}) + 2\mathrm{Tr}(\boldsymbol{A\Sigma B\Sigma}) \tag{A.24}$$

$$\mathbb{E}[Q_{\boldsymbol{A}}(\boldsymbol{Z})\boldsymbol{Z}] = \boldsymbol{0} \tag{A.25}$$

Proof. The first equation is obtained directly from Proposition A.4 and the fact that $\boldsymbol{\mu} = \boldsymbol{0}$. For the second, note that, following Lemma A.1, $\mathbb{E}[Q_{\boldsymbol{A}}(\boldsymbol{Z})Q_{\boldsymbol{B}}(\boldsymbol{Z})] = A_{ij}B_{kl}\Upsilon_{ijkl}$, but, by virtue of Proposition A.6, $\Upsilon_{ijkl} = \Sigma_{ij}\Sigma_{kl} + \Sigma_{ik}\Sigma_{jl} + \Sigma_{jk}\Sigma_{il}$ and then, using $\boldsymbol{\Sigma}$ symmetry:

$$\begin{aligned}\mathbb{E}[Q_{\boldsymbol{A}}(\boldsymbol{Z})Q_{\boldsymbol{B}}(\boldsymbol{Z})] &= A_{ij}\Sigma_{ij}B_{kl}\Sigma_{kl} + A_{ij}\Sigma_{ki}B_{kl}\Sigma_{lj} + A_{ij}\Sigma_{jk}B_{kl}\Sigma_{li}, \\ &= \mathrm{Tr}(\boldsymbol{A\Sigma})\mathrm{Tr}(\boldsymbol{B\Sigma}) + 2\mathrm{Tr}(\boldsymbol{A\Sigma B\Sigma}).\end{aligned}$$

Finally, for the third one, following again Lemma A.1, $\mathbb{E}[Q_{\boldsymbol{A}}(\boldsymbol{Z})\boldsymbol{Z}] = A_{ij}\Lambda_{ijk}\boldsymbol{e}_k$, and then, using Proposition A.6 we obtain that $\mathbb{E}[Q_{\boldsymbol{A}}(\boldsymbol{Z})\boldsymbol{Z}] = \boldsymbol{0}$. □

We can now prove the following result:

Proposition A.7: Variance of a SQF and covariance of two SQF under normallity

Let $\boldsymbol{Z} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ be an n-dimensional multivariate normally distributed random vector with expected value $\boldsymbol{\mu}$ and variance - covariance matrix $\boldsymbol{\Sigma}$.
Then, if $\boldsymbol{A}$ and $\boldsymbol{B}$ are symmetric:

$$\mathrm{Cov}(Q_{\boldsymbol{A}}(\boldsymbol{X}), Q_{\boldsymbol{B}}(\boldsymbol{X})) = 2\mathrm{Tr}(\boldsymbol{A\Sigma B\Sigma}) + 4\boldsymbol{\mu}^\intercal\boldsymbol{A\Sigma B}\boldsymbol{\mu}. \tag{A.26}$$

In particular:

$$\mathrm{Var}(Q_{\boldsymbol{A}}(\boldsymbol{X})) = 2\mathrm{Tr}(\boldsymbol{A\Sigma A\Sigma}) + 4\boldsymbol{\mu}^\intercal\boldsymbol{A\Sigma A}\boldsymbol{\mu}. \tag{A.27}$$

Proof. We use the expression:

$$\mathrm{Cov}(X, Y) = \mathbb{E}[XY] - \mathbb{E}[X]\mathbb{E}[Y],$$

with $X = Q_{\boldsymbol{A}}(\boldsymbol{X})$ and $Y = Q_{\boldsymbol{B}}(\boldsymbol{X})$. The first term in the right-hand side may be developed in terms of $\boldsymbol{Z} = \boldsymbol{X} - \boldsymbol{\mu}$ using symmetry of matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ as:

$$\begin{aligned}
&\mathbb{E}[Q_{\boldsymbol{A}}(\boldsymbol{X})Q_{\boldsymbol{B}}(\boldsymbol{X})],\\
&= \mathbb{E}[(\boldsymbol{Z}^\intercal\boldsymbol{A}\boldsymbol{Z} + \boldsymbol{Z}^\intercal\boldsymbol{A}\boldsymbol{\mu} + \boldsymbol{\mu}^\intercal\boldsymbol{A}\boldsymbol{Z} + \boldsymbol{\mu}^\intercal\boldsymbol{A}\boldsymbol{\mu})(\boldsymbol{Z}^\intercal\boldsymbol{B}\boldsymbol{Z} + \boldsymbol{Z}^\intercal\boldsymbol{B}\boldsymbol{\mu} + \boldsymbol{\mu}^\intercal\boldsymbol{B}\boldsymbol{Z} + \boldsymbol{\mu}^\intercal\boldsymbol{B}\boldsymbol{\mu})],\\
&= \mathbb{E}[\boldsymbol{Z}^\intercal\boldsymbol{A}\boldsymbol{Z}\boldsymbol{Z}^\intercal\boldsymbol{B}\boldsymbol{Z}] + 2\boldsymbol{\mu}^\intercal\boldsymbol{B}\mathbb{E}[\boldsymbol{Z}^\intercal\boldsymbol{A}\boldsymbol{Z}\boldsymbol{Z}] + 2\boldsymbol{\mu}^\intercal\boldsymbol{A}\mathbb{E}[\boldsymbol{Z}^\intercal\boldsymbol{B}\boldsymbol{Z}\boldsymbol{Z}] + 4\boldsymbol{\mu}^\intercal\boldsymbol{A}\mathbb{E}[\boldsymbol{Z}\boldsymbol{Z}^\intercal]\boldsymbol{B}\boldsymbol{\mu}\\
&+ \mathbb{E}[\boldsymbol{Z}^\intercal\boldsymbol{A}\boldsymbol{Z}]\boldsymbol{\mu}^\intercal\boldsymbol{B}\boldsymbol{\mu} + \mathbb{E}[\boldsymbol{Z}^\intercal\boldsymbol{B}\boldsymbol{Z}]\boldsymbol{\mu}^\intercal\boldsymbol{A}\boldsymbol{\mu} + 2\boldsymbol{\mu}^\intercal\boldsymbol{A}\mathbb{E}[\boldsymbol{Z}] + 2\boldsymbol{\mu}^\intercal\boldsymbol{B}\mathbb{E}[\boldsymbol{Z}] + \boldsymbol{\mu}^\intercal\boldsymbol{A}\boldsymbol{\mu}\boldsymbol{\mu}^\intercal\boldsymbol{B}\boldsymbol{\mu},\\
&= \mathbb{E}[Q_{\boldsymbol{A}}(\boldsymbol{Z})Q_{\boldsymbol{B}}(\boldsymbol{Z})] + 2\boldsymbol{\mu}^\intercal\mathbb{E}[Q_{\boldsymbol{A}}(\boldsymbol{Z})\boldsymbol{Z}] + 2\boldsymbol{\mu}^\intercal\mathbb{E}[Q_{\boldsymbol{B}}(\boldsymbol{Z})\boldsymbol{Z}] + 4\boldsymbol{\mu}^\intercal\boldsymbol{A}\boldsymbol{\Sigma}\boldsymbol{B}\boldsymbol{\mu}\\
&+ \mathbb{E}[Q_{\boldsymbol{A}}(\boldsymbol{Z})]\boldsymbol{\mu}^\intercal\boldsymbol{B}\boldsymbol{\mu} + \mathbb{E}[Q_{\boldsymbol{B}}(\boldsymbol{Z})]\boldsymbol{\mu}^\intercal\boldsymbol{A}\boldsymbol{\mu} + \boldsymbol{\mu}^\intercal\boldsymbol{A}\boldsymbol{\mu}\boldsymbol{\mu}^\intercal\boldsymbol{B}\boldsymbol{\mu},\\
&= \mathrm{Tr}(\boldsymbol{A}\boldsymbol{\Sigma})\mathrm{Tr}(\boldsymbol{B}\boldsymbol{\Sigma}) + 2\mathrm{Tr}(\boldsymbol{A}\boldsymbol{\Sigma}\boldsymbol{B}\boldsymbol{\Sigma}) + \mathrm{Tr}(\boldsymbol{A}\boldsymbol{\Sigma})\boldsymbol{\mu}^\intercal\boldsymbol{B}\boldsymbol{\mu} + \mathrm{Tr}(\boldsymbol{B}\boldsymbol{\Sigma})\boldsymbol{\mu}^\intercal\boldsymbol{A}\boldsymbol{\mu}\\
&+ 4\boldsymbol{\mu}^\intercal\boldsymbol{A}\boldsymbol{\Sigma}\boldsymbol{B}\boldsymbol{\mu} + \boldsymbol{\mu}^\intercal\boldsymbol{A}\boldsymbol{\mu}\boldsymbol{\mu}^\intercal\boldsymbol{B}\boldsymbol{\mu},
\end{aligned}$$

where, in the last equality, we have used Lemma A.2. The second term in the right-hand side is obtained analogously:

$$\begin{aligned}
&\mathbb{E}[Q_{\boldsymbol{A}}(\boldsymbol{X})]\mathbb{E}[Q_{\boldsymbol{B}}(\boldsymbol{X})],\\
&= \mathbb{E}[\boldsymbol{Z}^\intercal\boldsymbol{A}\boldsymbol{Z} + \boldsymbol{Z}^\intercal\boldsymbol{A}\boldsymbol{\mu} + \boldsymbol{\mu}^\intercal\boldsymbol{A}\boldsymbol{Z} + \boldsymbol{\mu}^\intercal\boldsymbol{A}\boldsymbol{\mu}]\mathbb{E}[\boldsymbol{Z}^\intercal\boldsymbol{B}\boldsymbol{Z} + \boldsymbol{Z}^\intercal\boldsymbol{B}\boldsymbol{\mu} + \boldsymbol{\mu}^\intercal\boldsymbol{B}\boldsymbol{Z} + \boldsymbol{\mu}^\intercal\boldsymbol{B}\boldsymbol{\mu}],\\
&= (\mathbb{E}[\boldsymbol{Z}^\intercal\boldsymbol{A}\boldsymbol{Z}] + 2\boldsymbol{\mu}^\intercal\boldsymbol{A}\mathbb{E}[\boldsymbol{Z}] + \boldsymbol{\mu}^\intercal\boldsymbol{A}\boldsymbol{\mu})(\mathbb{E}[\boldsymbol{Z}^\intercal\boldsymbol{B}\boldsymbol{Z}] + 2\boldsymbol{\mu}^\intercal\boldsymbol{B}\mathbb{E}[\boldsymbol{Z}] + \boldsymbol{\mu}^\intercal\boldsymbol{B}\boldsymbol{\mu}),\\
&= (\mathbb{E}[Q_{\boldsymbol{A}}(\boldsymbol{Z})] + \boldsymbol{\mu}^\intercal\boldsymbol{A}\boldsymbol{\mu})(\mathbb{E}[Q_{\boldsymbol{B}}(\boldsymbol{Z})] + \boldsymbol{\mu}^\intercal\boldsymbol{B}\boldsymbol{\mu}),\\
&= (\mathrm{Tr}(\boldsymbol{A}\boldsymbol{\Sigma}) + \boldsymbol{\mu}^\intercal\boldsymbol{A}\boldsymbol{\mu})(\mathrm{Tr}(\boldsymbol{B}\boldsymbol{\Sigma}) + \boldsymbol{\mu}^\intercal\boldsymbol{B}\boldsymbol{\mu}),\\
&= \mathrm{Tr}(\boldsymbol{A}\boldsymbol{\Sigma})\mathrm{Tr}(\boldsymbol{B}\boldsymbol{\Sigma}) + \boldsymbol{\mu}^\intercal\boldsymbol{A}\boldsymbol{\mu}\boldsymbol{\mu}^\intercal\boldsymbol{B}\boldsymbol{\mu} + \mathrm{Tr}(\boldsymbol{A}\boldsymbol{\Sigma})\boldsymbol{\mu}^\intercal\boldsymbol{B}\boldsymbol{\mu} + \boldsymbol{\mu}^\intercal\boldsymbol{A}\boldsymbol{\mu}\mathrm{Tr}(\boldsymbol{B}\boldsymbol{\Sigma}).
\end{aligned}$$

Again, in the last equality, we have used Lemma A.2. Therefore, we have:

$$\mathbb{E}[Q_{\boldsymbol{A}}(\boldsymbol{X})Q_{\boldsymbol{B}}(\boldsymbol{X})] - \mathbb{E}[Q_{\boldsymbol{A}}(\boldsymbol{X})]\mathbb{E}[Q_{\boldsymbol{B}}(\boldsymbol{X})] = 2\mathrm{Tr}(\boldsymbol{A}\boldsymbol{\Sigma}\boldsymbol{B}\boldsymbol{\Sigma}) + 4\boldsymbol{\mu}^\intercal\boldsymbol{A}\boldsymbol{\Sigma}\boldsymbol{B}\boldsymbol{\mu}.$$

□

A.2.2 Stochastic distances

Proposition A.8: Expected value of the squared distance

Let $\mathcal{V}$ be a linear manifold and let $\pi_{\mathcal{V}}$ be the orthogonal projection on $\mathcal{V}$, $\boldsymbol{p} \in \mathbb{R}^n$, $\boldsymbol{X} \in \mathcal{V}$ a random vector and $D = d(\boldsymbol{p}, \boldsymbol{X})$. Let us suppose that $\boldsymbol{\mu} = \mathbb{E}[\boldsymbol{X}]$ and $\boldsymbol{\Sigma} = \mathbf{Cov}(\boldsymbol{X})$ are finite. Then:

$$\mathbb{E}[D^2] = \|\boldsymbol{p} - \pi_{\mathcal{V}}(\boldsymbol{p})||^2 + ||\pi_{\mathcal{V}}(\boldsymbol{p}) - \boldsymbol{\mu}\|^2 + \mathrm{Tr}(\boldsymbol{\Sigma}). \tag{A.28}$$

Proof. Using Pythagoras theorem we have:

$$\begin{aligned}
D^2 &= \|\boldsymbol{p} - \boldsymbol{X}\|^2,\\
&= \|\boldsymbol{p} - \pi_{\mathcal{V}}(\boldsymbol{p})\|^2 + \|\pi_{\mathcal{V}}(\boldsymbol{p}) - \boldsymbol{X}\|^2.
\end{aligned} \tag{A.29}$$

Therefore:

$$\begin{aligned}\mathbb{E}[D^2] &= \mathbb{E}\left[\|\boldsymbol{p}-\pi_{\mathcal{V}}(\boldsymbol{p})\|^2+\|\pi_{\mathcal{V}}(\boldsymbol{p})-\boldsymbol{X}\|^2\right], \\ &= \|\boldsymbol{p}-\pi_{\mathcal{V}}(\boldsymbol{p})\|^2+\mathbb{E}\left[\|\pi_{\mathcal{V}}(\boldsymbol{p})-\boldsymbol{X}\|^2\right]. \end{aligned} \tag{A.30}$$

If we define the random vector $\boldsymbol{X}' = \pi_{\mathcal{V}}(\boldsymbol{p}) - \boldsymbol{X}$, whose expected value is $\mathbb{E}[\boldsymbol{X}'] = \pi_{\mathcal{V}}(\boldsymbol{p}) - \boldsymbol{\mu}$ and whose variance - covariance matrix is $\mathbf{Cov}(\boldsymbol{X}') = \boldsymbol{\Sigma}$, then it is possible to define an SQF $Q_{\boldsymbol{I}}(\boldsymbol{X}') = \|\pi_{\mathcal{V}}(\boldsymbol{p}) - \boldsymbol{X}\|^2$. Then, using Proposition A.4 we conclude:

$$\mathbb{E}\left[\|\pi_{\mathcal{V}}(\boldsymbol{p})-\boldsymbol{X}\|^2\right] = \mathrm{Tr}(\boldsymbol{\Sigma}) + ||\pi_{\mathcal{V}}(\boldsymbol{p})-\boldsymbol{\mu}||^2. \tag{A.31}$$

Combining Eqs. (A.30) and (A.31) we obtain the result.

□

An analogous result for the variance of D^2 can be derived.

Proposition A.9: Variance of the squared distance

Let $\mathcal{V}$ be a linear manifold and let $\pi_{\mathcal{V}}$ the orthogonal projection on $\mathcal{V}$, $\boldsymbol{p} \in \mathbb{R}^n$, $\boldsymbol{X} \in \mathcal{V}$ a random vector and $D = d(\boldsymbol{p}, \boldsymbol{X})$. Let us suppose that $\boldsymbol{\mu} = \mathbb{E}[\boldsymbol{X}]$, $\boldsymbol{\Sigma} = \mathbf{Cov}(\boldsymbol{X})$ and $\boldsymbol{\Upsilon} = \mathbb{M}_4(\boldsymbol{X} - \pi_{\mathcal{V}}(\boldsymbol{p}))$ the $\pi_{\mathcal{V}}(\boldsymbol{p})$-centered fourth order moment tensor of $\boldsymbol{X}$ are finite. Then:

$$\mathrm{Var}(D^2) = \boldsymbol{I} : \boldsymbol{\Upsilon} : \boldsymbol{I} - \left[\mathrm{Tr}(\boldsymbol{\Sigma}) + \|\pi_{\mathcal{V}}(\boldsymbol{p})-\boldsymbol{\mu}\|^2\right]^2. \tag{A.32}$$

Proof. We have seen in the previous proof that:

$$D^2 = \|\boldsymbol{p}-\pi_{\mathcal{V}}(\boldsymbol{p})\|^2+\|\pi_{\mathcal{V}}(\boldsymbol{p})-\boldsymbol{X}\|^2. \tag{A.33}$$

Then:

$$\begin{aligned}\mathrm{Var}(D^2) &= \mathrm{Var}\left(\|\boldsymbol{p}-\pi_{\mathcal{V}}(\boldsymbol{p})\|^2+\|\pi_{\mathcal{V}}(\boldsymbol{p})-\boldsymbol{X}\|^2\right), \\ &= \mathrm{Var}\left(\|\pi_{\mathcal{V}}(\boldsymbol{p})-\boldsymbol{X}\|^2\right). \end{aligned} \tag{A.34}$$

If we define the random vector $\boldsymbol{X}' = \pi_{\mathcal{V}}(\boldsymbol{p}) - \boldsymbol{X}$, whose expected value is $\mathbb{E}[\boldsymbol{X}'] = \pi_{\mathcal{V}}(\boldsymbol{p}) - \boldsymbol{\mu}$, whose variance - covariance matrix is $\mathrm{Cov}(\boldsymbol{X}') = \boldsymbol{\Sigma}$ and whose fourth order moment tensor is $\boldsymbol{\Upsilon}$, then, it is possible to define an SQF $Q_{\boldsymbol{I}}(\boldsymbol{X}') = \|\pi_{\mathcal{V}}(\boldsymbol{p}) - \boldsymbol{X}\|^2$. Then, using Proposition A.5 we conclude:

$$\mathrm{Var}\left(\|\pi_{\mathcal{V}}(\boldsymbol{p})-\boldsymbol{X}\|^2\right) = \boldsymbol{I} : \boldsymbol{\Upsilon} : \boldsymbol{I} - \left(\mathrm{Tr}(\boldsymbol{\Sigma}) + \|\pi_{\mathcal{V}}(\boldsymbol{p})-\boldsymbol{\mu}\|^2\right)^2. \tag{A.35}$$

Combining Eqs. (A.34) and Eqs. (A.35) we obtain the result.

□

Under normality conditions, we have the following result:

Proposition A.10: Variance of the squared distance under normality

Let $\mathcal{V}$ be a linear manifold and let $\pi_{\mathcal{V}}$ the orthogonal projection on $\mathcal{V}$, $\boldsymbol{p} \in \mathbb{R}^n$, $\boldsymbol{X} \in \mathcal{V}$ a random vector and $D = d(\boldsymbol{p}, \boldsymbol{X})$. Let us suppose that $\boldsymbol{X} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ is a d-dimensional multivariate distributed random vector with expected value $\boldsymbol{\mu}$ and variance - covariance matrix $\boldsymbol{\Sigma}$. Then:

$$\mathrm{Var}(D^2) = 2\mathrm{Tr}(\boldsymbol{\Sigma}^2) + 4\left(\pi_{\mathcal{V}}(\boldsymbol{p}) - \boldsymbol{\mu}\right)^\intercal \boldsymbol{\Sigma} \left(\pi_{\mathcal{V}}(\boldsymbol{p}) - \boldsymbol{\mu}\right). \tag{A.36}$$

Proof. We have seen in the previous proof that:

$$\mathrm{Var}(D^2) = \mathrm{Var}\left(\|\pi_{\mathcal{V}}(\boldsymbol{p}) - \boldsymbol{X}\|^2\right). \tag{A.37}$$

If we define the random vector $\boldsymbol{X}' = \pi_{\mathcal{V}}(\boldsymbol{p}) - \boldsymbol{X}$, then $\boldsymbol{X}' \sim \mathcal{N}(\pi_{\mathcal{V}}(\boldsymbol{p}) - \boldsymbol{\mu}, \boldsymbol{\Sigma})$ being then possible to define an SQF $Q_{\boldsymbol{I}}(\boldsymbol{X}') = \|\pi_{\mathcal{V}}(\boldsymbol{p}) - \boldsymbol{X}\|^2$. Therefore, using Proposition A.7 we conclude:

$$\mathrm{Var}\left(\|\pi_{\mathcal{V}}(\boldsymbol{p}) - \boldsymbol{X}\|^2\right) = 2\mathrm{Tr}(\boldsymbol{\Sigma}^2) + 4\left(\pi_{\mathcal{V}}(\boldsymbol{p}) - \boldsymbol{\mu}\right)^\intercal \boldsymbol{\Sigma} \left(\pi_{\mathcal{V}}(\boldsymbol{p}) - \boldsymbol{\mu}\right). \tag{A.38}$$

Combining Eqs. (A.37) and (A.38) we obtain the result.

□

Proposition A.11: Squared distance distributional properties

Let $\mathcal{V}$ be a linear manifold and let $\pi_{\mathcal{V}}$ be the orthogonal projection on $\mathcal{V}$, $\boldsymbol{p} \in \mathbb{R}^n$, $\boldsymbol{X} \in \mathcal{V}$ a random vector and $D = d(\boldsymbol{p}, \boldsymbol{X})$. Let us assume that $\boldsymbol{X}$ follows a multivariate normal distribution $\boldsymbol{X} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$. Let $\chi^2 = (\boldsymbol{p} - \boldsymbol{X})^\intercal (\boldsymbol{\Sigma})^{-1} (\boldsymbol{p} - \boldsymbol{X})$. Then χ^2 follows a non-central χ^2 distribution with $k = \dim(\mathcal{V})$ degrees of freedom and non-centrality parameter $\lambda = (\pi_{\mathcal{V}}(\boldsymbol{p}) - \boldsymbol{\mu})^\intercal (\boldsymbol{\Sigma})^{-1} (\pi_{\mathcal{V}}(\boldsymbol{p}) - \boldsymbol{\mu})$.

Proof. Let $\boldsymbol{X}' = \boldsymbol{p} - \boldsymbol{X}$. $\boldsymbol{X}'$ follow a multivariate normal distribution $\boldsymbol{X}' \sim \mathcal{N}(\boldsymbol{p} - \boldsymbol{\mu}, \boldsymbol{\Sigma})$. Then, $\boldsymbol{\Sigma}^{-1/2}\boldsymbol{X}' \sim \mathcal{N}(\boldsymbol{\Sigma}^{-1/2}(\boldsymbol{p} - \boldsymbol{\mu}), \boldsymbol{I})$. Using the non-central χ^2 distribution definition:

$$\chi^2 = \boldsymbol{X}'^\intercal \boldsymbol{\Sigma}^{-1} \boldsymbol{X}' = (\boldsymbol{\Sigma}^{-1/2}\boldsymbol{X}')^\intercal (\boldsymbol{\Sigma}^{-1/2}\boldsymbol{X}') \sim \chi^2(k, \lambda), \tag{A.39}$$

where $k = \mathrm{rang}(\boldsymbol{\Sigma}) = \dim(\mathcal{V})$ and $\lambda = (\boldsymbol{\Sigma}^{-1/2}(\boldsymbol{p} - \boldsymbol{\mu}))^\intercal (\boldsymbol{\Sigma}^{-1/2}(\boldsymbol{p} - \boldsymbol{\mu})) = (\boldsymbol{p} - \boldsymbol{\mu})^\intercal \boldsymbol{\Sigma}^{-1} (\boldsymbol{p} - \boldsymbol{\mu})$. But we have, $\boldsymbol{p} - \boldsymbol{\mu} = (\boldsymbol{p} - \pi_{\mathcal{V}}(\boldsymbol{p})) + (\pi_{\mathcal{V}}(\boldsymbol{p})) - \boldsymbol{\mu})$. As $\boldsymbol{p} - \pi_{\mathcal{V}}(\boldsymbol{p}) \in \ker(\boldsymbol{\Sigma}^{-1})$, and then $(\boldsymbol{p} - \boldsymbol{\mu})^\intercal \boldsymbol{\Sigma}^{-1} (\boldsymbol{p} - \boldsymbol{\mu}) = (\pi_{\mathcal{V}}(\boldsymbol{p}) - \boldsymbol{\mu})^\intercal \boldsymbol{\Sigma}^{-1} (\pi_{\mathcal{V}}(\boldsymbol{p}) - \boldsymbol{\mu})$. □

B. Finite Element implementation of cell culture evolution in microfluidic devices

B.1 3D Finite Element implementation

Our starting point is the system of partial differential equations (PDEs) given by Eqs. (6.1) and (6.4), defined in Chapter 6, which we reproduce here.

$$\frac{\partial C_i}{\partial t} + (\boldsymbol{v}\cdot\boldsymbol{\nabla})C_i + \boldsymbol{\nabla}\cdot\boldsymbol{q}_i = C_i F_i - \sum_{\substack{j=1\\ j\neq i}}^{n} C_i F_{ij} + \sum_{\substack{j=1\\ j\neq i}}^{n} C_j F_{ji}, \quad i = 1,\dots,n, \tag{B.1a}$$

$$\frac{\partial C'_i}{\partial t} + (\boldsymbol{v}\cdot\boldsymbol{\nabla})C'_i + \boldsymbol{\nabla}\cdot\boldsymbol{q}'_i = \sum_{j=1}^{n} C_j F'_{ij}, \quad i = 1,\dots,m, \tag{B.1b}$$

where, from Eqs. (6.17) and (6.30), we have:

$$\boldsymbol{q}_i = \boldsymbol{q}_{\mathrm{D},i} + \boldsymbol{q}_{\mathrm{M},i} + \boldsymbol{q}_{\mathrm{E},i} + \boldsymbol{q}_{\mathrm{T},i} + \boldsymbol{q}_{\mathrm{C},i}, \quad i = 1,\dots,n, \tag{B.2a}$$

$$\boldsymbol{q}'_i = \boldsymbol{q}'_{\mathrm{D},i}, \quad i = 1,\dots,m. \tag{B.2b}$$

where, from Eqs. (6.18), (6.20), (6.22), (6.24) and (6.26):

$$\boldsymbol{q}_{\mathrm{D},i} = -\boldsymbol{K}_{\mathrm{D},i}\boldsymbol{\nabla}C_i, \quad i = 1,\dots,n, \tag{B.3a}$$

$$\boldsymbol{q}_{\mathrm{M},i} = \sum_{j=1}^{k} \boldsymbol{K}_{\mathrm{M},i,j}\boldsymbol{\nabla}p_j, \quad i = 1,\dots,n, \tag{B.3b}$$

$$\boldsymbol{q}_{\mathrm{E},i} = \boldsymbol{K}_{\mathrm{E},i}\boldsymbol{\nabla}V, \quad i = 1,\dots,n, \tag{B.3c}$$

$$\boldsymbol{q}_{\mathrm{T},i} = \boldsymbol{K}_{\mathrm{T},i}\boldsymbol{\nabla}\theta, \quad i = 1,\dots,n, \tag{B.3d}$$

$$\boldsymbol{q}_{\mathrm{C},i} = \sum_{j=1}^{m} \boldsymbol{K}_{\mathrm{D},i,j}\boldsymbol{\nabla}C'_j, \quad i = 1,\dots,n, \tag{B.3e}$$

$$\boldsymbol{q}'_i = -\boldsymbol{K}'_{\mathrm{D},i}\boldsymbol{\nabla}C'_i, \quad i = 1,\dots,m. \tag{B.3f}$$

In order to abbreviate the notations, let us define:

$$\boldsymbol{A}_i = C_i\boldsymbol{v} + \boldsymbol{q}_i, \quad i = 1, \dots, n, \tag{B.4a}$$
$$\boldsymbol{A}'_i = C'_i\boldsymbol{v} + \boldsymbol{q}'_i, \quad i = 1, \dots, m, \tag{B.4b}$$
$$a_i = C_iF_i + \sum_{\substack{j=1 \\ j\neq i}}^{n}(C_jF_{ji} - C_iF_{ij}) \quad i = 1, \dots, m, \tag{B.4c}$$
$$a'_i = \sum_{j=1}^{m} C_j(F'_{ji} - F'_{ij}), \quad i = 1, \dots, m. \tag{B.4d}$$

Assuming incompressible flow ($\boldsymbol{\nabla} \cdot \boldsymbol{v} = 0$), we obtain:

$$\boldsymbol{\nabla} \cdot \boldsymbol{A}_i = \boldsymbol{\nabla}C_i \cdot \boldsymbol{v}C_i\boldsymbol{\nabla} + \boldsymbol{v} + \boldsymbol{\nabla} \cdot \boldsymbol{q}_i = \boldsymbol{v} \cdot \boldsymbol{\nabla}C_i + \boldsymbol{\nabla} \cdot \boldsymbol{q}_i, \quad i = 1, \dots, n, \tag{B.5a}$$
$$\boldsymbol{\nabla} \cdot \boldsymbol{A}'_i = \boldsymbol{\nabla}C'_i \cdot \boldsymbol{v}C'_i\boldsymbol{\nabla} + \boldsymbol{v} + \boldsymbol{\nabla} \cdot \boldsymbol{q}'_i = \boldsymbol{v} \cdot \boldsymbol{\nabla}C'_i + \boldsymbol{\nabla} \cdot \boldsymbol{q}'_i, \quad i = 1, \dots, m. \tag{B.5b}$$

Therefore, Eqs. (B.1) writes:

$$\frac{\partial C_i}{\partial t} + \boldsymbol{\nabla} \cdot \boldsymbol{A}_i = a_i, \quad i = 1, \dots, n, \tag{B.6a}$$
$$\frac{\partial C'_i}{\partial t} + \boldsymbol{\nabla} \cdot \boldsymbol{A}'_i = a'_i, \quad i = 1, \dots, m. \tag{B.6b}$$

The boundary conditions associated with these equations are Eqs. (6.2), (6.3), (6.5) and (6.5), that are:

$$C_i = f_i, \quad i = 1, \dots, n, \quad \text{in } \Gamma_{D_i}, \tag{B.7a}$$
$$C'_i = f'_i, \quad i = 1, \dots, m, \quad \text{in } \Gamma_{D'_i}, \tag{B.7b}$$

$$\kappa_iC_i + \frac{\partial C_i}{\partial \boldsymbol{n}} = g_i, \quad i = 1, \dots, n, \quad \text{in } \Gamma_{R_i}, \tag{B.8a}$$
$$\kappa'_iC'_i + \frac{\partial C_i}{\partial \boldsymbol{n}} = g'_i, \quad i = 1, \dots, m, \quad \text{in } \Gamma_{R'_i}. \tag{B.8b}$$

B.1.1 Weak form

Evolution equation

Now, the weak form of Eqs. (B.1) is derived. Let $\phi \in \mathcal{H}^1_0(\Omega)$, being $\mathcal{H}^1_0(\Omega)$ the closure in $\mathcal{H}^1(\Omega)$ ($\mathcal{H}^1(\Omega) = W^{1,2}(\Omega)$, the Sobolev space with respect to L^2 norm of differentiable functions (in the weak sense) of order 1 of infinitely differentiable functions compactly supported in Ω), or, equivalently, the space of functions in $\mathcal{H}^1(\Omega)$ that vanish at the boundary $\partial\Omega$. Eqs. (B.6) can be multiplied by a test function $\phi \in \mathcal{H}^1_0(\Omega)$ and integrated in Ω to obtain:

$$\int_\Omega \phi\frac{\partial C_i}{\partial t} + \phi\boldsymbol{\nabla} \cdot \boldsymbol{A}_i \, d\Omega = \int_\Omega a_i\phi \, d\Omega, \quad i = 1, \dots, n. \tag{B.9}$$

Integrating by parts Eq. (B.9), splitting the boundary $\partial\Omega = \Gamma_{D_i} \cup \Gamma_{R_i}$ where respectively Dirichlet or Robin-Neumann boundary conditions are applied, and using that ϕ vanishes at the boundary, the equation results in:

$$\int_\Omega \left(\phi \frac{\partial C_i}{\partial t} - \boldsymbol{\nabla}\phi \cdot \boldsymbol{A}_i \right) \, d\Omega = \int_\Omega \phi a_i \, d\Omega - \int_{\Gamma_{R_i}} \phi \frac{\partial C_i}{\partial \boldsymbol{n}} \, d\Gamma, \quad i = 1, \ldots, n. \tag{B.10}$$

We obtain an analogous expression for the chemical species:

$$\int_\Omega \left(\phi \frac{\partial C_i'}{\partial t} - \boldsymbol{\nabla}\phi \cdot \boldsymbol{A}_i' \right) \, d\Omega = \int_\Omega \phi a_i' \, d\Omega - \int_{\Gamma_{R_i'}} \phi \frac{\partial C_i'}{\partial \boldsymbol{n}} \, d\Gamma, \quad i = 1, \ldots, m. \tag{B.11}$$

Boundary conditions

Multiplying Eq. (B.7) by a test function ϕ and integrating in Γ_{D_i}, we arrive to

$$\int_{\Gamma_{D_i}} \phi C_i \, d\Gamma = \int_{\Gamma_{D_i}} \phi f_i \, d\Gamma, \quad i = 1, \ldots, n. \tag{B.12}$$

Repeating the process with Eq. (B.8) and integrating in Γ_{R_i}, we obtain:

$$\int_{\Gamma_{R_i}} \kappa_i \phi C_i \, d\Gamma + \int_{\Gamma_{R_i}} \phi \frac{\partial C_i}{\partial \boldsymbol{n}} \, d\Gamma = \int_{\Gamma_{R_i}} \phi g_i \, d\Gamma, \quad i = 1, \ldots, n. \tag{B.13}$$

The analogous expressions for the species equations are:

$$\int_{\Gamma_{D_i}} \phi C_i' \, d\Gamma = \int_{\Gamma_{D_i'}} \phi f_i' \, d\Gamma, \quad i = 1, \ldots, m, \tag{B.14}$$

$$\int_{\Gamma_{R_i'}} \kappa_i \phi C_i' \, d\Gamma + \int_{\Gamma_{R_i'}} \phi \frac{\partial C_i'}{\partial \boldsymbol{n}} \, d\Gamma = \int_{\Gamma_{R_i'}} \phi g_i' \, d\Gamma, \quad i = 1, \ldots, m. \tag{B.15}$$

B.1.2 Spatial discretisation

All scalar and vectorial fields involved in the problem are discretised using a finite basis of dimension N, $\mathcal{B} = \{\phi_r, r = 1, \ldots, N\}$, that is:

$$C_i(\boldsymbol{x}, t) = \sum_{r=1}^{N} C_i^r(t)\phi_r(\boldsymbol{x}), \quad i = 1, \ldots, n. \tag{B.16a}$$

$$C_i'(\boldsymbol{x}, t) = \sum_{r=1}^{N} C_i'^r(t)\phi_r(\boldsymbol{x}), \quad i = 1, \ldots, m, \tag{B.16b}$$

$$\theta(\boldsymbol{x}, t) = \sum_{r=1}^{N} \theta^r(t)\phi_r(\boldsymbol{x}), \tag{B.16c}$$

$$V(\boldsymbol{x}, t) = \sum_{r=1}^{N} V^r(t)\phi_r(\boldsymbol{x}), \tag{B.16d}$$

$$p_k(\boldsymbol{x}, t) = \sum_{r=1}^{N} p_k^r(t)\phi_r(\boldsymbol{x}), \quad i = 1, \ldots, k, \tag{B.16e}$$

$$\boldsymbol{v} = \sum_{r=1}^{N} \boldsymbol{v}^r(t)\phi_r(\boldsymbol{x}), \tag{B.16f}$$

Note that all fields are approximated using the same basis and a separation representation in space-time is postulated. Even if it is not the most general approach, the fact that Eqs. (B.10) involve up to first order derivatives is consistent with this finite elements (FE) approximation.

B.1.2.1 Cell populations.

Evolution equations

Plugging Eq. (B.16) into Eq. (B.10), we obtain:

$$\int_\Omega \left(\phi \sum_{r=1}^{N} \frac{\partial C_i^r}{\partial t} \phi_r \, d\Omega - \int_\Omega \boldsymbol{\nabla}\phi \cdot \boldsymbol{A}_i \right) \, d\Omega = \int_\Omega \phi a_i \, d\Omega - \int_{\Gamma_{R_i}} \phi \frac{\partial C_i}{\partial \boldsymbol{n}} \, d\Gamma, \;\; i = 1, \ldots, n. \tag{B.17}$$

Using linear properties of $\boldsymbol{\nabla}$ operator, Eqs. (B.3) may be written as:

$$\boldsymbol{q}_{\mathrm{D},i} = -\sum_{r=1}^{N} \left(C_i^r \boldsymbol{K}_{\mathrm{D},i} \right) \boldsymbol{\nabla}\phi_r, \quad i = 1, \ldots, n, \tag{B.18a}$$

$$\boldsymbol{q}_{\mathrm{M},i} = \sum_{r=1}^{N} \left(\sum_{j=1}^{k} p_j^r \boldsymbol{K}_{\mathrm{M},i,j} \right) \boldsymbol{\nabla}\phi_r, \quad i = 1, \ldots, n, \tag{B.18b}$$

$$\boldsymbol{q}_{\mathrm{E},i} = \sum_{r=1}^{N} \left(\theta^r \boldsymbol{K}_{\mathrm{E},i} \right) \boldsymbol{\nabla}\phi_r, \quad i = 1, \ldots, n, \tag{B.18c}$$

$$\boldsymbol{q}_{\mathrm{T},i} = \sum_{r=1}^{N} \left(V^r \boldsymbol{K}_{\mathrm{T},i} \right) \boldsymbol{\nabla}\phi_r, \quad i = 1, \ldots, n, \tag{B.18d}$$

$$\boldsymbol{q}_{\mathrm{C},i} = \sum_{r=1}^{N} \left(\sum_{j=1}^{n} C_j'^r \boldsymbol{K}_{\mathrm{C},i,j} \right) \boldsymbol{\nabla}\phi_r, \quad i = 1, \ldots, n. \tag{B.18e}$$

So, as $\boldsymbol{A}_i = C_i \boldsymbol{v} + \boldsymbol{q}_i$, we obtain:

$$\begin{aligned} \boldsymbol{A}_i = & \left(\sum_{r=1}^{N} C_i^r \phi_r \right) \boldsymbol{v} \\ & + \sum_{r=1}^{N} \left(-C_i^r \boldsymbol{K}_{\mathrm{D},i} + \sum_{j=1}^{k} p_j^r \boldsymbol{K}_{\mathrm{M},i,j} + \sum_{j=1}^{m} C_j'^r \boldsymbol{K}_{\mathrm{C},i,j} + \theta^r \boldsymbol{K}_{\mathrm{E},i} + V^r \boldsymbol{K}_{\mathrm{T},i} \right) \boldsymbol{\nabla}\phi_r, \end{aligned} \tag{B.19a}$$

and $\boldsymbol{\nabla}\phi \cdot \boldsymbol{A}_i$ writes:

$$\begin{aligned} \boldsymbol{\nabla}\phi \cdot \boldsymbol{A}_i = & \sum_{r=1}^{N} C_i^r \left(\boldsymbol{\nabla}\phi_i \cdot \phi_r \boldsymbol{v} - \boldsymbol{\nabla}\phi_i \cdot \left(\boldsymbol{K}_{D,i} \boldsymbol{\nabla}\phi_r \right) \right) \\ & + \sum_{r=1}^{N} \boldsymbol{\nabla}\phi \cdot \left(\sum_{j=1}^{k} p_j^r \boldsymbol{K}_{m,i,j} + \sum_{j=1}^{m} C_j'^r \boldsymbol{K}_{s,i,j} + V^r \boldsymbol{K}_{i,E} + \theta^r \boldsymbol{K}_{i,T} \right) \boldsymbol{\nabla}\phi_r. \end{aligned} \tag{B.20a}$$

Besides, as $C_i^r = C_i^r(t)$, $\phi\frac{\partial C_i}{\partial t}$ writes:

$$\phi\frac{\partial C_i^r}{\partial t} = \phi\sum_{r=1}^{N}\dot{C}_i^r\phi_r, \quad i = 1,\dots,n. \tag{B.21}$$

choosing $\phi = \phi_s, s = 1,\dots,N$ (Galerkin method) and reorganising terms in Eq. (B.20), we may write:

$$\boldsymbol{\nabla}\phi_s\cdot\boldsymbol{A}_i = \sum_{r=1}^{N}\hat{\mathbf{K}}_{sr}^{(i)}C_i^r + \sum_{j=1}^{m}\sum_{r=1}^{N}\hat{\mathbf{K}}_{sr}^{(ij)}C_j'^r + \hat{\boldsymbol{F}}_s^{(i)}, \quad i = 1,\dots,n. \tag{B.22}$$

with:

$$\hat{\mathbf{K}}_{sr}^{(i)} = \left(\boldsymbol{\nabla}\phi_s\cdot(\phi_r\boldsymbol{v}) - \boldsymbol{\nabla}\phi_s\cdot(\boldsymbol{K}_{\mathrm{D},i}\boldsymbol{\nabla}\phi_r)\right), \quad i = 1,\dots,n, \tag{B.23a}$$

$$\hat{\mathbf{K}}^{(ij)} = \boldsymbol{\nabla}\phi_s\cdot\boldsymbol{K}_{\mathrm{C},i,j}\boldsymbol{\nabla}\phi_r, \quad i = 1,\dots,n, \tag{B.23b}$$

$$\hat{\mathbf{F}}_s^{(i)} = \boldsymbol{\nabla}\phi_s\cdot\left[\sum_{r=1}^{N}\left((\sum_{j=1}^{k}p_j^r)\boldsymbol{K}_{\mathrm{M},i,j} + V^r\boldsymbol{K}_{\mathrm{E},i} + \theta^r\boldsymbol{K}_{\mathrm{T},i}\right)\boldsymbol{\nabla}\phi_r\right], \quad i = 1,\dots,n, \tag{B.23c}$$

and Eq. (B.21) becomes:

$$\phi_s\frac{\partial C_i}{\partial t} = \sum_{r=1}^{N}\hat{\mathbf{M}}_{sr}C_i^r, \quad i = 1,\dots,n, \tag{B.24}$$

where we have defined:

$$\hat{\mathbf{M}}_{sr}^{(i)} = \phi_s\phi_r. \tag{B.25}$$

Boundary conditions

Finally, Eq. (B.12), with $\phi = \phi_s$, $C_i = \sum_{r=1}^{N}C_i^r\phi_r$ and $f_i = \sum_{r=1}^{N}f_i^r\phi_r$, becomes:

$$C_i^r = f_i^r, \quad i = 1,\dots,n. \tag{B.26}$$

and Eq. (B.13), with $\phi = \phi_s$ and $C_i = \sum_{r=1}^{N}C_i^r\phi_r$:

$$\sum_{r=1}^{N}\left(\int_{\Gamma_{R_i}}\kappa_i\phi_s\phi_r\,d\Gamma\right)C_i^r + \int_{\Gamma_{R_i}}\phi_s\frac{\partial C_i}{\partial\boldsymbol{n}}\,d\Gamma = \int_{\Gamma_{R_i}}\phi_s g_i\,d\Gamma, \quad i = 1,\dots,n. \tag{B.27}$$

B.1.2.2 Chemical species

Evolution equation

In the same way as above, plugging now Eq. (B.16) into the analogue of Eq. (B.10) for species, we obtain:

$$\int_\Omega\left(\phi\sum_{r=1}^{N}\frac{\partial C_i'^r}{\partial t}\phi_r\,d\Omega - \int_\Omega\boldsymbol{\nabla}\phi\cdot\boldsymbol{A}_i'\right)d\Omega = \int_\Omega\phi a_i'\,d\Omega - \int_{\Gamma_{R_i'}}\phi\frac{\partial C_i}{\partial\boldsymbol{n}}\,d\Gamma,\ i = 1,\dots,m. \tag{B.28}$$

Using again the linear properties of $\boldsymbol{\nabla}$ operator, Eq. (B.3) writes:

$$\boldsymbol{q}'_i = -\sum_{r=1}^{N} \left(C_i'^r \boldsymbol{K}'_{\mathrm{D},i} \right) \boldsymbol{\nabla}\phi_r, \quad i = 1, \ldots, m. \tag{B.29}$$

So, as $\boldsymbol{A}'_i = C'_i \boldsymbol{v} + \boldsymbol{q}'_i$, it is obtained:

$$\boldsymbol{A}_i = (\sum_{r=1}^{N} C_i'^r \phi_r)\boldsymbol{v} - \sum_{r=1}^{N} \left(C_i'^r \boldsymbol{K}'_{\mathrm{D},i} \right) \boldsymbol{\nabla}\phi_r, \quad i = 1, \ldots, m. \tag{B.30}$$

Consequently, $\boldsymbol{\nabla}\phi \cdot \boldsymbol{A}'_i$ writes:

$$\boldsymbol{\nabla}\phi \cdot \boldsymbol{A}'_i = \sum_{r=1}^{N} C_i'^r \left(\boldsymbol{\nabla}\phi \cdot \phi_r \boldsymbol{v} - \boldsymbol{\nabla}\phi \cdot (\boldsymbol{K}'_{D,i} \boldsymbol{\nabla}\phi_r) \right), \quad i = 1, \ldots, m. \tag{B.31}$$

Besides, as $C_i'^r = C_i'^r(t)$, $\phi \frac{\partial C'_i}{\partial t}$ writes

$$\phi \frac{\partial C_i}{\partial t} \phi_r = \phi \sum_{r=1}^{N} \dot{C}_i'^r \phi_r, \quad i = 1, \ldots, m. \tag{B.32}$$

Choosing, as done before, $\phi = \phi_s, s = 1, \ldots, N$, and reorganising terms in Eq. (B.31), we may write:

$$\boldsymbol{\nabla}\phi_s \cdot \boldsymbol{A}_i = \sum_{r=1}^{N} \hat{\mathbf{K}}_{sr}'^{(i)} C_i'^r, \quad i = 1, \ldots, m, \tag{B.33}$$

where we have defined:

$$\hat{\boldsymbol{K}}_{sr}'^{(i)} = \left(\boldsymbol{\nabla}\phi_s \cdot (\phi_r \boldsymbol{v}) - \boldsymbol{\nabla}\phi_s \cdot \boldsymbol{K}'_{\mathrm{D},i} \boldsymbol{\nabla}\phi_r \right), \quad i = 1, \ldots, m, \tag{B.34}$$

and Eq. (B.32) becomes:

$$\phi_s \sum_{r=1}^{N} \frac{\partial C_i'^r}{\partial t} \phi_r = \sum_{r=1}^{N} \hat{\mathbf{M}}_{sr}^{(i)} C_i'^r, \quad i = 1, \ldots, m. \tag{B.35}$$

Boundary conditions

Finally, Eq. (B.14), with $\phi = \phi_s$, $C_i = \sum_{r=1}^{N} C_i'^r \phi_r$ and $f'_i = \sum_{r=1}^{N} f_i'^r \phi_r$, becomes:

$$C_i'^r = f_i'^r, \quad i = 1, \ldots, m, \tag{B.36}$$

and Eq. (B.15), with $\phi = \phi_s$ and $S_i = \sum_{r=1}^{N} C_i'^r \phi_r$:

$$\sum_{r=1}^{N} (\int_{\Gamma_{R'_i}} \kappa'_i \phi_s \phi_r \, d\Gamma) C_i'^r + \int_{\Gamma_{R'_i}} \phi_s \frac{\partial C_i}{\partial \boldsymbol{n}} \, d\Gamma = \int_{\Gamma_{R'_i}} \phi_s g'_i \, d\Gamma, \quad i = 1, \ldots, m. \tag{B.37}$$

B.1.3 Compact form.

For notation purposes, we define the vectors $\mathbf{C}_i = (C_i^1, \dots, C_i^N)$, $\mathbf{C}'_i = (C_i'^1, \dots, C_i'^N)$, $\boldsymbol{\theta} = (\theta^1, \dots, \theta^N)$, $\mathbf{V} = (V^1, \dots, V^N)$, $\mathbf{W} = (\mathbf{v}^1, \dots, \mathbf{v}^N)$ and $\mathbf{P}_k = (p_k^1, \dots, p_k^N)$.

With these notations, Eq. (B.17), with $\phi = \phi_s$ and using Eq. (B.27), becomes:

$$\mathbf{M}^{(i)}\dot{\mathbf{C}}_i + \mathbf{K}^{(i)}\mathbf{C}_i + \sum_{j=1}^{m} \mathbf{K}^{(ij)}\mathbf{C}'_j = \mathbf{F}^{(i)}, \quad i = 1, \dots, n. \tag{B.38}$$

with:

$$\mathbf{K}_{sr}^{(i)} = \int_\Omega \hat{\mathbf{K}}_{sr}^{(i)}\, d\Omega + \int_{\Gamma_{R_i}} \kappa_i \phi_s \phi_r\, d\Gamma, \quad i = 1, \dots, n, \tag{B.39a}$$

$$\mathbf{K}_{sr}^{(ij)} = \int_\Omega \hat{\mathbf{K}}_{sr}^{(ij)}\, d\Omega + \int_{\Gamma_{R_j}} \kappa_j \phi_s \phi_r\, d\Gamma, \quad i = 1, \dots, n, \tag{B.39b}$$

$$\mathbf{M}_{sr}^{(i)} = \int_\Omega \hat{\mathbf{M}}_{sr}\, d\Omega, \quad i = 1, \dots, n, \tag{B.39c}$$

$$\mathbf{F}_s^{(i)} = \int_\Omega f_i \phi_s\, d\Omega + \int_{\Gamma_{R_i}} \phi_s g_i\, d\Gamma - \int_\Omega \hat{\mathbf{F}}_s^{(i)}\, d\Omega, \quad i = 1, \dots, n. \tag{B.39d}$$

Defining $\mathbf{C} = (\mathbf{C}_1, \dots, \mathbf{C}_n)^\intercal$, $\mathbf{C}' = (\mathbf{C}'_1, \dots, \mathbf{C}'_m)^\intercal$, $\mathbf{P} = (\mathbf{P}_1, \dots, \mathbf{P}_k)^\intercal$, $\mathbf{M} = \bigoplus_{i=1}^n \mathbf{M}^{(i)}$,

$$\mathbf{K}_C = \begin{bmatrix} \mathbf{K}^{(1)} & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{0} & \mathbf{K}^{(2)} & \dots & \mathbf{0} \\ \dots & \dots & \dots & \dots \\ \mathbf{0} & \mathbf{0} & \dots & \mathbf{K}^{(nn)} \end{bmatrix}, \tag{B.40}$$

$$\mathbf{K}_{CS} = \begin{bmatrix} \mathbf{K}^{(11)} & \mathbf{K}^{(12)} & \dots & \mathbf{K}^{(1m)} \\ \mathbf{K}^{(21)} & \mathbf{K}^{(22)} & \dots & \mathbf{K}^{(2m)} \\ \dots & \dots & \dots & \dots \\ \mathbf{K}^{(n1)} & \mathbf{K}^{(n2)} & \dots & \mathbf{K}^{(nm)} \end{bmatrix}, \tag{B.41}$$

and $\mathbf{F}_C = (\mathbf{F}^{(1)}; \dots, \mathbf{F}^{(n)})$, Eq. (B.38) for $i = 1, \dots, n$ can be expressed in a compact form as:

$$\mathbb{M}_C\dot{\mathbf{C}} + \mathbb{K}_C\mathbf{C} + +\mathbb{K}_{CS}\mathbf{S} = \mathbf{F}_C. \tag{B.42}$$

Using the same vectorial notations, Eq. (B.28) becomes:

$$\mathbf{M}'^{(i)}\dot{\mathbf{S}}_i + \mathbf{K}'^{(i)}\mathbf{C}'_i = \mathbf{F}'^{(i)}, \quad i = 1, \dots, m, \tag{B.43}$$

where we have denoted:

$$\mathbf{K}_{sr}'^{(i)} = \int_\Omega \hat{\mathbf{K}}_{sr}'^{(i)}\, d\Omega + \int_{\Gamma_{R'_i}} \kappa'_i \phi_s \phi_r\, d\Gamma, \quad i = 1, \dots, m, \tag{B.44a}$$

$$\mathbf{M}_{sr}'^{(i)} = \int_\Omega \hat{\mathbf{M}}_{sr}\, d\Omega, \quad i = 1, \dots, m, \tag{B.44b}$$

$$\mathbf{F}_s'^{(i)} = \int_\Omega \hat{\mathbf{F}}_s'^{(i)}\, d\Omega + \int_{\Gamma_{R'_i}} \phi_s g'_i\, d\Gamma, \quad i = 1, \dots, m. \tag{B.44c}$$

Defining $\mathbf{M}' = \bigoplus_{i=1}^{m} \mathbb{M}'^{(i)}$, $\mathbb{K}_S = \bigoplus_{i=1}^{m} \mathbb{K}'^{(i)}$, and $\mathbf{F}_S = (\mathbf{F}'^{(1)}; \ldots, \mathbf{F}'^{(m)})$, Eq. (B.43) for $i = 1, \ldots, m$ can be expressed in a compact form as:

$$\mathbb{M}_S \dot{\mathbf{S}} + \mathbf{K}_S \mathbf{S} = \mathbf{F}_S. \tag{B.45}$$

Besides, using again vectorial notation, the regeneration term may be stated as:

$$\dot{\mathbf{P}} = \mathbf{R}(\mathbf{P}, \mathbf{C}, \boldsymbol{\theta}). \tag{B.46}$$

Eqs. (B.42) and (B.45) may be combined if we define $\mathbf{U} = (\mathbf{C}, \mathbf{C}')$, $\mathbf{H} = (\mathbf{W}, \mathbf{V}, \boldsymbol{\theta})$, $\mathbf{M} = \mathbf{M} \oplus \mathbf{M}'$, $\mathbf{F} = (\mathbf{F}, \mathbf{F}')$, and

$$\mathbf{K} = \begin{bmatrix} \mathbf{K}_C & \mathbf{K}_{CS} \\ \mathbf{0} & \mathbf{K}_S \end{bmatrix}, \tag{B.47}$$

arriving finally to the equations:

$$\mathbf{M}\dot{\mathbf{U}} + \mathbf{K}\mathbf{U} = \mathbf{F}, \tag{B.48a}$$

$$\dot{\mathbf{P}} = \mathbf{R}. \tag{B.48b}$$

Note that these equations are nonlinear, so if we explicitly state the functional dependencies, it is possible to write:

$$\mathbf{M}\dot{\mathbf{U}} + \mathbf{K}(\mathbf{U}, \mathbf{P}, \mathbf{H})\mathbf{U} = \mathbf{F}(\mathbf{U}, \mathbf{P}, \mathbf{H}), \tag{B.49a}$$

$$\dot{\mathbf{P}} = \mathbf{R}(\mathbf{U}, \mathbf{P}, \mathbf{H}). \tag{B.49b}$$

Eq. (B.48) may be combined with Dirichlet boundary conditions, given in Eqs. (B.26) and (B.36). For the former, $\mathbf{C}_i = \mathbf{f}_i(t)$ is known at nodes belonging to Γ_{D_i} while for the latter, $\mathbf{C}'_i = \mathbf{f}'_i(t)$ is known at nodes belonging to $\Gamma_{D'_i}$. If $\bar{\mathbf{U}} = \bar{\mathbf{U}}(t)$ are the known values, Eq. (B.48a) may be split symbolically in free and constrained variables:

$$\mathbf{M}_f \dot{\mathbf{U}}_f + \mathbf{M}_c \dot{\bar{\mathbf{U}}} + \mathbf{K}_f \mathbf{U}_f + \mathbf{K}_c \bar{\mathbf{U}} = \mathbf{F}. \tag{B.50}$$

So, we finally obtain:

$$\mathbf{M}^* \dot{\mathbf{U}}^* + \mathbf{K}^* \mathbf{U}^* = \mathbf{F}^*. \tag{B.51}$$

where $\mathbf{M}^* = \mathbf{M}_f$, $\mathbf{K}^* = \mathbf{K}_f$, $\mathbf{F}^* = \mathbf{F} - \mathbf{M}_c \dot{\bar{\mathbf{U}}} - \mathbf{K}_c \bar{\mathbf{U}}$ and $\mathbf{U}^* = \mathbf{U}_f$ are the unknowns of the problem.

The final system of ordinary differential equations (ODEs) to be solved is, therefore:

$$\mathbf{M}^* \dot{\mathbf{U}}^* + \mathbf{K}^*(\mathbf{U}^*, \mathbf{P}, \mathbf{H}(t))\mathbf{U}^* = \mathbf{F}(\mathbf{U}^*, \mathbf{P}, \mathbf{H}(t), \bar{\mathbf{U}}(t), \dot{\bar{\mathbf{U}}}(t)), \tag{B.52a}$$

$$\dot{\mathbf{P}} = \mathbf{R}(\mathbf{U}^*, \mathbf{H}(t), \bar{\mathbf{U}}(t)). \tag{B.52b}$$

B.1.4 Time integration

In order to solve Eq. (B.52), that is, to compute $\mathbf{U}^* = \mathbf{U}^*(t)$ and $\mathbf{P} = \mathbf{P}(t)$, for $t \in [0;T]$, it is necessary to define the initial conditions $\mathbf{U}_0^* = \mathbf{U}^*(t=0)$ and $\mathbf{P}_0 = \mathbf{P}(t=0)$ and to specify $\mathbf{H} = \mathbf{H}(t)$ (physical stimuli, including electrical, thermal, and flow stimuli) and $\bar{\mathbf{U}} = \bar{\mathbf{U}}(t)$, (boundary conditions) for $t \in [0;T]$.

This problem may be expressed in a standard form using the symbolic notation:

$$\mathbf{x} = (\mathbf{U}^*, \mathbf{P}), \tag{B.53}$$

$$\mathbf{\Lambda} = (\mathbf{H}, \bar{\mathbf{U}}, \dot{\bar{\mathbf{U}}}), \tag{B.54}$$

$$\mathbf{f}(\mathbf{x},t) = \begin{bmatrix} \mathbf{M}^{-1}\left(\mathbf{F}^*(\mathbf{U}^*,\mathbf{P},\mathbf{\Lambda}(t)) - \mathbf{K}^*(\mathbf{U}^*,\mathbf{P},\mathbf{\Lambda}(t))\right) \\ \mathbf{R}(\mathbf{U}^*,\mathbf{\Lambda}(t)) \end{bmatrix}, \tag{B.55}$$

$$\mathbf{x}_0 = (\mathbf{U}_0^*, \mathbf{P}_0), \tag{B.56}$$

where $\mathbf{x} = \mathbf{x}(t)$ are the evolution variables, $\mathbf{\Lambda} = \mathbf{\Lambda}(t)$ are the external excitations, $\mathbf{f} = \mathbf{f}(\mathbf{x}, \mathbf{\Lambda}) = \mathbf{f}(\mathbf{x}, t)$ is the dynamic flow, and $\mathbf{x}_0$ the initial state.

Thereby, the equations to solve are:

$$\frac{d\mathbf{x}}{dt} = \mathbf{f}(\mathbf{x}, t), \tag{B.57a}$$

$$\mathbf{x}(t=0) = \mathbf{x}_0. \tag{B.57b}$$

Eq. (B.57) is a nonlinear ODE system that may be solved using different numerical schemes, accounting for the high nonlinearity, coupling between variables and stiff behaviour of the differential equation. Many numerical integrators have been implemented, including forward Euler, backward Euler, Midpoint, Adams-Bashforth, or adaptative Runge-Kutta solvers, such as the Dormand-Prince scheme (Dormand and Prince, 1980) or the Bogacki-Shampine scheme (Bogacki and Shampine, 1989).

Forward Euler method.

Forward Euler is the simplest numerical integrator. Using forward Euler integration, Eq. (B.57) writes:

$$\mathbf{x}_{t+1} = \mathbf{x}_t + \mathbf{f}_t \Delta t, \tag{B.58}$$

where $\mathbf{x}_t = \mathbf{x}(t)$, $\mathbf{x}_{t+1} = \mathbf{x}(t+\Delta t)$, and $\mathbf{f}_t = \mathbf{f}(\mathbf{x}_t, t)$. Eq. (B.58) implies just an evaluation because it has an explicit nature. However, it is known that forward Euler method can also be numerically unstable, especially for stiff equations, requiring very small time-steps for obtaining accurate results.

Backward Euler combined with Broyden method.

Backward Euler method is a numerical integrator that may work for greater time-steps than forward Euler, due to its implicit nature. However, because of this, at each time-step, a multidimensional nonlinear equation must be solved. Eq. (B.57) discretised by means of backward Euler method writes:

$$\mathbf{x}_{t+1} = \mathbf{x}_t + \mathbf{f}_{t+1} \Delta t, \tag{B.59}$$

where $\mathbf{x}_t = \mathbf{x}(t)$, $\mathbf{x}_{t+1} = \mathbf{x}(t+\Delta t)$ and $\mathbf{f}_{t+1} = \mathbf{f}(\mathbf{x}_{t+1}, t+\Delta t)$. Eq. (B.59) is nonlinear in general and has to be solved iteratively. Computation of the tangent operator, $\frac{\partial \mathbf{f}}{\partial \mathbf{x}}$ is computationally expensive and hard to derive from the expression of the function $\mathbf{f}$. Instead of using the classical Newton method, based on the tangent operator, the multidimensional generalisation of the secant method, the Broyden method (Broyden, 1965), is going to be used. At each iteration, a secant operator, $\mathbf{J}_n$, is computed and an update of the variables is performed as $\mathbf{x}_{t+1,n+1} = \mathbf{x}_{t+1,n} - \mathbf{J}_n^{-1}\mathbf{r}_{t+1,n}$, where we have defined the residual at iteration n:

$$\mathbf{r}_{t+1,n} = \mathbf{x}_{t+1,n} - \mathbf{x}_{t,n} - \mathbf{f}_{t+1,n}\Delta t, \tag{B.60}$$

and the secant operator is defined such as satisfying

$$\mathbf{J}_n \Delta \mathbf{x}_{t+1,n} = \Delta \mathbf{f}_{t+1,n}, \tag{B.61}$$

where $\Delta \mathbf{x}_{t+1,n} = \mathbf{x}_{t+1,n+1} - \mathbf{x}_{t+1,n}$, and $\Delta \mathbf{f}_{t+1,n} = \mathbf{f}_{t+1,n+1} - \mathbf{f}_{t+1,n}$.

Of course, if the dimensions of $\mathbf{x}$ and $\mathbf{f}$ are greater than one, Eq. (B.61) is undetermined and further conditions shall be supplied. A possibility is to use the current estimate of the secant operator $\mathbf{J}_{n-1}$ and improving upon it by taking the solution to the secant equation that is a minimal modification to $\mathbf{J}_{n-1}$ in terms of the Frobenius norm, that is, $||\mathbf{J}_n - \mathbf{J}_{n-1}||_{\mathrm{F}}$ is minimal. Thus:

$$\mathbf{J}_n = \mathbf{J}_{n-1} + \frac{\Delta \mathbf{f}_n - \mathbf{J}_{n-1}\Delta \mathbf{x}_n}{||\Delta \mathbf{x}_n||^2} \Delta \mathbf{x}_n^\intercal. \tag{B.62}$$

Using the Sherman-Morrison formula (Sherman and Morrison, 1950), the inverse of the secant operator may be updated from Eq. (B.62) as follows:

$$\mathbf{J}_n^{-1} = \mathbf{J}_{n-1}^{-1} + \frac{\Delta \mathbf{x}_n - \mathbf{J}_{n-1}^{-1}\Delta \mathbf{f}_n}{\Delta \mathbf{x}_n^\intercal \mathbf{J}_{n-1}^{-1} \Delta \mathbf{f}_n} \Delta \mathbf{x}_n^\intercal \mathbf{J}_{n-1}^{-1}. \tag{B.63}$$

Finally, at each iteration, the residual given by Eq. (B.60) is evaluated, where

$$\mathbf{r}_{t+1,n} = \mathbf{x}_{t+1,n} - \mathbf{x}_{t,n} - \mathbf{f}_{t+1,n}\Delta t. \tag{B.64}$$

The iteration step stops when $||\mathbf{r}_{t+1,n}|| < \mathrm{TOL}$. Note that with this approach, at each iteration, we have an evaluation of function $\mathbf{f}_n$, that is, a construction of operators $\mathbf{M}$, $\mathbf{K}$, and $\mathbf{f}$.

B.2 1D Finite Element implementation

B.2.1 One-dimensional equations

B.2.1.1 Evolution equations

Differential Eqs. (B.1) with boundary conditions (B.7) and (B.7) may be easily expressed for one-dimensional problems. Let us define:

$$u_i = C_i, \quad i = 1, \dots, n, \tag{B.65a}$$
$$u_{n+i} = C_i', \quad i = 1, \dots, m, \tag{B.65b}$$
$$u_{n+m+i} = p_i, \quad i = 1, \dots, k, \tag{B.65c}$$

and $\boldsymbol{u} = (u_0, \ldots, u_{n+m+k})^\intercal$.

Also, let us define:[38]

$$f_i = -A_i, \quad i = 1, \ldots, n, \tag{B.66a}$$
$$f_{n+i} = -A'_i, \quad i = 1, \ldots, m, \tag{B.66b}$$
$$f_{n+m+i} = 0, \quad i = 1, \ldots, k, \tag{B.66c}$$

and $\boldsymbol{f} = (f_1, \ldots, f_{n+m+k})^\intercal$.

Finally, let us define:

$$s_i = a_i, \quad i = 1, \ldots, n, \tag{B.67a}$$
$$s_{n+i} = a'_i, \quad i = 1, \ldots, m, \tag{B.67b}$$
$$s_{n+m+i} = r_i, \quad i = 1, \ldots, k, \tag{B.67c}$$

and $\boldsymbol{s} = (s_1, \ldots, s_{n+m+k})^\intercal$. With these notations, it is possible to summarise the governing equations as:

$$\frac{\partial \boldsymbol{u}}{\partial t} = \frac{\partial \boldsymbol{f}}{\partial x} + \boldsymbol{s}, \tag{B.68}$$

where $\boldsymbol{f} = \boldsymbol{f}(x, t, \boldsymbol{u}, \frac{\partial \boldsymbol{u}}{\partial x})$ and $\boldsymbol{s} = \boldsymbol{s}(x, t, \boldsymbol{u})$.

B.2.1.2 Boundary and initial conditions

Eq. (B.68) has sense if and only if we define suitable boundary conditions and initial conditions. Boundary conditions are for each variable $u_i, i = 1, \ldots, n+m$ given by Eqs. (B.7) or (B.8) except for the fact that we replace directional derivatives $\frac{\partial}{\partial \mathbf{n}}$ by partial derivatives $\frac{\partial}{\partial x}$ (with the proper orientation). For the terms $u_i, i = n+m, \ldots, n+m+k$, we have merely a system of ODEs as no flux terms are involved in their variations.

Finally, initial conditions are:

$$\boldsymbol{u} = \boldsymbol{u}^0(x), \tag{B.69}$$

where $u_i^0 = u_i^0$ for $i = 1, \ldots, n$, $u_{n+i}^0 = C_i'^0$ for $i = 1, \ldots, m$ and $u_{n+m+i}^0 = p_i^0$ for $i = 1, \ldots, k$.

B.2.2 Weak form

The PDEs (B.68) with boundary conditions (B.7) and (B.8) and initial conditions (B.69) is a nonlinear parabolic PDE in time and one space variable. We solve numerically this equation using a method based on a simple piecewise nonlinear Galerkin second-order method accurate in space (Skeel and Berzins, 1990), which is compatible with this kind of nonlinear equations and boundary conditions[39]. Multiplying the PDE by a test function $\phi \in \mathcal{H}_0^1([\alpha; \beta])$ and integrating by parts in $[\alpha; \beta]$, we arrive to:

$$\phi(\beta)\boldsymbol{f}(\beta) - \phi(\alpha)\boldsymbol{f}(\alpha) - \int_\alpha^\beta \frac{\partial \phi}{\partial x}\boldsymbol{f}\, dx = \int_\alpha^\beta \phi \boldsymbol{Q}\, dx, \tag{B.70}$$

[38] Note that, as the problem is one-dimensional, the flux term is now a scalar function.

[39] This is true for moderately low fluxes and convection terms.

where $\boldsymbol{Q} = \boldsymbol{Q}(x, t, \boldsymbol{u}, \frac{\partial \boldsymbol{u}}{\partial t}) = \frac{\partial \boldsymbol{u}}{\partial t} - \boldsymbol{s}(x, t, \boldsymbol{u})$. As test function we select:

$$\phi_\alpha(x) = \frac{\beta - x}{\beta - \alpha},$$
$$\phi_\beta(x) = \frac{x - \alpha}{\beta - \alpha}.$$

When using ϕ_α, Eq. (B.70) becomes:

$$-\int_\alpha^\beta \boldsymbol{f} \frac{\partial \phi_\alpha}{\partial x}\, dx = \boldsymbol{f}(\alpha) + \int_\alpha^\beta \boldsymbol{Q}\phi_\alpha\, dx, \tag{B.71}$$

and when using ϕ_β, we get:

$$\int_\alpha^\beta \boldsymbol{f} \frac{\partial \phi_\beta}{\partial x}\, dx = -\boldsymbol{f}(\beta) + \int_\alpha^\beta \boldsymbol{Q}\phi_\beta\, dx. \tag{B.72}$$

B.2.3 Numerical solution

B.2.3.1 Spatial discretisation

After numerical quadrature we get, for $\xi \in [\alpha; \beta]$:

$$-\boldsymbol{f}(\xi, \boldsymbol{u}(\xi), \frac{\partial \boldsymbol{u}}{\partial x}(\xi)) \int_\alpha^\beta \frac{\partial \phi_\alpha}{\partial x}\, dx = \boldsymbol{f}(\alpha) + \boldsymbol{Q}(\xi, t, \boldsymbol{u}(\xi), \frac{\partial \boldsymbol{u}}{\partial t}(\xi)) \int_\alpha^\beta \phi_\alpha\, dx. \tag{B.73}$$

Identifying $[\alpha; \beta]$ with $[x_{j-1}; x_j]$ we get, after a discretisation:

$$\mathbf{f}_{j-1/2} = \mathbf{v}_{j-1} + (\xi_{j-1/2} - x_{j-1})(\dot{\mathbf{u}}_{j-1} - \mathbf{s}_{j-1/2}), \tag{B.74}$$

where $\mathbf{v}_{j-1} = \mathbf{f}_{j-1}$ is considered as a secondary variable of the problem. In a similar way, using ϕ_β, we arrive to:

$$-\mathbf{f}_{j-1/2} = -\mathbf{v}_j + (x_j - \xi_{j-1/2})(\dot{\mathbf{u}}_j - \mathbf{s}_{j-1/2}). \tag{B.75}$$

Adding (B.74) at index $j + 1$ and (B.75), we obtain:

$$\mathbf{f}_{j+1/2} - \mathbf{f}_{j-1/2} = (\xi_{j+1/2} - x_j)(\dot{\mathbf{u}}_j - \mathbf{s}_{j+1/2}) + (x_j - \xi_{j-1/2})(\dot{\mathbf{u}}_j - \mathbf{s}_{j-1/2}). \tag{B.76}$$

B.2.3.2 Time integration

Eqs. (B.74), (B.75) and (B.76), including boundary conditions, form a system of differential-algebraic equations (DAEs) that are integrated using MATLAB ODE suite (Shampine et al., 1999). For the time integrator, an absolute tolerance of 10^{-6} is fixed and a relative tolerance of 10^{-3}.

B.3 Code validation

Next, we present some results that give robustness to the implemented code. It is important to note that the next simulations have no particular biological meaning (even if the variable units have physical meaning) and are here exposed only for numerical validation purposes.

B.3.1 Two-dimensional problem validation

B.3.1.1 Description of 2D the problem

In the problem solved there is one cell phenotype C and one chemical species C', whose properties are going to be chosen with the aim of reducing the integration time, and therefore they don't have any physical meaning. In the following paragraphs, the main aspects regarding the simulation are presented.

Geometry

A 603 node mesh is defined using 400 four-node quadrilateral elements (with linear approximation) situated in two identical rows with 200 elements each. The elements' dimensions are $0.005\,\text{cm} \times 0.005\,\text{cm}$ and the total dimension of the mesh is $1\,\text{cm} \times 0.01\,\text{cm}$. It must be noted that in order to compare the results obtained with this mesh with a 1D problem, the mesh must be symmetrical about the horizontal axis. When we compare the results, we will take into account the solution in the central line of nodes.

Initial conditions

The initial condition for the cell population is chosen to have a step shape, as can be seen for the central line of nodes in Figure B.1. This is done with illustrative purposes, so that all the phenomena that are going to be tested have significant effects on the cell concentration profile.

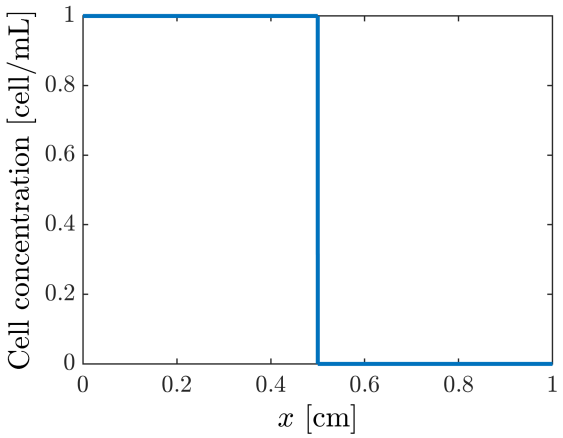


Figure B.1: Initial cell concentration profile along the central line of nodes in the mesh.

Regarding the chemical species, the initial concentration is chosen to be (Figure B.2):

$$C'(x, t = 0) = \sqrt{(x)}. \tag{B.77}$$

If dead cells appear, their initial concentration is assumed to be equal to zero in the whole simulation domain.

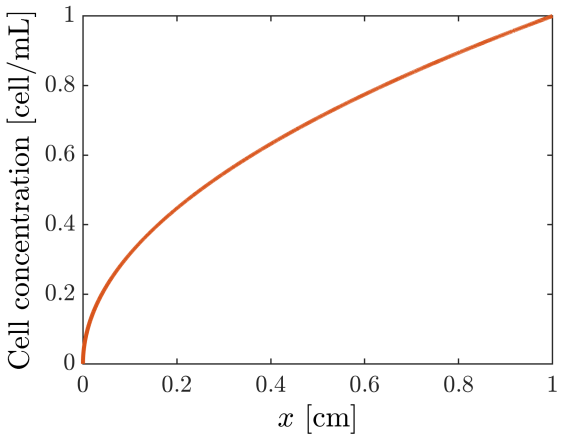


Figure B.2: Initial chemical species concentration profile along the central line of nodes in the mesh.

Boundary conditions

Neumann boundary conditions are selected for the upper and lower edges of the mesh (with $g_1 = g_1' = 0$, meaning there is no flux through these axes). This is done to ensure that there's no movement of cells or oxygen along the vertical axis and then we can compare the central line of nodes to a 1D problem as we intend to do. Dirichlet boundary conditions are selected for the lateral edges of the mesh, meaning that we fix the population concentration at these nodes. For the cell populations, we have:

$$f_1(x = 0) = 1\,\text{cell/mL}, \qquad f_1(x = L) = 0, \tag{B.78}$$

where L is the length of the mesh along the x axis (in this case, $L = 1$ cm). Equivalently, for the chemical species considered the boundary conditions at the lateral edges are:

$$f_1'(x = 0) = 0, \qquad f_1'(x = L) = 1\,\text{mmHg}. \tag{B.79}$$

With regard to the dead cells phenotype (appearing only in some problems), we select Neumann boundary conditions for all the contours, as they are the natural boundary conditions when there is no cell flux.

Numerical parameters and solver

The problem is solved using the MATLAB solver `ode45`, based on an explicit Runge-Kutta (4,5) formula, the Dormand-Prince pair. It is a single-step solver – in computing the solution at a precise time, it needs only the solution at the immediately preceding time point (Dormand and Prince, 1980; Shampine and Reichelt, 1997). The final time of the simulations is 0.25 s.

B.3.1.2 Description of 1D the problem

The results obtained for the 2D problem implemented were compared to the results obtained with a 1D model, solved `pdepe`, which solves initial-boundary value problems

for parabolic-elliptic PDEs in 1-D in the one space variable x and time t (Skeel and Berzins, 1990).

To solve the problem, a 1D mesh is defined, whose elements have a length of 0.002 cm. Dirichlet boundary conditions are imposed on both the right and the left sides, and the initial conditions are selected to coincide with those of the 2D problem.

B.3.1.3 Validation procedure

Since the presented model comprises many phenomena, we are going to validate the different phenomena separately, in different phases.

Next, the parameters used in each phase as well as the obtained results are presented.

Diffusion

In this phase, only diffusion, both for the cell phenotype C and for the chemical species C', is considered. Therefore, the equations that must be solved are:

$$\frac{\partial C}{\partial t} + \boldsymbol{\nabla} \cdot \boldsymbol{p} = 0, \tag{B.80}$$

$$\frac{\partial C'}{\partial t} + \boldsymbol{\nabla} \cdot \boldsymbol{p}' = 0. \tag{B.81}$$

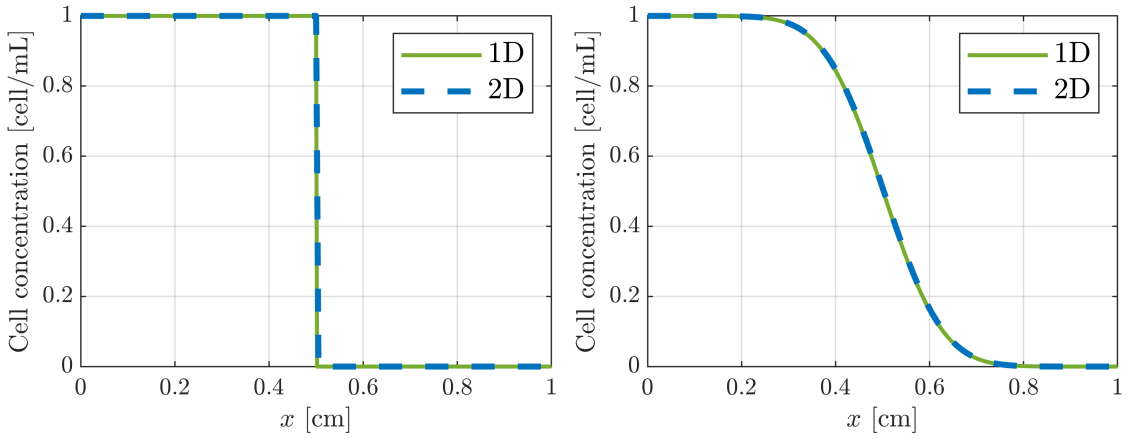
$\boldsymbol{p}$ and $\boldsymbol{p}$' are the flux terms of the cell concentration and chemical species equations and they are only formed with the diffusion term, such that:

$$\boldsymbol{p} = -K_D \boldsymbol{\nabla} C, \tag{B.82}$$

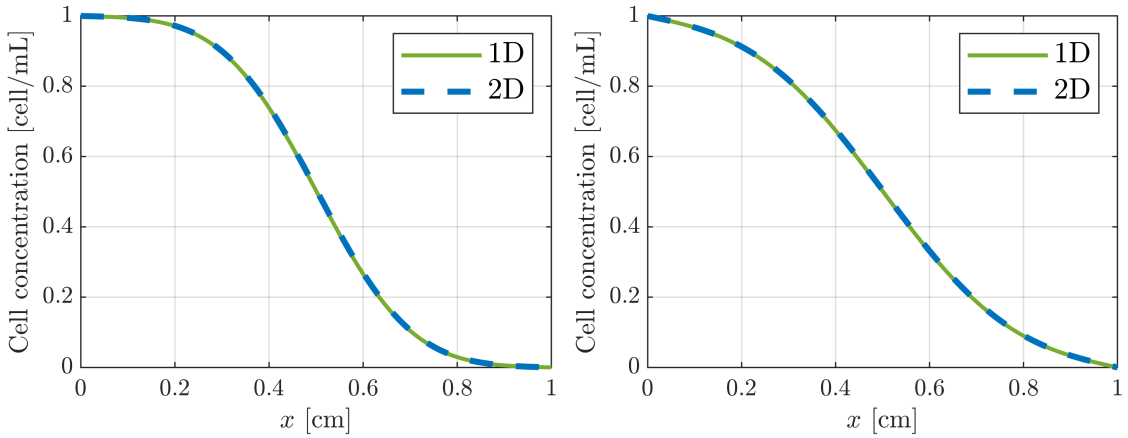
$$\boldsymbol{p}' = -K \boldsymbol{\nabla} C', \tag{B.83}$$

with $K_D = K = 0.1\,\mathrm{cm}^2/\mathrm{s}$ the diffusion coefficients. All the other terms involved in the general governing equations (Eqs. (B.1)) are considered equal to zero.

In Figs. B.3 and B.4 the results obtained with the 1D and 2D problems for both the cell phenotype and the chemical species are represented at different simulation times from the initial situation ($t = 0$) to the final time ($t = 0.25$ s). The results obtained with the general code coincide with the ones from the 1D MATLAB solver.

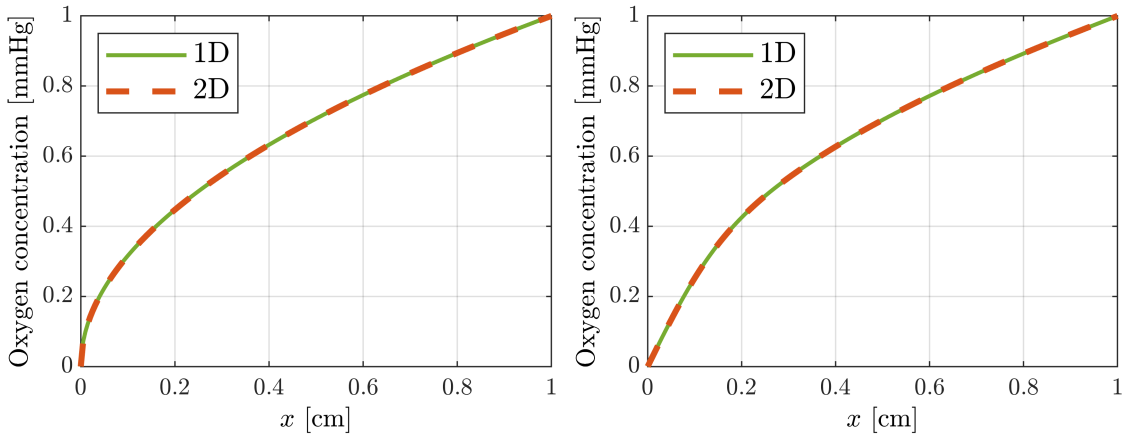


(a) $t = 0$ s. **(b)** $t = 0.05$ s.

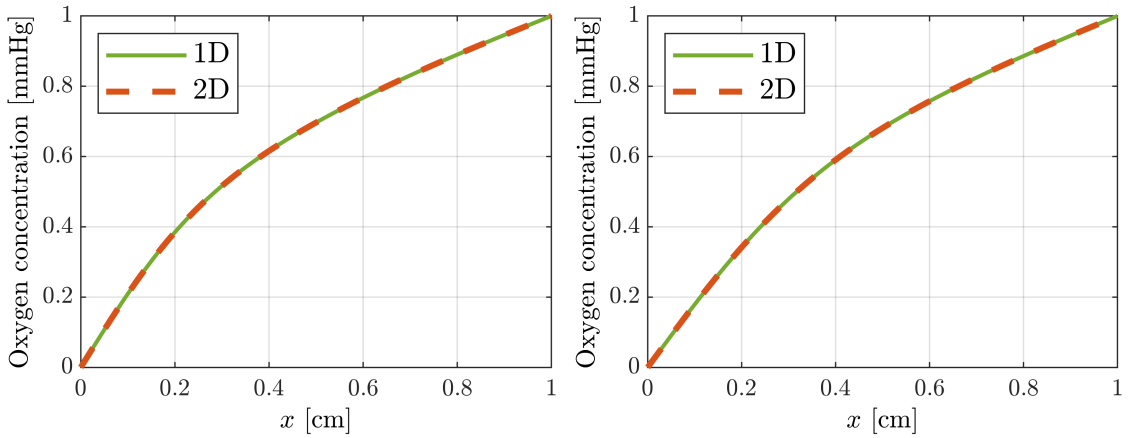


(c) $t = 0.125$ s. **(d)** $t = 0.25$ s.

Figure B.3: Cell concentration profiles obtained with both 1D and 2D models for diffusion validation.



(a) $t = 0$ s. **(b)** $t = 0.05$ s.



(c) $t = 0.125$ s. **(d)** $t = 0.25$ s.

Figure B.4: Oxygen concentration profiles obtained with both 1D and 2D models for diffusion validation.

Chemotaxis

In this phase, we consider diffusion and chemotaxis, both for the cell phenotype C and for the chemical species C'. Therefore, the equations that must be solved are:

$$\frac{\partial C}{\partial t} + \boldsymbol{\nabla} \cdot \boldsymbol{p} = 0, \tag{B.84}$$

$$\frac{\partial C'}{\partial t} + \boldsymbol{\nabla} \cdot \boldsymbol{p}' = 0. \tag{B.85}$$

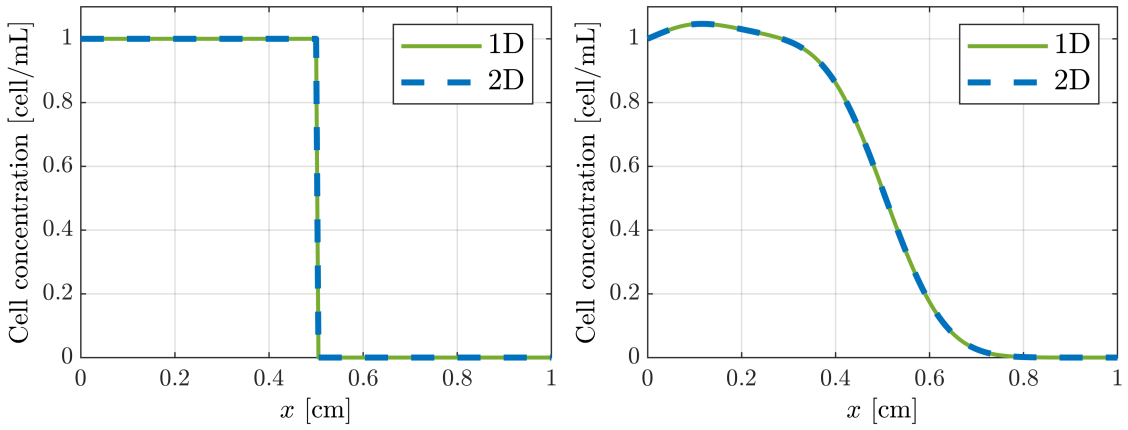
$\boldsymbol{p}$ and $\boldsymbol{p}$' are the flux terms of the cell concentration and chemical species equations and they are only formed with the diffusion term, such that:

$$\boldsymbol{p} = -K_D \boldsymbol{\nabla} C + \chi C \boldsymbol{\nabla} C', \tag{B.86}$$

$$\boldsymbol{p}' = -K \boldsymbol{\nabla} C', \tag{B.87}$$

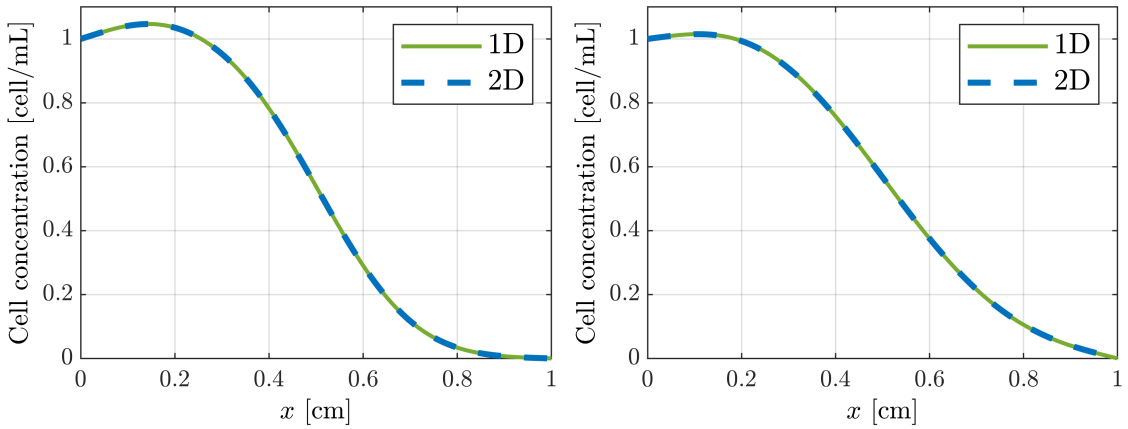
with $K_D = K = 0.1\,\mathrm{cm}^2/\mathrm{s}$ the diffusion coefficients and $\chi = 0.1\,\mathrm{cm}^2/(\mathrm{mmHg} \cdot \mathrm{s})$. All the other terms involved in the general governing equations (Eqs. (B.1)) are considered equal to zero.

In Figs. B.5 and B.6 the results obtained with the 1D and 2D problems for both the cell phenotype and the chemical species are represented at different simulation times from the initial situation ($t = 0$) to the final time ($t = 0.25$ s). The results obtained with the general code coincide again with the ones from the 1D MATLAB solver.



(a) $t = 0$ s.

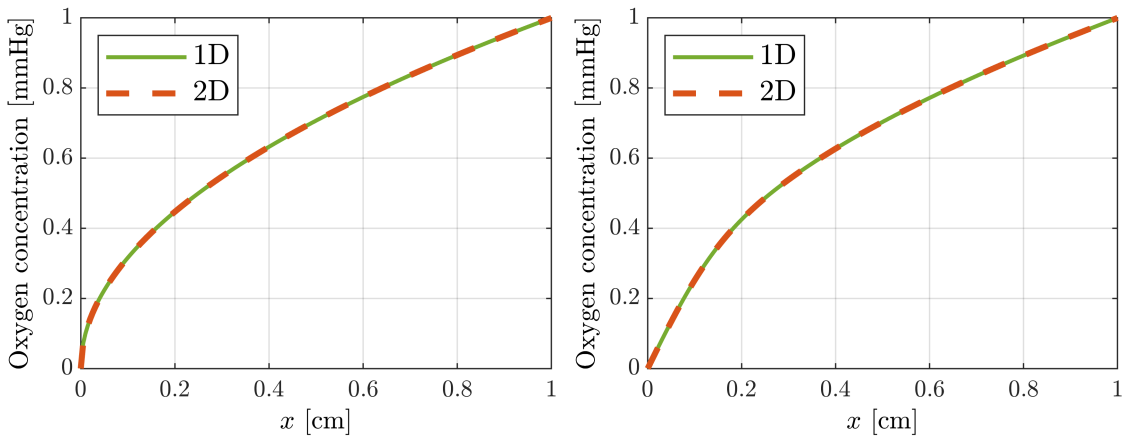
(b) $t = 0.05$ s.



(c) $t = 0.125$ s.

(d) $t = 0.25$ s.

Figure B.5: Cell concentration profiles obtained with both 1D and 2D models for chemotaxis validation.



(a) $t = 0$ s. **(b)** $t = 0.05$ s.

(c) $t = 0.125$ s. **(d)** $t = 0.25$ s.

Figure B.6: Oxygen concentration profiles obtained with both 1D and 2D models for chemotaxis validation.

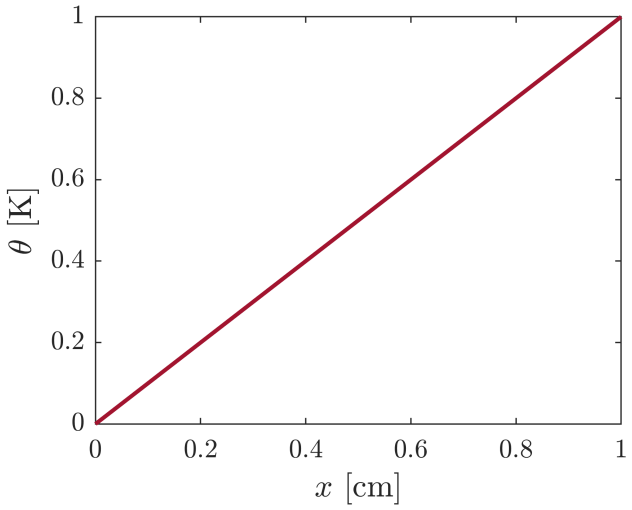


Figure B.7: Temperature profile.

Thermotaxis or electrotaxis

In this phase, we recreate the evolution of a cell phenotype C affected by diffusion and by migration directed by an external stimuli, in this case, thermotaxis (migration due to a temperature gradient). The electrotaxis term is implemented in the same way so thermotaxis and electrotaxis are validated in one stroke. The governing equation for C is then:

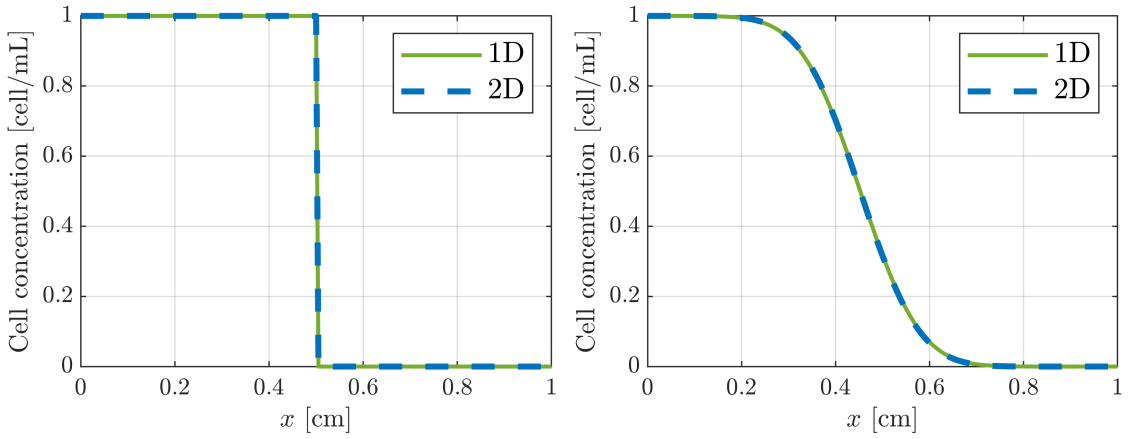
$$\frac{\partial C}{\partial t} + \boldsymbol{\nabla} \cdot \boldsymbol{p} = 0, \tag{B.88}$$

where the source term $\boldsymbol{p}$ can be written as:

$$\boldsymbol{p} = -K_D \boldsymbol{\nabla} \boldsymbol{C} + K_T C \boldsymbol{\nabla} \theta. \tag{B.89}$$

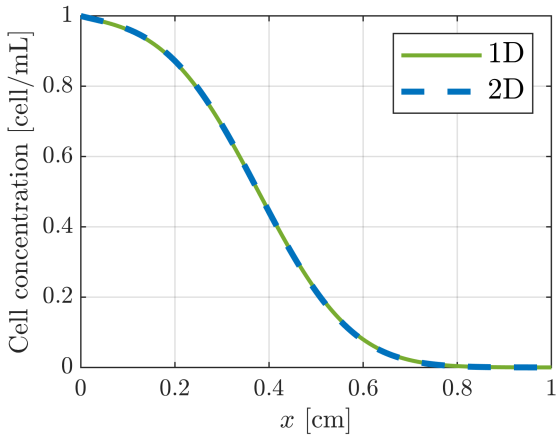
In this equation, θ represents the temperature field, K_D the diffusion coefficient and K_T the thermotaxis coefficient. The values of these coefficients are $K_D = 0.1\,\mathrm{cm}^2/\mathrm{s}$ and $K_T = 1\,\mathrm{cm}^2/(\mathrm{K}\cdot\mathrm{s})$. The temperature field is defined as represented in Figure B.7. The gradient shape is selected to obtain observable thermotaxis effects.

The simulation results are shown in Figure B.8.

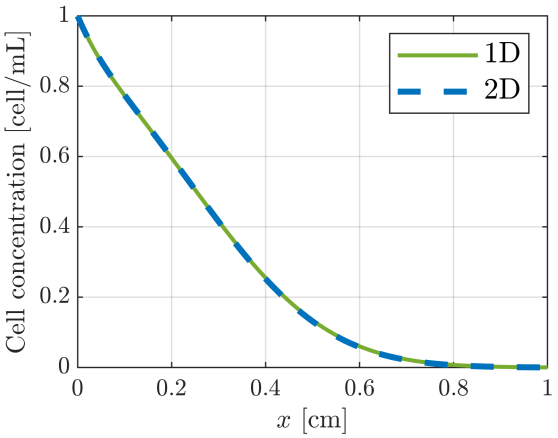


(a) $t = 0$ s.

(b) $t = 0.05$ s.



(c) $t = 0.125$ s.



(d) $t = 0.25$ s.

Figure B.8: Cell concentration profiles obtained with both 1D and 2D models for thermotaxis or electrotaxis validation.

External flow

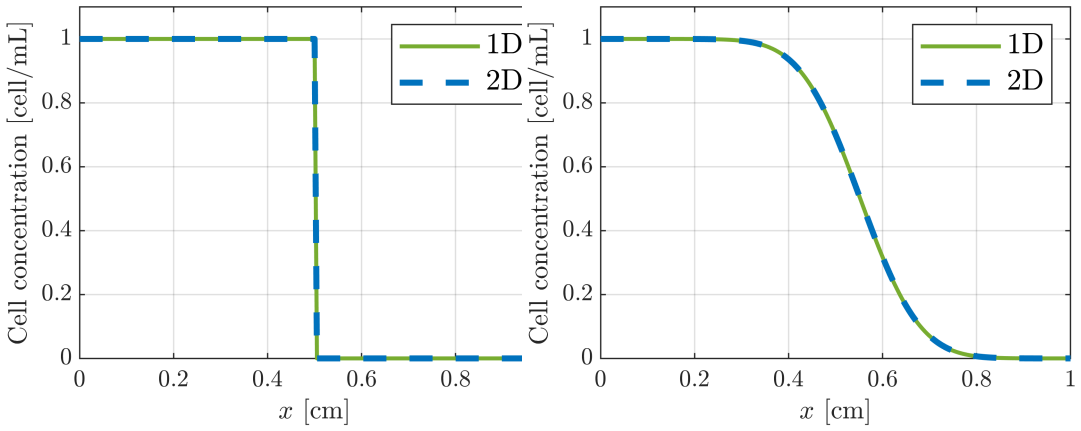
In this phase, we want to validate the convection due to an external flow. Therefore, we consider a cell phenotype C subjected to diffusion and to a constant flow in the direction of the x axis. The equation to be solved is then:

$$\frac{\partial C}{\partial t} + (\boldsymbol{v} \cdot \boldsymbol{\nabla})C + \boldsymbol{\nabla} \cdot \boldsymbol{p} = 0, \tag{B.90}$$

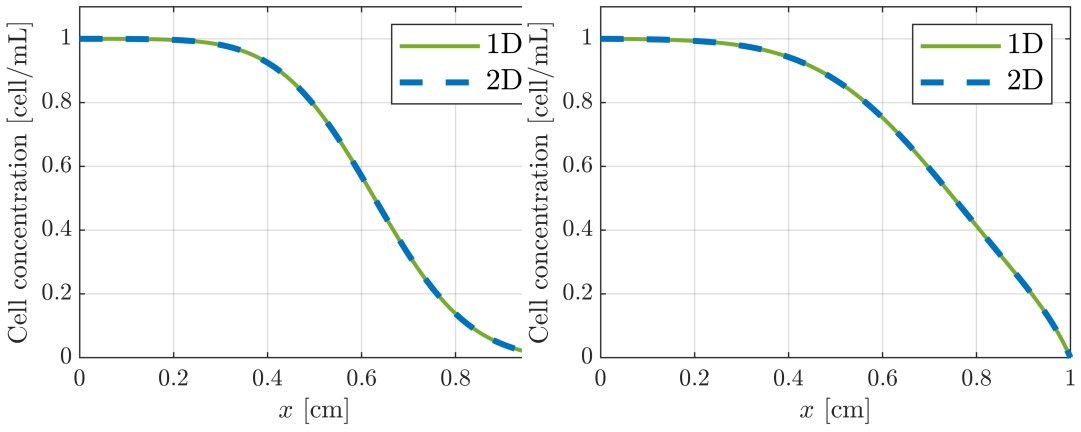
where $v = 0.1\,\mathrm{m}^2/\mathrm{s}$ is the flow velocity and the flux term $\boldsymbol{p}$ is written as:

$$\boldsymbol{p} = -K_D \boldsymbol{\nabla} C, \tag{B.91}$$

with $K_D = 0.1\,\mathrm{cm}^2/\mathrm{s}$ the diffusion coefficient. The simulation results are shown in Figure B.9, showing again good agreement between 1D and 2D results.



(a) $t = 0$ s. **(b)** $t = 0.05$ s.



(c) $t = 0.125$ s. **(d)** $t = 0.25$ s.

Figure B.9: Cell concentration profiles obtained with both 1D and 2D models for external flow validation.

Source term

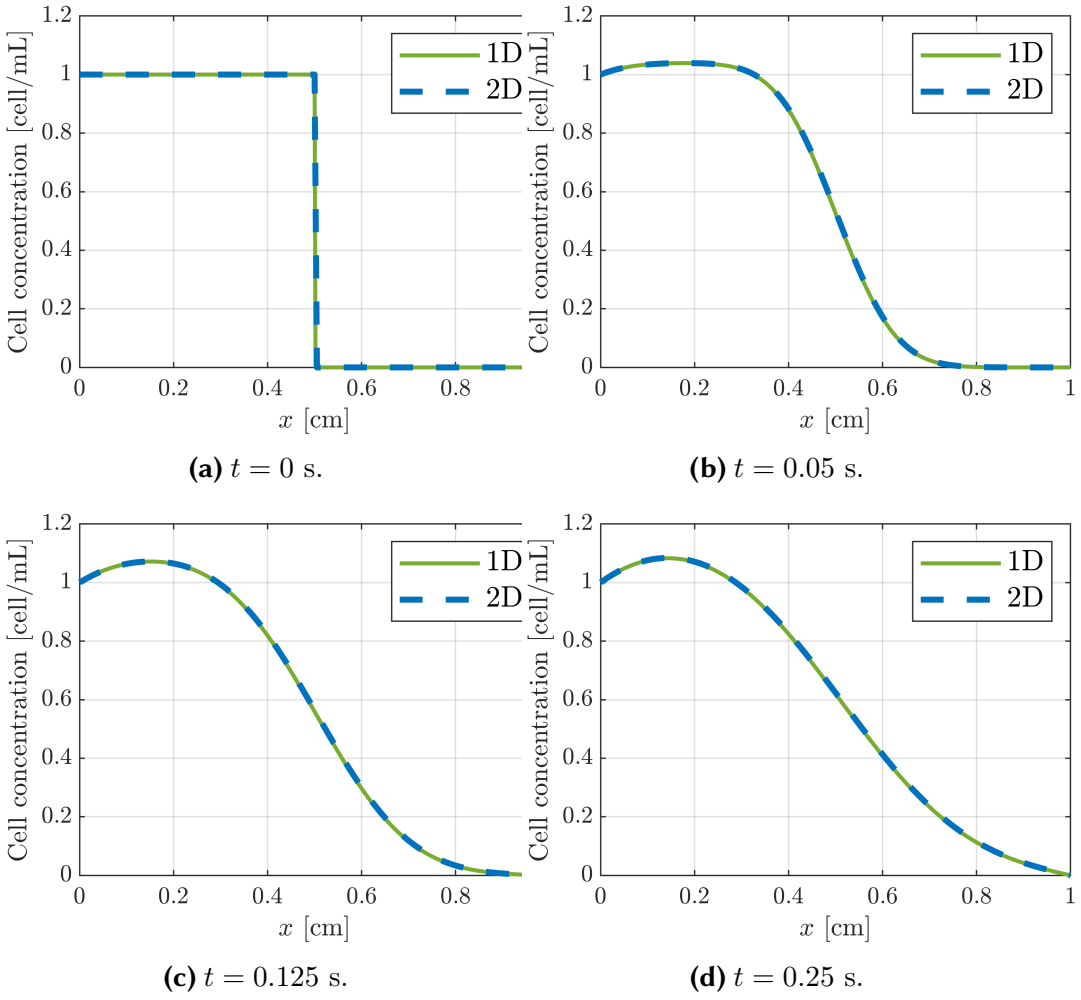
In this phase, we want to validate the source term affecting the cell phenotype equation. For that purpose, we consider a cell phenotype C subjected to diffusion and with a source term accounting for cell proliferation. The equation to be solved is then:

$$\frac{\partial C}{\partial t} + \boldsymbol{\nabla} \cdot \boldsymbol{p} = \alpha C, \tag{B.92}$$

where $\alpha = 1\,\mathrm{s}^{-1}$ is the growth rate and the flux term $\boldsymbol{p}$ is written as:

$$\boldsymbol{p} = -K_D \boldsymbol{\nabla} C, \tag{B.93}$$

with $K_D = 0.1\,\mathrm{cm}^2/\mathrm{s}$ the diffusion coefficient. The simulation results are shown in Figure B.9, showing once more good agreement between 1D and 2D results.



(a) $t = 0$ s. **(b)** $t = 0.05$ s.

(c) $t = 0.125$ s. **(d)** $t = 0.25$ s.

Figure B.10: Cell concentration profiles obtained with both 1D and 2D models for source term validation.

Complete problem with source and chemotaxis

In this phase, we consider most of the phenomena involved in a real cancer evolution problem. That is, we consider diffusion, chemotaxis, proliferation and death for the alive cells as well as diffusion and consumption for the chemical species. The initial concentration for the chemical species is, in this case, linear, as shown in Figure B.2.

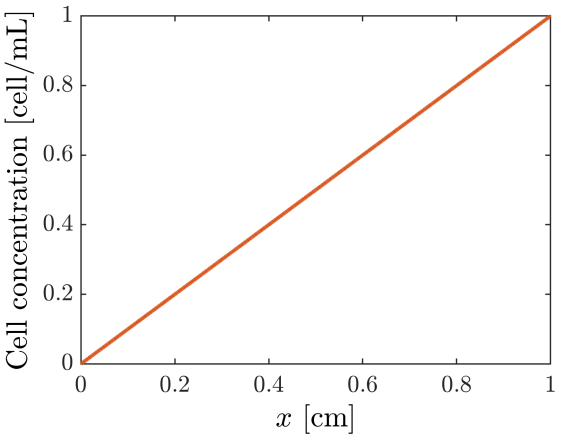


Figure B.11: Initial chemical species concentration profile along the central line of nodes in the mesh for the complete validation problem.

The equation implemented for the cell phenotype C is then:

$$\frac{\partial C_1}{\partial t} + \boldsymbol{\nabla} \cdot \boldsymbol{p} = a\Pi_{\mathrm{gr}} F_{\mathrm{gr}} C_1 - bF_{\mathrm{d}} C_1, \tag{B.94}$$

where the flux term is expressed as:

$$\boldsymbol{p} = -K_D \boldsymbol{\nabla} C_1 + K_{\mathrm{go}} \Pi_{\mathrm{go}} F_{\mathrm{go}} C_1 \boldsymbol{\nabla} C_1'. \tag{B.95}$$

Eqs. (B.94) and (B.95) incorporate a number of correction and activation functions accounting for cell metabolism. These functions are:

$$\Pi_{\mathrm{go}}(C_1') = \begin{cases} 1 - C_1'/\beta & \text{if} \quad 0 \leq C_1' \leq \beta \\ 0 & \text{if} \quad C_1' > \beta \end{cases}, \tag{B.96}$$

$$\Pi_{\mathrm{gr}}(C_1') = \begin{cases} C_1'/\beta & \text{if} \quad 0 \leq C_1' \leq \beta \\ 1 & \text{if} \quad C_1' > \beta \end{cases}, \tag{B.97}$$

$$F_{\mathrm{go}}(C_1) = \begin{cases} 1 - C_1/C^* & \text{if} \quad 0 \leq C_1 \leq C^* \\ 0 & \text{if} \quad C_1 > C^* \end{cases}, \tag{B.98}$$

$$F_{\mathrm{gr}}(C_1, C_2) = \left(1 - \frac{C_1 + C_2}{C^*}\right), \tag{B.99}$$

$$\Pi_{\mathrm{d}}(C_1') = \frac{1}{2}\left(1 - \tanh\left(\frac{C_1' - \gamma}{\Delta\gamma}\right)\right). \tag{B.100}$$

The equation for the dead cells can be written as:

$$\frac{\partial C_2}{\partial t} = bF_{\mathrm{d}} C_1, \tag{B.101}$$

and the one describing the evolution of the chemical species is of the form:

$$\frac{\partial C_1'}{\partial t} + \boldsymbol{\nabla} \cdot \boldsymbol{p}' = -\alpha F_{\mathrm{c}} C_1, \tag{B.102}$$

where $\boldsymbol{p}$ is the flux term:

$$\boldsymbol{p}' = -K \boldsymbol{\nabla} C_1', \tag{B.103}$$

and Π_{c} is a correction function:

$$\Pi_{\mathrm{c}}(C_1') = \frac{C_1'}{C_1' + \kappa}. \tag{B.104}$$

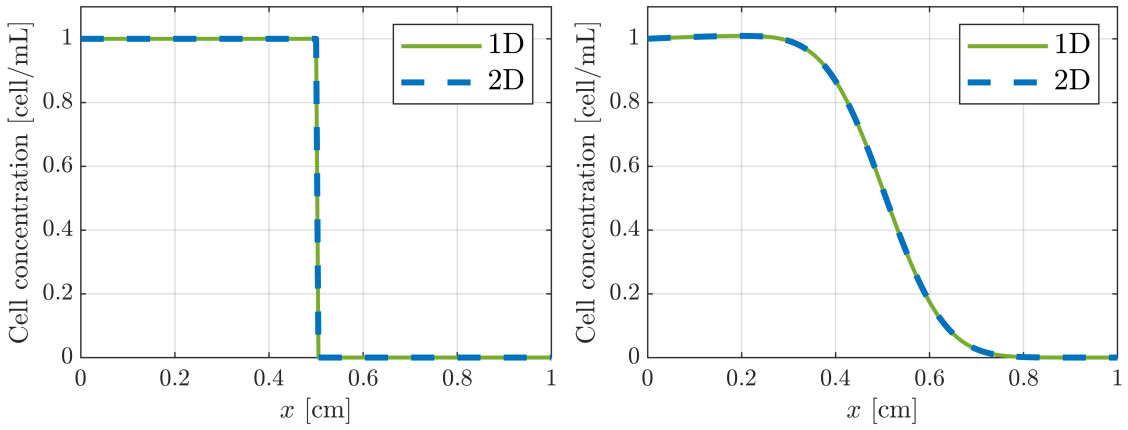
The equations and functions described above include a number of parameters, whose value is defined (with illustrative purposes) in Table B.1.

Parameter	Value	Units
a	1	s^{-1}
b	0.05	s^{-1}
K_D	0.1	cm^2/s
K_{go}	0.1	$\mathrm{cm}^2/(\mathrm{mmHg} \cdot \mathrm{s})$
K	0.1	cm^2/s
β	1	mmHg
C^*	5	cell/mL
γ	1	mmHg
$\Delta\gamma$	10	mmHg
α	1	mmHg
κ	2.5	mmHg

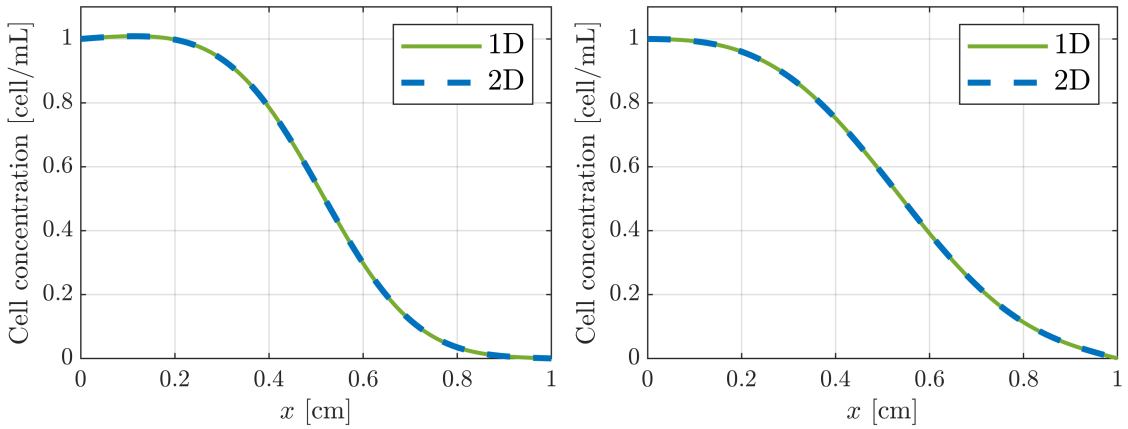
Table B.1: Parameters used in the complete validation problem.

The simulation results for both one-dimensional and two-dimensional problems are represented in Figs. B.12 (for the alive cell phenotype C_1), B.13 (for the dead phenotype C_2) and B.14 (for the chemical species C_1'), showing good agreement in all the cases.

To provide a two-dimensional visualisation of the results, the two-dimensional concentration profiles for the three fields studied (C,C_d and S) are represented for the final simulation time ($t = 0.25$ s) in Figures B.15, B.16 and B.17 respectively.

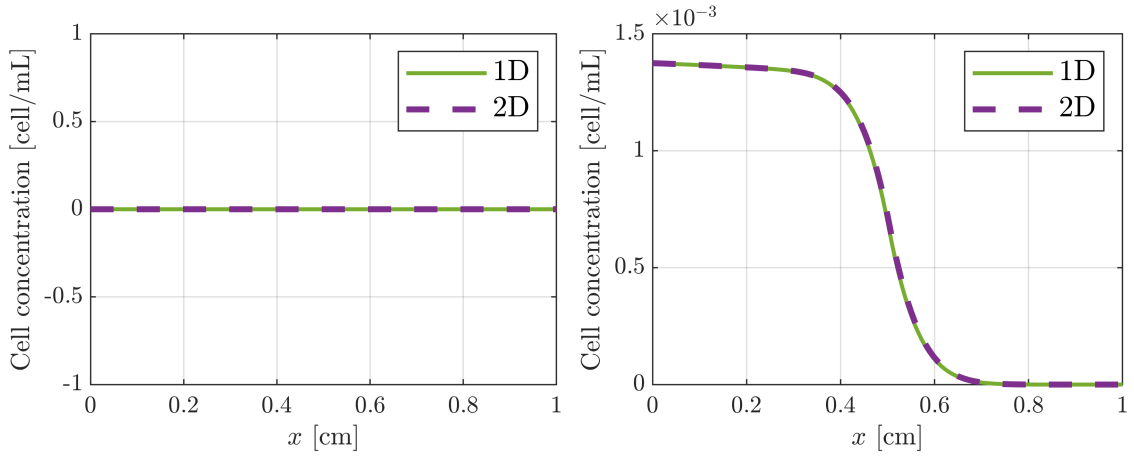


(a) $t = 0$ s. **(b)** $t = 0.05$ s.



(c) $t = 0.125$ s. **(d)** $t = 0.25$ s.

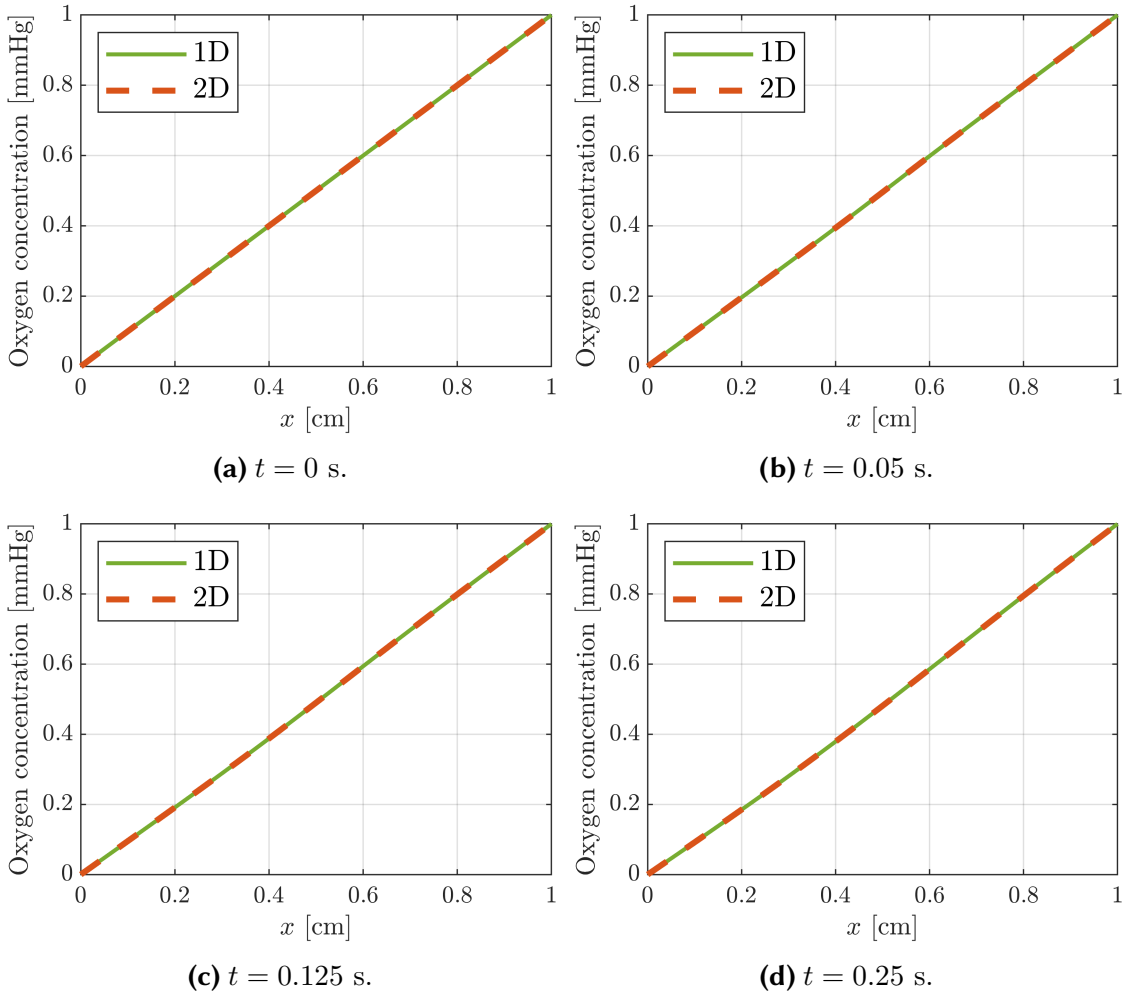
Figure B.12: Alive cells concentration profiles obtained with both 1D and 2D models for the complete problem validation.



(a) $t = 0$ s. **(b)** $t = 0.05$ s.

(c) $t = 0.125$ s. **(d)** $t = 0.25$ s.

Figure B.13: Dead cells concentration profiles obtained with both 1D and 2D models for the complete problem validation.



(a) $t = 0$ s. **(b)** $t = 0.05$ s.

(c) $t = 0.125$ s. **(d)** $t = 0.25$ s.

Figure B.14: Oxygen concentration profiles obtained with both 1D and 2D models for the complete problem validation.

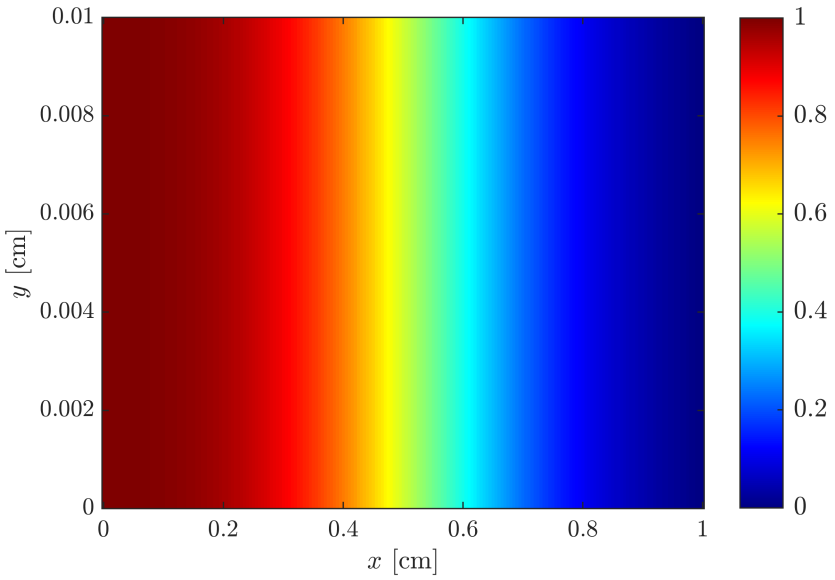


Figure B.15: Two-dimensional alive cells concentration field for $t = 0.25$ s.

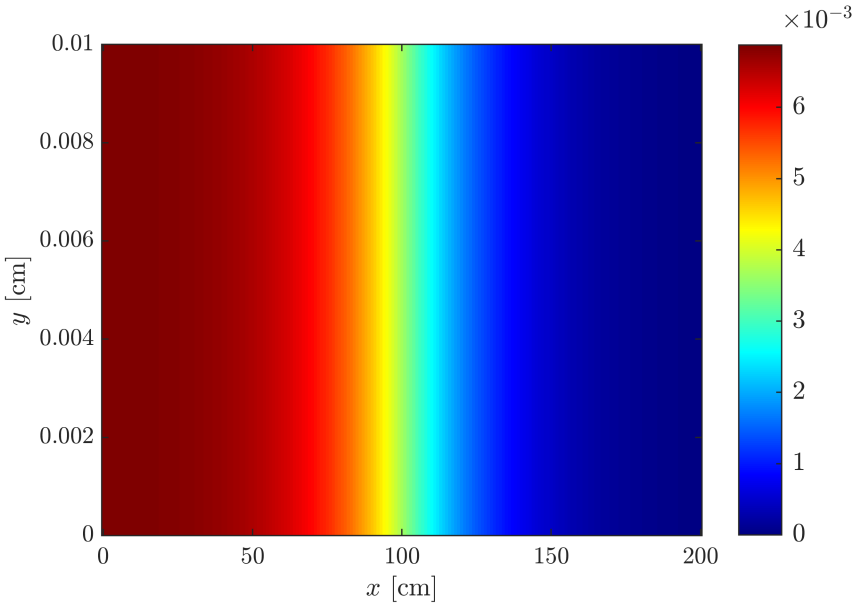


Figure B.16: Two-dimensional dead cells concentration field for $t = 0.25$ s.

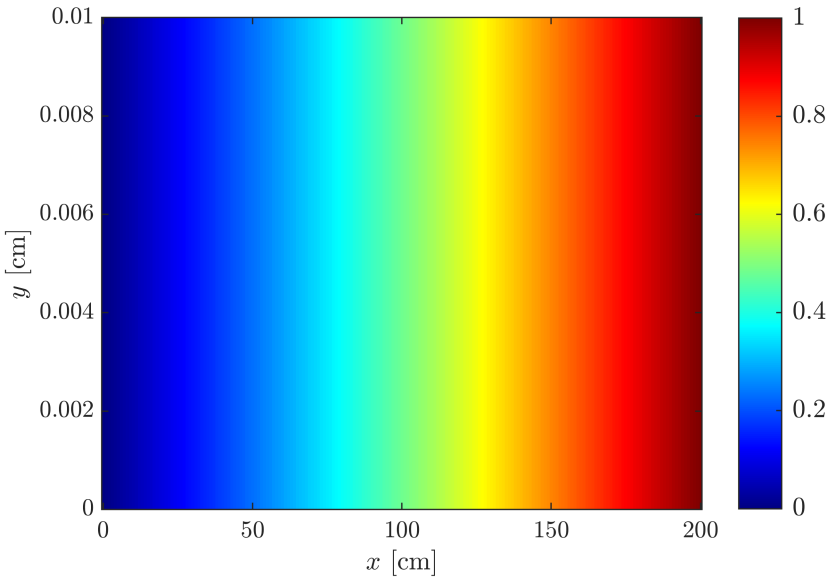


Figure B.17: Two-dimensional oxygen concentration field for $t = 0.25$ s.

B.3.2 Three-dimensional problem validation

In this section the same problem described in Section B.3.1 is going to be solved, but with a three-dimensional mesh. This mesh contains 400 eight-node hexahedral elements (making up a total of 909 nodes) with dimensions $0.01\,\text{cm} \times 0.15\,\text{cm} \times 0.15\,\text{cm}$. A scheme of the mesh can be seen in Figure B.18; where the nodes with Dirichlet boundary conditions have been marked with a red star and the nodes with a Neumann boundary condition have been marked with a blue dot.

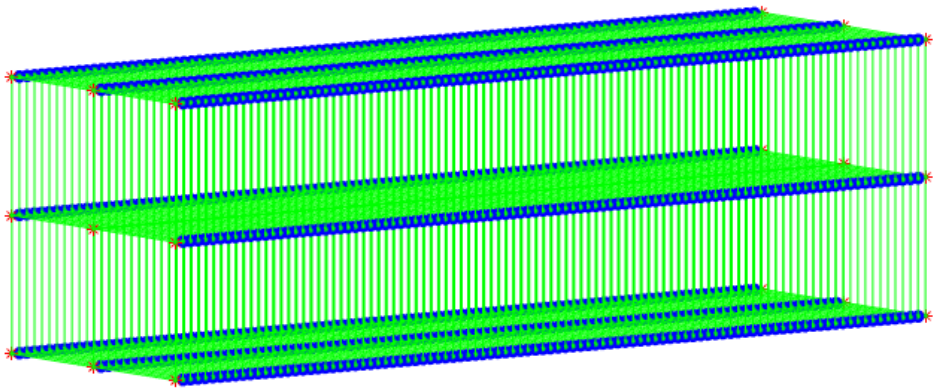


Figure B.18: Scheme of the three-dimensional mesh.

The results obtained for the central line of nodes are compared to the ones obtained with the one-dimensional simulations, and are shown in Figure B.19 for the alive cells, in Figure B.20 for the dead phenotype and in Figure B.21 for the chemical species. The results obtained for the three-dimensional model perfectly agree with the one-dimensional results.

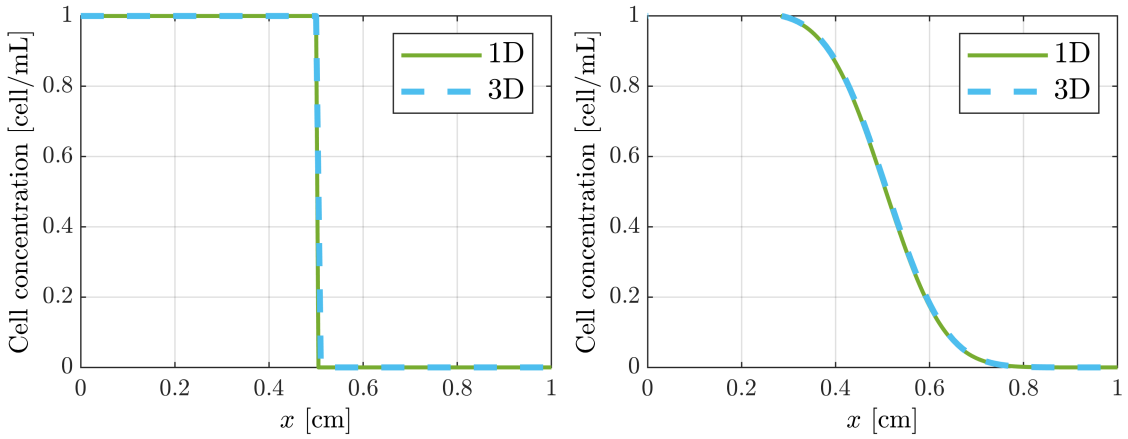
B.4 Integrator validation

Finally, in this section, the two-dimensional problem described in B.3.1.3 is solved using different integrators that are implemented in the MicroMiCo code. These integrators are:

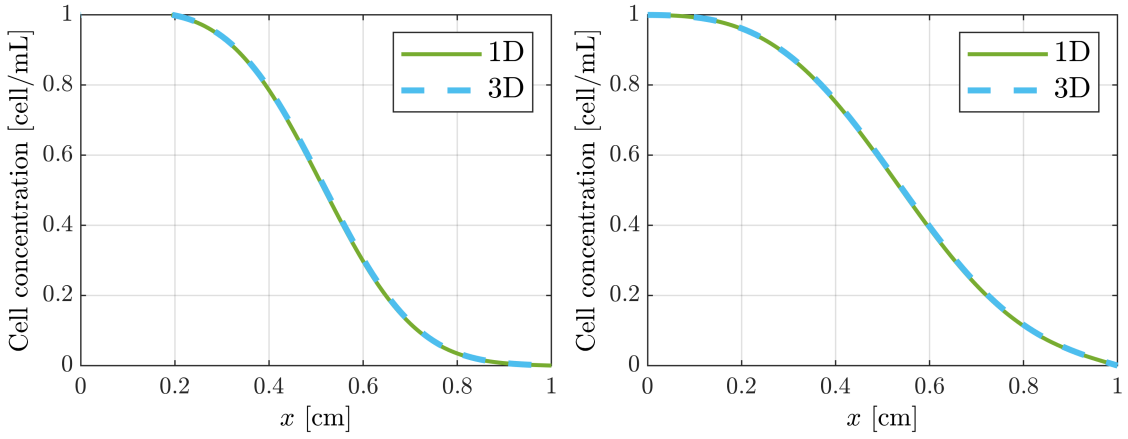
- Forward Euler method (explicit, with one step and one station) using a time step of $\Delta t = 1 \times 10^{-5}$ s.
- Midpoint method (Explicit, with one step and two stations) using a time step of $\Delta t = 1 \times 10^{-5}$ s.
- Backward Euler method combined with Broyden method (Implicit, with one step and one station) using a time step of $\Delta t = 1 \times 10^{-4}$ s.
- Adams-Bashforth method (Explicit, with two steps and one station) using a time step of $\Delta t = 1 \times 10^{-5}$ s.

For more accurate descriptions of each integration methods, the reader is referred to Butcher (2016).

The results provided by all of the previously mentioned integration methods, compared to the ones obtained when solving the one-dimensional problem are shown for the



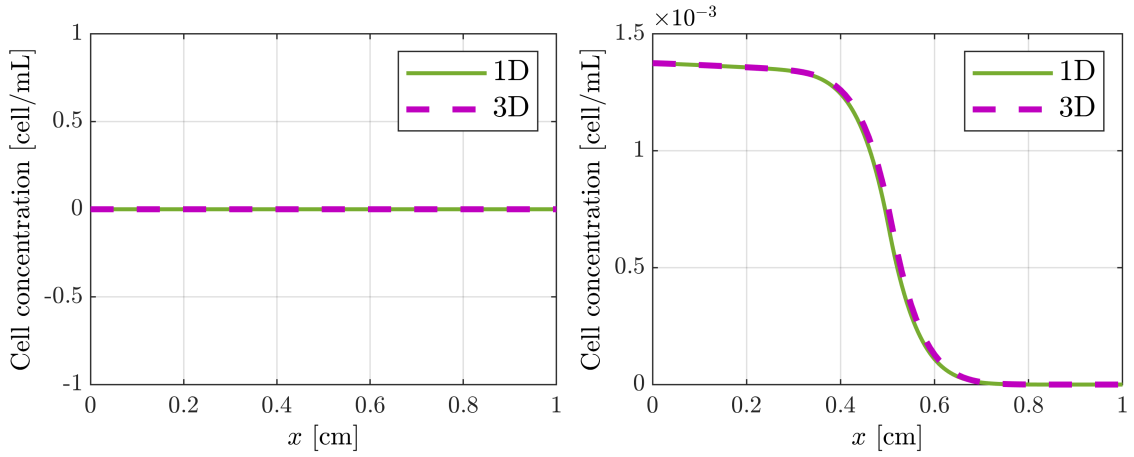
(a) $t = 0$ s. **(b)** $t = 0.05$ s.



(c) $t = 0.125$ s. **(d)** $t = 0.25$ s.

Figure B.19: Alive cells concentration profiles obtained with both 1D and 3D models for the complete problem.

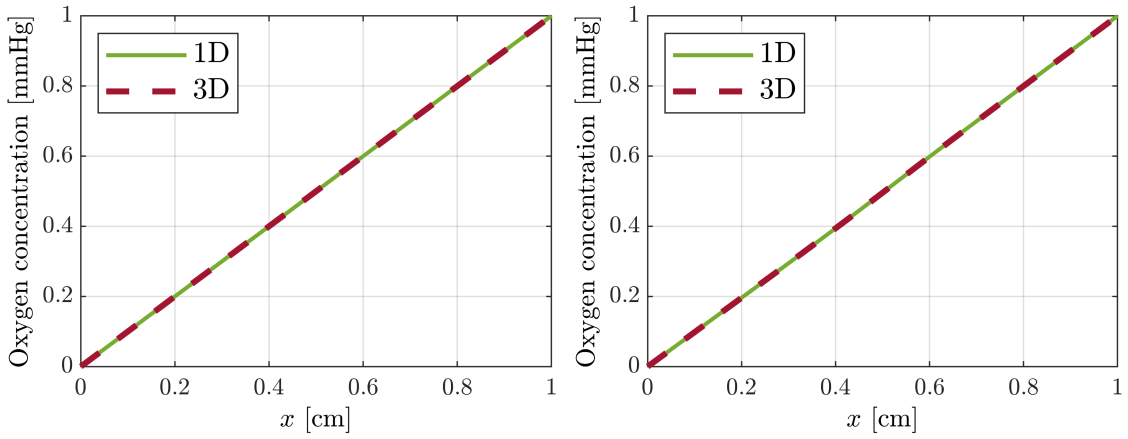
final simulation time ($t = 0.25$ s) in Figure B.22 for the three populations considered (alive and dead cell phenotypes and the chemical species respectively). All integrators provide accurate results.



(a) $t = 0$ s. **(b)** $t = 0.05$ s.

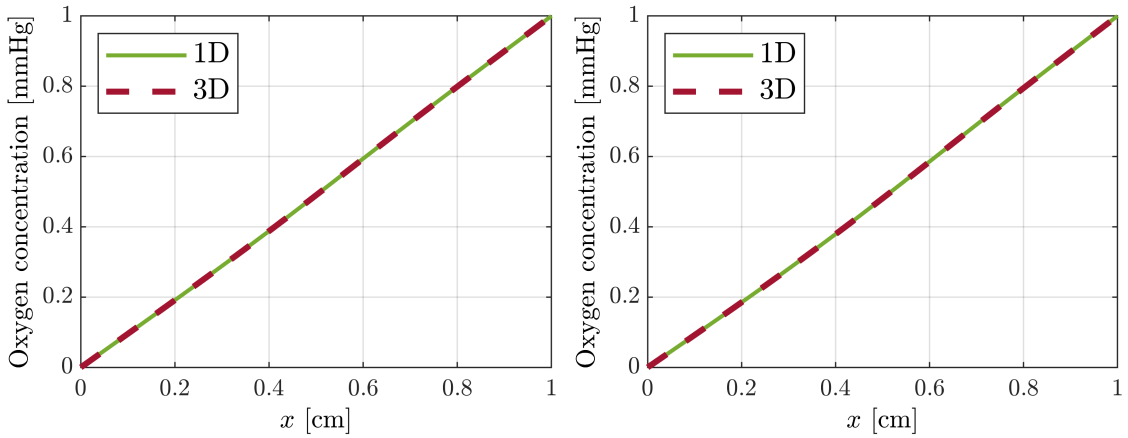
(c) $t = 0.125$ s. **(d)** $t = 0.25$ s.

Figure B.20: Dead cells concentration profiles obtained with both 1D and 3D models for the complete problem.



(a) $t = 0$ s.

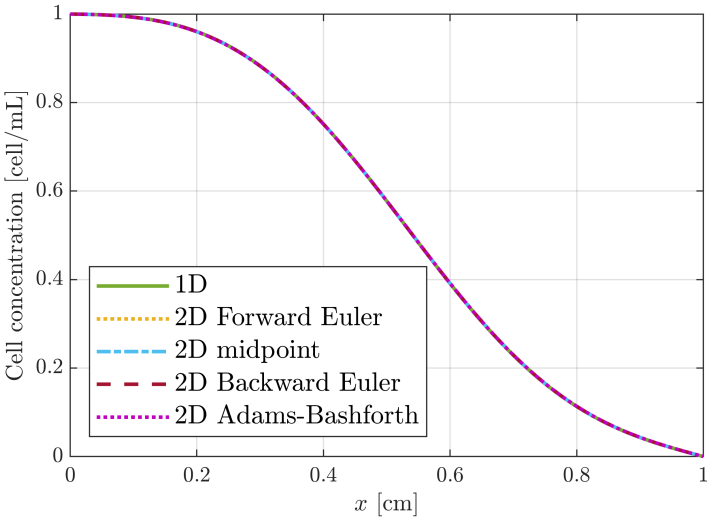
(b) $t = 0.05$ s.



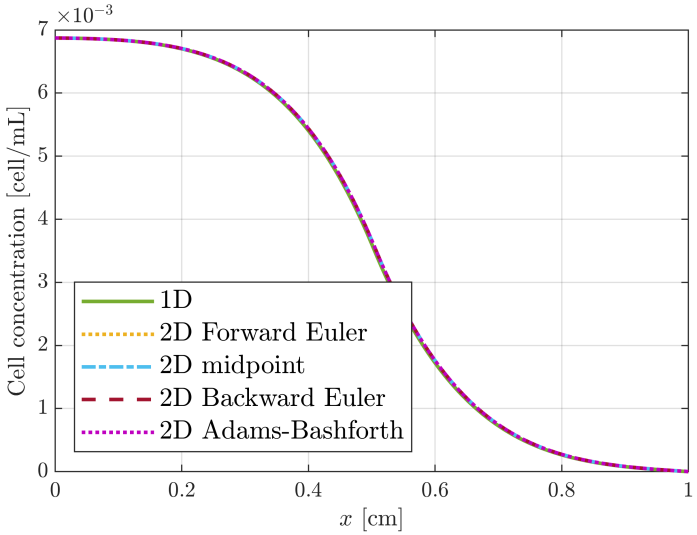
(c) $t = 0.125$ s.

(d) $t = 0.25$ s.

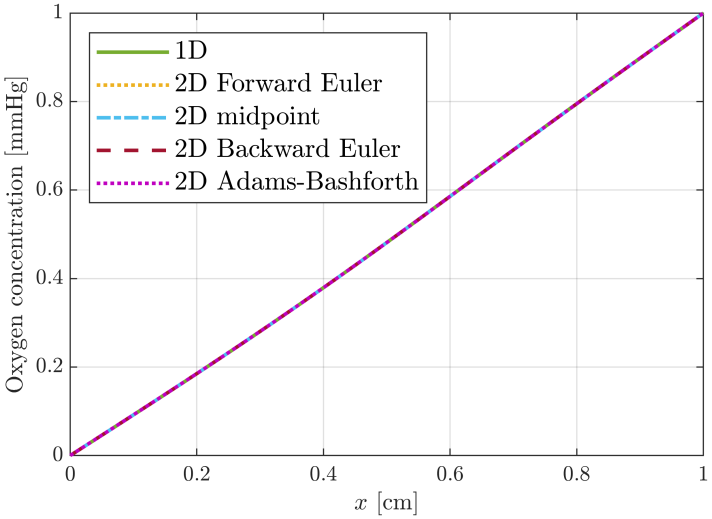
Figure B.21: Oxygen concentration profiles obtained with both 1D and 3D models for the complete problem.



(a) Alive cell phenotype C_1.



(b) Dead cell phenotype C_2.



(c) Chemical species C_1'.

Figure B.22: Results obtained using different integrators.

B.5 Test on a 3D microdevice geometry

In order to illustrate the kind of situations that our computational code is able to simulate, we are going to consider a "toy model" representative of the problem in hands, that is, cell culture evolution in microfluidic devices. When a cell culture is going to be seeded on a chip, the objective is to reproduce, in the microfluidic device, the desired tumour micro-environment (TME), that is, appropriate mechanical properties, nutrient and oxygen supplies, and gradients. The goal is to design properly the experiment in order to take the suitable conclusions in a reasonable time. For that, it is necessary to define a precise device geometry and boundary conditions so that cell cultures are subjected to the desired chemical and mechanical stimuli.

In order to illustrate all this, a simulation in a 3D chip-like geometry is going to be presented for a very fundamental model. For a more accurate model, depicting glioblastoma (GBM) cell progression in microfluidic devices, the reader is addressed to Chapter 6.

Two cell phenotypes are going to be considered, alive and dead cells, C_n and C_d, respectively, and the oxygen O_2 is going to be the driving chemoattractant.

B.5.1 Set-up

Model and parameters

Equations of the model are the following ones:

$$\frac{\partial C_n}{\partial t} = D_n \boldsymbol{\nabla}^2 C_n - \chi \boldsymbol{\nabla} \cdot (C_n \boldsymbol{\nabla} O_2) + \frac{1}{\tau_{\mathrm{gr}}} \left(1 - \frac{C_n}{C_{\mathrm{sat}}}\right) C_n - \frac{1}{\tau_{\mathrm{d}}} \Pi_{\mathrm{d}}(O_2) C_n, \quad \text{(B.105a)}$$

$$\frac{\partial C_d}{\partial t} = \frac{1}{\tau_{\mathrm{d}}} H\left(O_2^A - O_2\right) C_n, \quad \text{(B.105b)}$$

$$\frac{\partial O_2}{\partial t} = D_{O_2} \boldsymbol{\nabla}^2 O_2 - \alpha \left(\frac{O_2}{O_2 + k_{\mathrm{m}}}\right) C_n. \quad \text{(B.105c)}$$

Here, D_n is diffusion coefficient of normoxic phenotype, χ is the chemotaxis coefficient, τ_{gr} and τ_{d} are the growth and death characteristic times, C_{sat} is the cell capacity, α is the oxygen consumption and k_{m} is the Michaelis-Menten constant of cellular respiration kinetics. The function H is the Heaviside step function, that is $H(O_2^A - O_2) = 1$ when $O_2 \leq O_2^A$ and $H(O_2^A - O_2) = 0$ when $O_2 > O_2^A$. The presented toy model is a particular case of the general case given by Eqs. (B.1) where $n = 2$, $m = 1$, $C_1 = C_n$, $C_2 = C_d$, $C_1' = O_2$ and:

- $\boldsymbol{v} = \mathbf{0}$, that is, there is no fluid flow.
- $F_1 = \frac{1}{\tau_{\mathrm{gr}}} \left(1 - \frac{C_n}{C_{\mathrm{sat}}}\right)$ (logistic growth model), $F_{12} = \frac{1}{\tau_{\mathrm{d}}} H(O_2^A - C_1')$ (anoxia mediated death), $F_2 = 0$ and $F_{21} = 0$ (dead cells considered as an inert population).
- $F_{11}' = -\alpha \frac{C_1'}{C_1' + k_{\mathrm{m}}}$, that is, oxygen consumption is assumed to follow Michaelis-Menten kinetics.
- $\boldsymbol{q}_1 = -D_n \boldsymbol{\nabla} C_1 + \chi C_1 \boldsymbol{\nabla} C_1'$, that is, we consider a pedesis term and an oxygen-mediated chemotaxis term for the normoxic phenotype.

Symbol	Meaning	Value
D_{O_2}	Oxygen diffusion	$5.0 \cdot 10^{-5}\,\mathrm{cm}^2/\mathrm{s}$
D_n	Cell diffusion	$3.3 \cdot 10^{-6}\,\mathrm{cm}^2/\mathrm{s}$
C_{sat}	Cell capacity	$5.0 \cdot 10^{7}\,\mathrm{cell/mL}$
χ	Chemotaxis coefficient	$3.8 \cdot 10^{-5}\,\mathrm{cm}^2/(\mathrm{mmHg} \cdot \mathrm{s})$
τ_{gr}	Growth characteristic time	4 h
τ_{d}	Death characteristic time	0.8 h
α	Oxygen consumption rate	$5.0 \cdot 10^{-8}\,\mathrm{cm}^3 \cdot \mathrm{mmHg}/(\mathrm{cell} \cdot \mathrm{s})$
k_{m}	Michaelis-Menten constant	2.5 mmHg
O_2^A	Anoxia threshold	1.6 mmHg

Table B.2: Parameters of the toy model.

- $\boldsymbol{q}_2 = \mathbf{0}$, that is, necrotic phenotype (dead cells) are considered as an inert population.
- $\boldsymbol{q}_1' = -D_n \boldsymbol{\nabla} C_1$.

In Table B.2 the values of the parameters selected for the illustrative simulation are listed.

Geometry

Fig. B.23 shows the geometry and the mesh of the culture chamber of a microfluidic device. Geometry and dimensions are representative and of the order of hundreds of micrometeres. The geometry presented has a respective maximum width, length, and height of 600 μm, 600 μm, and 70 μm, respectively.

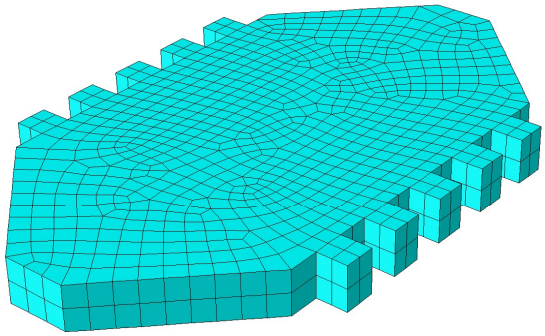


Figure B.23: Geometry and mesh of the culture chamber of a microfluidic device. We have considered a hexahedral mesh with two elements along the microdevice thickness.

Boundary conditions

All faces of the chip are considered as impermeable to chemical species and cells except for the supply surfaces, which are marked in red in Fig. B.24. Therefore, boundary

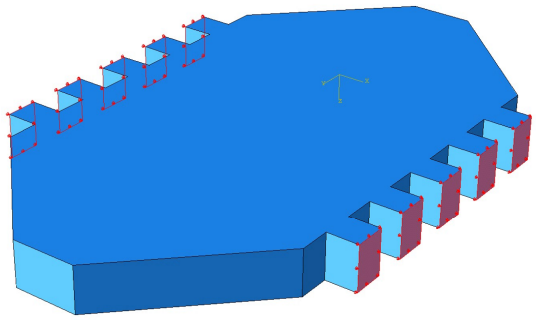


Figure B.24: Boundary conditions. Surfaces where Dirichlet boundary conditions are applied are marked in red. For the rest of the boundary, homogeneous Neumann boundary conditions are applied.

conditions are of Neumann type in the rest of the contour of the chamber such that cell and oxygen flux is specified to be 0. In the red-marked surfaces, we assume Dirichlet boundary conditions. As an example, suppose that cell concentration is set to 1×10^6 cell/mL and 0 for alive and dead cell phenotypes respectively and oxygen supply is fixed to $O_2^s = 2\,\text{mmHg}$.

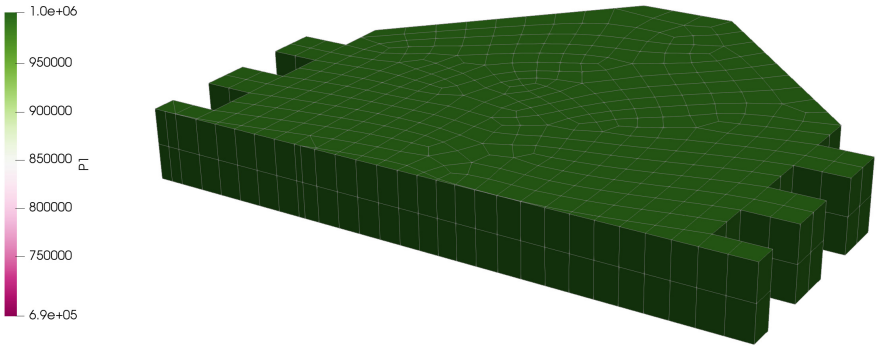
Initial conditions

As initial conditions, we assume that, at the beginning, there are no dead cells at the culture chamber and the concentration of alive cells is homogeneous and equal to $C^0 = 1 \cdot 10^6\,\text{cell/mL}$. Finally, it is assumed that oxygen pressure is homogeneous at $t = 0$ and equal to $O_2^0 = 2\,\text{mmHg}$.

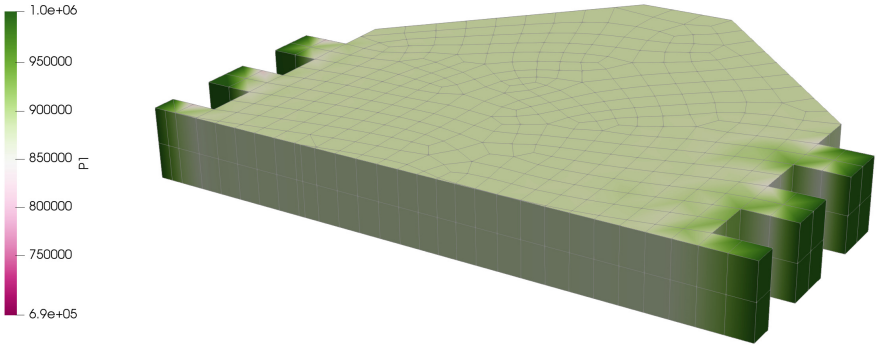
B.5.2 Results

Fig. B.25 shows the evolution over time of alive cells and Fig. B.26 the evolution of dead cells on the culture chamber. As expected, cell concentration remains high next to the supply channels and decreases in the central part of the chamber, where oxygen consumption induces anoxia. Therefore, once the oxygen threshold of $O_2^A = 1.6\,\text{mmHg}$ is achieved, cell death is promoted. This explains, analogously, why dead cell concentration increases in the same regions. However, cell chemotaxis explains why alive cells are even more concentrated at oxygen supply points: cells migrate in the direction of the oxygen gradient and when they arrive to a well oxygenated point, proliferation occurs normally, because the conditions are now favourable and cell concentration is below the capacity limit.

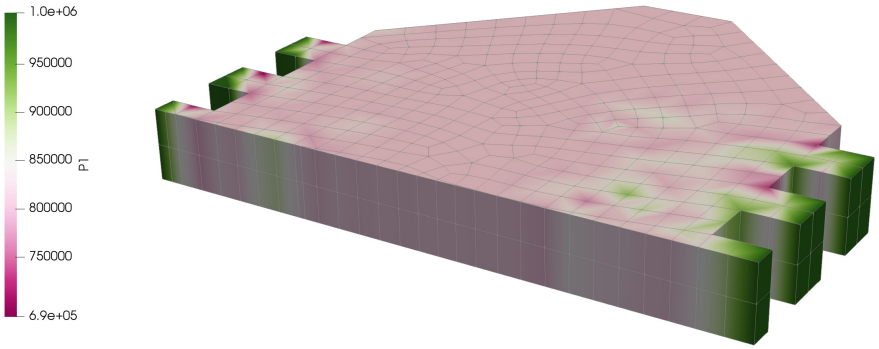
The results of the simulation show how the cell culture is going to evolve during the virtual experiment. With the presented parameters and boundary conditions, the lack of oxygen diffusion along the culture chamber results in the fast appearance of a necrotic core occupying almost the entire chamber. This kind of *in silico* predictions can be extrapolated to more complex evolution models, cell populations and tissues, other geometries and other mechanical frameworks relatives to the experiment. Moreover, the simulation of the different processes allows the access to all the values



(a) $t = 0\,\mathrm{s}$

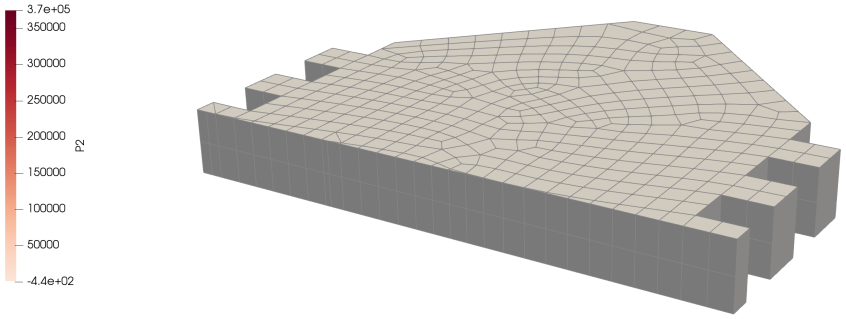


(b) $t = 42\,\mathrm{h}$

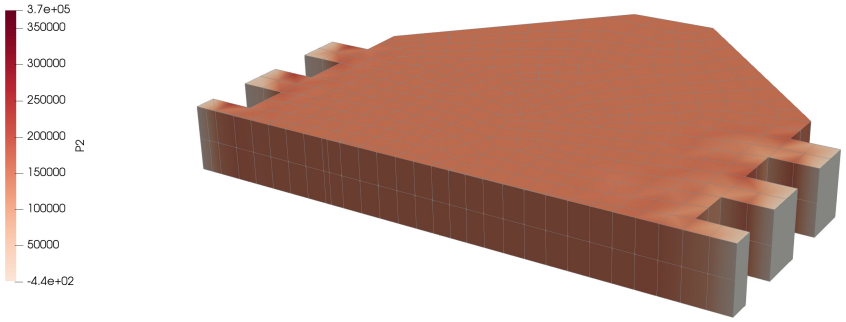


(c) $t = 70\,\mathrm{h}$

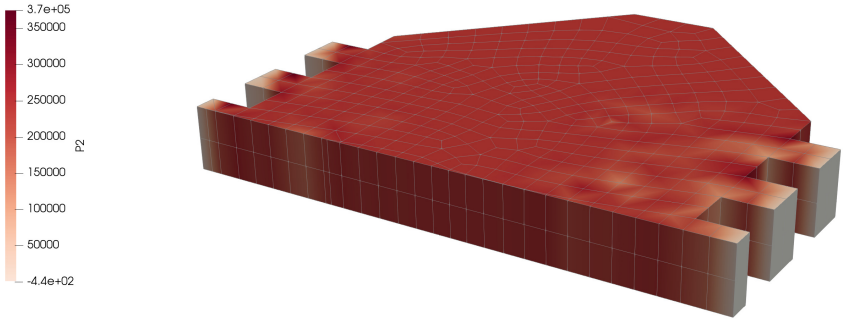
Figure B.25: Evolution of alive cells in the culture chamber. Cell concentration is computed in cell/mL.



(a) $t = 0\,\mathrm{s}$



(b) $t = 42\,\mathrm{h}$



(c) $t = 70\,\mathrm{h}$

Figure B.26: Evolution of dead cells in the culture chamber. Cell concentration is computed in cell/mL.

of field variables, which can, in turn, be interesting to identify correlations between phenomena or variables of clinical or physiological interest, that would be inaccessible from an experimental point of view due to technical considerations (difficulty or impossibility of field variables monitorisation).

As a final remark, we have to state that model definition and parameter fitting is a very complicated task. Even if frequently simplified, the multiphysics nature of the TME is very complex: many different phenomena are coupled and many scales are involved, resulting in a hard nonlinear problem where even the semiquantitative analysis is often complicated. Too simplistic models lead to the failure of predictive simulation models while complex and sophisticated ones result in a difficult parameter estimation (due to both numerical and experimental difficulties and validation). Moreover, highly nonlinear and coupled models in different physical scales may involve very expensive simulations from the computational point of view.

C. Scientific dissemination

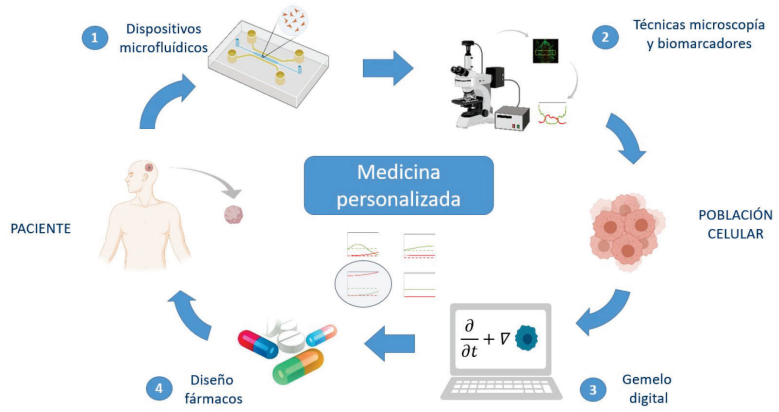
It is said that the work of a scientist is not only to investigate, but also to disseminate the results of their research to the general public. In that sense, I was given the opportunity to write an article at the local newspaper "Heraldo de Aragón", in which I summarised the overview and main results of the thesis. This article was published on December 21$^{\text{th}}$, and is reproduced in the next page.

DIVULGA TU TESIS Con la colaboración de la Unidad de Cultura Científica de la Universidad de Zaragoza

ABRIENDO LA CAJA NEGRA DEL CÁNCER

>IA Y FÍSICA CONTRA EL GLIOBLASTOMA

El glioblastoma es el tumor cerebral más común y letal. En los últimos años se ha avanzado en su conocimiento, pero el número de fármacos aprobados se ha estancado y la supervivencia de los pacientes no aumenta. Nuevas técnicas de cultivo celular permiten recrear el cerebro en condiciones más realistas y generan gran cantidad de datos, lo que abre la puerta a usar inteligencia artificial (IA) para responder a nuevas preguntas. ¿Cómo incorporar lo que ya sabemos sobre el glioblastoma? Haciendo que la física guíe los algoritmos



La combinación de técnicas avanzadas de cultivo celular, biomarcadores y simulación computacional será fundamental para la medicina personalizada del futuro. UNIZAR

DATOS Y ECUACIONES Hoy en día, y cada vez más, generamos una cantidad ingente de imágenes, textos y contenido audiovisual. Esta información es procesada por millones de ordenadores que, como respuesta, nos recomiendan películas, ofertan productos, calculan rutas óptimas o ayudan a los médicos a diagnosticar una enfermedad. Esta capacidad para transformar datos brutos en información verdaderamente relevante es mérito de los famosos algoritmos de inteligencia artificial. Así la inteligencia artificial ha entrado de lleno en ámbitos como el cuidado de la salud, las ciencias sociales y el comercio electrónico, no sin exponerse a cierto escepticismo por parte de algunos expertos.

En paralelo, y con el auge de la supercomputación, cada vez son más numerosos los modelos computacionales que recrean por ordenador la realidad. Son los llamados modelos 'in silico', por oposición a los modelos 'in vivo' (en animales o personas) o 'in vitro' (en laboratorios). Con modelos de este tipo se ha podido recrear la progresión de muchos tumores, como por ejemplo el glioblastoma, un tumor cerebral de muy mal pronóstico (14 meses de supervivencia) y difícil tratamiento (la recurrencia es casi inevitable).

Con el desarrollo de la tecnología microfluídica, que consiste en cultivar células en entornos biomiméticos de una forma automatizada y generando muchos datos, comienza a ser posible utilizar las herramientas de la inteligencia artificial en el ámbito de los cultivos celulares, para desentrañar los complejos cambios metabólicos que se producen como respuesta a fármacos o estímulos externos.

Combinar estas herramientas de inteligencia artificial con los modelos matemáticos de evolución de tumores ayuda enormemente a explicar fenómenos desconocidos y predecir mejor la progresión tumoral, avanzando hacia la medicina personalizada.

LA TESIS

- **Trabajo** Tesis doctoral realizada en el grupo Tissue MicroEnvironment Lab (TMELab), en el Instituto de Investigación en Ingeniería de Aragón (I3A) e Instituto de Investigación Sanitaria de Aragón (IIS).
- **Autor** Jacobo Ayensa Jiménez, matemático e Ingeniero de caminos, doctorando del programa de Ingeniería Mecánica de la Universidad de Zaragoza.
- **Directores** Manuel Doblaré y Mohamed H. Doweidar.

¿POR QUÉ INTELIGENCIA ARTIFICIAL GUIADA POR FÍSICA?

La inteligencia artificial está cambiando la forma que tenemos de entender la ciencia. En los últimos años, los modelos científicos sobre el universo han sido paulatinamente sustituidos por inteligencias artificiales capaces de generar patrones complejos de datos desestructurados, es decir, capaces de transformar la información en conocimiento. Sin embargo, esto arroja inquietantes preguntas. ¿Vamos a sustituir el conocimiento científico adquirido a lo largo de la humanidad por las cajas negras de la inteligencia artificial? ¿De verdad vamos a olvidarnos del legado de genios como Newton y Einstein y sustituirlos por inteligencias artificiales? Muy recientemente, ha empezado a surgir una corriente crítica hacia la inteligencia artificial, que clama por una mayor interpretabilidad de los modelos y que se hace eco de los fracasos de la inteligencia artificial en diversos ámbitos (salud, conducción autónoma, reconocimiento de imagen...). La forma de acabar con este escepticismo es dar con métodos más interpretables, introduciendo directamente en los cálculos la física del mundo que nos rodea para poder entender mejor las predicciones. Además, así no se pierde el foco, y la inteligencia artificial guiada por física, además de desentrañar lo que aún no conocemos del universo, aprende más rápido, necesita menos datos y filtra mejor el ruido que la inteligencia artificial convencional.

¿CÓMO SE TRATARÁ A LOS PACIENTES EN EL FUTURO?

Gracias a la combinación de los dispositivos microfluídicos, la monitorización de cultivos usando biomarcadores y las técnicas de inteligencia artificial guiada por física, en el futuro la medicina tendrá un enfoque personalizado. Un paciente llegará a una clínica y se le extraerá una muestra de tumor, que se cultivará en un dispositivo y se evaluará cómo responde a estímulos externos. Mediante las herramientas de inteligencia artificial guiada por física, se creará un gemelo virtual del tumor de ese paciente en concreto. Utilizando simulaciones por ordenador, se podrá evaluar, sin necesidad de hacer experimentos y esperar un tiempo de prueba, qué efectos tendría sobre ese paciente en concreto probar un determinado fármaco o una determinada terapia y elegir el que mejor resultados dé. Con simulaciones virtuales se pueden hacer pruebas sin que estas duren semanas, y el tiempo es lo más valioso para el paciente.

JACOBO AYENSA JIMÉNEZ MATEMÁTICO E INGENIERO DE CAMINOS, DOCTORANDO DEL PROGRAMA DE INGENIERÍA MECÁNICA DE LA UNIVERSIDAD DE ZARAGOZA

¿PUEDE ESTE ENFOQUE RECONCILIAR EMPIRISMO Y RACIONALISMO?

Desde el nacimiento de la ciencia moderna y del método científico, dos grandes tendencias han dividido a los grandes pensadores. El racionalismo, cuyo máximo exponente es René Descartes, acentúa el papel de la razón en la construcción del conocimiento, construye teorías para luego validarlas por observación. El empirismo, que tiene en David Hume a su cara más visible, fía el conocimiento en primera instancia a la pura observación del entorno y la naturaleza. Ninguna de las dos corrientes ha desplazado a la otra. De hecho, Newton (racionalista) y Kepler (empirista) llegaron a la deducción del movimiento planetario por vías diferentes. Con las herramientas de inteligencia artificial guiada por física, los datos pueden combinarse con sólidas teorías científicas, para enriquecerlas. Tener una inteligencia artificial guiada por física es como sentar en una mesa a Newton y Kepler y hacerles colaborar.

Bibliography

Lucie P Aarts and Peter Van Der Veer. Neural network method for solving partial differential equations. *Neural Processing Letters*, 14(3):261–271, 2001. (cited on Page 43)

Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, et al. Tensorflow: Large-scale machine learning on heterogeneous distributed systems. *arXiv preprint arXiv:1603.04467*, 2016. (cited on Page 167)

Hervé Abdi and Lynne J Williams. Principal component analysis. *Wiley interdisciplinary reviews: computational statistics*, 2(4):433–459, 2010. (cited on Page 34)

Diab W Abueidda, Mohammad Almasri, Rami Ammourah, Umberto Ravaioli, Iwona M Jasiuk, and Nahil A Sobh. Prediction and optimization of mechanical properties of composites using convolutional neural networks. *Composite Structures*, 227:111264, 2019. (cited on Page 36)

Diab W Abueidda, Qiyue Lu, and Seid Koric. Deep learning collocation method for solid mechanics: Linear elasticity, hyperelasticity, and plasticity as examples. *arXiv e-prints*, pages arXiv–2012, 2020. (cited on Page 48)

Diab W Abueidda, Seid Koric, Nahil A Sobh, and Huseyin Sehitoglu. Deep learning for plasticity and thermo-viscoplasticity. *International Journal of Plasticity*, 136: 102852, 2021. (cited on Page 36)

Amina Adadi and Mohammed Berrada. Peeking inside the black-box: a survey on explainable artificial intelligence (xai). *IEEE access*, 6:52138–52160, 2018. (cited on Page 12 and 155)

Jill U Adams. Genetics: big hopes for big data. *Nature*, 527(7578):S108–S109, 2015. (cited on Page 10)

Franz Adler. Yates' correction and the statisticians. *Journal of the American Statistical Association*, 46(256):490–501, 1951. (cited on Page 272)

Abramo Agosti, Clara Cattaneo, Chiara Giverso, Davide Ambrosi, and Pasquale Ciarletta. A computational framework for the personalized clinical treatment of glioblastoma multiforme. *ZAMM-Journal of Applied Mathematics and Mechanics/Zeitschrift für Angewandte Mathematik und Mechanik*, 2018a. (cited on Page 258, 259, 260, 261, and 262)

Abramo Agosti, Chiara Giverso, Elena Faggiano, Aymeric Stamm, and Pasquale Ciarletta. A personalized mathematical tool for neuro-oncology: A clinical case study. *International Journal of Non-Linear Mechanics*, 2018b. (cited on Page 259, 260, 261, and 262)

Ankit Agrawal and Alok Choudhary. Perspective: Materials informatics and big data: Realization of the "fourth paradigm" of science in materials science. *Apl Materials*, 4(5):053208, 2016. (cited on Page 57)

Nabil Ali Ahmed and DV Gokhale. Entropy expressions and their estimators for multivariate distributions. *IEEE Transactions on Information Theory*, 35(3): 688–692, 1989. (cited on Page 247)

Hirotogu Akaike. Information theory and an extension of the maximum likelihood principle. In *Selected papers of hirotugu akaike*, pages 199–213. Springer, 1998. (cited on Page 52)

MS Al-Haik, H Garmestani, and I Michael Navon. Truncated-newton training algorithm for neurocomputational viscoplastic model. *Computer methods in applied mechanics and engineering*, 192(19):2249–2267, 2003. (cited on Page 36)

MS Al-Haik, MY Hussaini, and H Garmestani. Prediction of nonlinear viscoelastic behavior of polymeric composites using an artificial neural network. *International journal of plasticity*, 22(7):1367–1392, 2006. (cited on Page 36)

Mark Alber, Adrian Buganza Tepole, William R Cannon, Suvranu De, Salvador Dura-Bernal, Krishna Garikipati, George Karniadakis, William W Lytton, Paris Perdikaris, Linda Petzold, et al. Integrating machine learning and multiscale modeling—perspectives, challenges, and opportunities in the biological, biomedical, and behavioral sciences. *NPJ digital medicine*, 2(1):1–11, 2019. (cited on Page 37)

Christopher G Albert and Katharina Rath. Gaussian process regression for data fulfilling linear differential equations with localized sources. *Entropy*, 22(2):152, 2020. (cited on Page 39)

B Alekya, Sanjay Rao, and Hardik J Pandya. Engineering approaches for characterizing soft tissue mechanical properties: A review. *Clinical Biomechanics*, 69: 127–140, 2019. (cited on Page 36)

Icíar Alfaro, David González, Sergio Zlotnik, Pedro Díez, Elías Cueto, and Francisco Chinesta. An error estimator for real-time simulators based on model order reduction. *Advanced Modeling and Simulation in Engineering Sciences*, 2(1):1–16, 2015. (cited on Page 40)

Usman Ali, Waqas Muhammad, Abhijit Brahme, Oxana Skiba, and Kaan Inal. Application of artificial neural networks in micromechanics for polycrystalline metals. *International Journal of Plasticity*, 120:205–219, 2019. (cited on Page 36)

Cyrille Allery, Aziz Hamdouni, David Ryckelynck, and N Verdon. A priori reduction method for solving the two-dimensional burgers' equations. *Applied Mathematics and Computation*, 217(15):6671–6679, 2011. (cited on Page 40)

Hasan Alli, Ayşegül Uçar, and Yakup Demir. The solutions of vibration control problems using artificial neural networks. *Journal of the Franklin Institute*, 340(5): 307–325, 2003. (cited on Page 47)

Paul D Allison. Missing data: Sage university papers series on quantitative applications in the social sciences (07–136). *Thousand Oaks, CA*, 2001. (cited on Page 102)

Paul D Allison. Missing data: Quantitative applications in the social sciences. *British Journal of Mathematical and Statistical Psychology*, 55(1):193–196, 2002. (cited on Page 102)

Paul D Allison. Missing data techniques for structural equation modeling. *Journal of abnormal psychology*, 112(4):545, 2003. (cited on Page 102)

Tikvah Alper and P Howard-Flanders. Role of oxygen in modifying the radiosensitivity of e. coli b. *Nature*, 178(4540):978–979, 1956. (cited on Page 262 and 296)

Philipp M Altrock, Lin L Liu, and Franziska Michor. The mathematics of cancer: integrating quantitative models. *Nature Reviews Cancer*, 15(12):730–745, 2015. (cited on Page 10 and 234)

Mauricio A Alvarez, David Luengo, and Neil D Lawrence. Linear latent force models using gaussian processes. *IEEE transactions on pattern analysis and machine intelligence*, 35(11):2693–2705, 2013. (cited on Page 37)

M Amir Siddiq. Data-driven finite element method: Theory and applications. *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, 235(17):3329–3339, 2021. (cited on Page 58)

Amine Ammar, Francisco Chinesta, Pedro Diez, and Antonio Huerta. An error estimator for separated representations of highly multidimensional models. *Computer Methods in Applied Mechanics and Engineering*, 199(25-28):1872–1880, 2010. (cited on Page 40)

Víctor Jesús Amores, José María Benítez, and Francisco Javier Montáns. Average-chain behavior of isotropic incompressible polymers obtained from macroscopic experimental data. a simple structure-based wypiwyg model in julia language. *Advances in Engineering Software*, 130:41–57, 2019. (cited on Page 164)

Carol S Aneshensel. *Theory-based data analysis for the social sciences*. Sage, 2013. (cited on Page 24)

Andrew Ang and Joseph Chen. Asymmetric correlations of equity portfolios. *Journal of financial Economics*, 63(3):443–494, 2002. (cited on Page 235)

Cosmin Anitescu, Elena Atroshchenko, Naif Alajlan, and Timon Rabczuk. Artificial neural network methods for the solution of second order boundary value problems. *Computers, Materials and Continua*, 59(1):345–359, 2019. (cited on Page 48)

Julia Ankudinova and Matthias Ehrhardt. On the numerical solution of nonlinear black–scholes equations. *Computers & Mathematics with Applications*, 56(3): 799–812, 2008. (cited on Page 204)

Syed Muhammad Anwar, Muhammad Majid, Adnan Qayyum, Muhammad Awais, Majdi Alnowami, and Muhammad Khurram Khan. Medical image analysis using convolutional neural networks: a review. *Journal of medical systems*, 42(11):226, 2018. (cited on Page 147)

Farid Khalil Arya and Lan Zhang. Copula-based markov process for forecasting and analyzing risk of water quality time series. *Journal of Hydrologic Engineering*, 22 (6):04017005, 2017. (cited on Page 235)

Panagiotis G Asteris and Vagelis Plevris. Anisotropic masonry failure criterion using artificial neural networks. *Neural Computing and Applications*, 28(8):2207–2229, 2017. (cited on Page 36)

Steven Atkinson and Nicholas Zabaras. Structured bayesian gaussian process latent variable model: Applications to data-driven dimensionality reduction and high-dimensional inversion. *Journal of Computational Physics*, 383:166–195, 2019. (cited on Page 37)

Luigi Atzori, Antonio Iera, and Giacomo Morabito. The internet of things: A survey. *Computer networks*, 54(15):2787–2805, 2010. (cited on Page 23)

Diane J Aum, David H Kim, Thomas L Beaumont, Eric C Leuthardt, Gavin P Dunn, and Albert H Kim. Molecular and cellular heterogeneity: the hallmark of glioblastoma. *Neurosurgical focus*, 37(6):E11, 2014. (cited on Page 285)

Jacobo Ayensa-Jiménez, Mohamed H. Doweidar, Jose A. Sanz-Herrera, and Manuel Doblare. A new reliability-based data-driven approach for noisy experimental data with physical constraints. *Computer Methods in Applied Mechanics and Engineering*, 328:752–774, 1 2018. doi: https://doi.org/10.1016/j.cma.2017.08.027. (cited on Page 20)

Jacobo Ayensa-Jiménez, Mohamed H Doweidar, Jose A Sanz-Herrera, and Manuel Doblaré. A new reliability-based data-driven approach for noisy experimental data with physical constraints. *Computer Methods in Applied Mechanics and Engineering*, 328:752–774, 2018. (cited on Page 17, 20, 102, 117, and 146)

Jacobo Ayensa-Jiménez, Mohamed H Doweidar, Teodora Randelovic, Luis J Fernández, Sara Oliván, Ignacio Ochoa, and Manuel Doblaré. On the simulation of organ-on-chip cell processes: Application to an in vitro model of glioblastoma evolution. In *Advances in Biomechanics and Tissue Regeneration*, pages 313–341. Elsevier, 2019a. (cited on Page 18 and 20)

Jacobo Ayensa-Jiménez, Mohamed H Doweidar, Jose A Sanz-Herrera, and Manuel Doblaré. An unsupervised data completion method for physically-based data-driven models. *Computer Methods in Applied Mechanics and Engineering*, 344:120–143, 2019b. (cited on Page 17)

Jacobo Ayensa-Jiménez, Mohamed H Doweidar, Jose A Sanz-Herrera, and Manuel Doblaré. On the application of physically-guided neural networks with internal variables to continuum problems. *arXiv preprint arXiv:2011.11376*, 2020a. (cited on Page 18, 20, 294, 299, and 308)

Jacobo Ayensa-Jiménez, Marina Pérez-Aliacar, Teodora Randelovic, Sara Oliván, Luis Fernández, José Antonio Sanz-Herrera, Ignacio Ochoa, Mohamed H Doweidar, and Manuel Doblaré. Mathematical formulation and parametric analysis of in vitro cell models in microfluidic devices: application to different stages of glioblastoma evolution. *Scientific Reports*, 10(1):1–21, 2020b. (cited on Page 18, 20, 248, 286, 293, 295, 299, 307, and 315)

Jacobo Ayensa-Jiménez, Mohamed H Doweidar, Jose A Sanz-Herrera, and Manuel Doblaré. Prediction and identification of physical systems by means of physically-guided neural networks with meaningful internal layers. *Computer Methods in Applied Mechanics and Engineering*, 381:113816, 2021a. (cited on Page 18, 20, 190, 294, 299, and 308)

Jacobo Ayensa-Jiménez, Marina Pérez-Aliacar, Teodora Randelovic, José Antonio Sanz-Herrera, Mohamed H Doweidar, and Manuel Doblaré. Analysis of the parametric correlation in mathematical modeling of in vitro glioblastoma evolution using copulas. *Mathematics*, 9(1):27, 2021b. (cited on Page 19, 20, 265, and 276)

Jose M Ayuso, Haneen A Basheer, Rosa Monge, Pablo Sánchez-Álvarez, Manuel Doblaré, Steven D Shnyder, Victoria Vinader, Kamyar Afarinkia, Luis J Fernández, and Ignacio Ochoa. Study of the chemotactic response of multicellular spheroids in a microfluidic device. *PloS one*, 10(10):e0139515, 2015. (cited on Page 297)

Jose M Ayuso, María Virumbrales-Muñoz, Alodia Lacueva, Pilar M Lanuza, Elisa Checa-Chavarria, Pablo Botella, Eduardo Fernández, Manuel Doblare, Simon J Allison, Roger M Phillips, et al. Development and characterization of a microfluidic model of the tumour microenvironment. *Scientific reports*, 6(1):1–16, 2016. (cited on Page 236, 248, 249, 250, 285, 287, 297, 300, and 316)

Jose M Ayuso, Rosa Monge, Alicia Martínez-González, María Virumbrales-Muñoz, Guillermo A Llamazares, Javier Berganzo, Aurelio Hernández-Laín, Jorge Santolaria, Manuel Doblaré, Christopher Hubert, et al. Glioblastoma on a microfluidic chip: generating pseudopalisades and enhancing aggressiveness through blood vessel obstruction events. *Neuro-oncology*, 19(4):503–513, 2017. (cited on Page 10, 15, 234, 236, 248, 249, 250, 285, 287, 299, 316, 354, and 370)

Alberto Badías, Sarah Curtit, David González, Icíar Alfaro, Francisco Chinesta, and Elías Cueto. An augmented reality platform for interactive aerodynamic design and analysis. *International Journal for Numerical Methods in Engineering*, 120(1): 125–138, 2019. (cited on Page 57)

Alberto Badías, David González, Icíar Alfaro, Francisco Chinesta, and Elías Cueto. Real-time interaction of virtual and physical objects in mixed reality applications. *International Journal for Numerical Methods in Engineering*, 121(17):3849–3868, 2020. (cited on Page 57, 356, and 372)

Joan Baiges, Ramon Codina, Inocencio Castanar, and Ernesto Castillo. A finite element reduced-order model based on adaptive mesh refinement and artificial neural networks. *International Journal for Numerical Methods in Engineering*, 121 (4):588–601, 2020. (cited on Page 34)

Maciej J Balajewicz, Earl H Dowell, and Bernd R Noack. Low-dimensional modelling of high-reynolds-number shear flows incorporating constraints from the navier–stokes equation. *Journal of Fluid Mechanics*, 729:285–308, 2013. (cited on Page 50)

Markus Bär, Rainer Hegger, and Holger Kantz. Fitting partial differential equations to space-time dynamics. *Physical Review E*, 59(1):337, 1999. (cited on Page 51)

Rémi Bardenet, Mátyás Brendel, Balázs Kégl, and Michele Sebag. Collaborative hyperparameter tuning. In *International conference on machine learning*, pages 199–207, 2013. (cited on Page 227)

Martino Bardi, Italo Capuzzo Dolcetta, et al. *Optimal control and viscosity solutions of Hamilton-Jacobi-Bellman equations*, volume 12. Springer, 1997. (cited on Page 44)

Grigory Isaakovich Barenblatt, Vladimir Mordukhovich Entov, and Viktor Mikhaĭlovich Ryzhik. *Theory of fluid flows through natural rocks*. Norwell, MA (USA); Kluwer Academic Publishers, 1989. (cited on Page 204)

Guy Barles and Halil Mete Soner. Option pricing with transaction costs and a nonlinear black-scholes equation. *Finance and Stochastics*, 2(4):369–397, 1998. (cited on Page 204)

Gianni Bartoli, Claudio Mannini, and Tommaso Massai. Quasi-static combination of wind loads: A copula-based approach. *Journal of wind engineering and industrial aerodynamics*, 99(6-7):672–681, 2011. (cited on Page 235)

Frédéric Bastien, Pascal Lamblin, Razvan Pascanu, James Bergstra, Ian Goodfellow, Arnaud Bergeron, Nicolas Bouchard, David Warde-Farley, and Yoshua Bengio. Theano: new features and speed improvements. *arXiv preprint arXiv:1211.5590*, 2012. (cited on Page 145)

George Keith Batchelor. *An introduction to fluid dynamics*. Cambridge university press, 2000. (cited on Page 177)

Yavar Bathaee. The artificial intelligence black box and the failure of intent and causation. *Harv. JL & Tech.*, 31:889, 2017. (cited on Page 12)

Ulrike Baur, Christopher Beattie, Peter Benner, and Serkan Gugercin. Interpolatory projection methods for parameterized model reduction. *SIAM Journal on Scientific Computing*, 33(5):2489–2518, 2011. (cited on Page 34)

Elaine L Bearer, John S Lowengrub, Hermann B Frieboes, Yao-Li Chuang, Fang Jin, Steven M Wise, Mauro Ferrari, David B Agus, and Vittorio Cristini. Multiparameter computational modeling of tumor invasion. *Cancer research*, 69(10):4493–4501, 2009. (cited on Page 10, 13, 235, 259, 262, and 321)

Christian Beck, E Weinan, and Arnulf Jentzen. Machine learning approximation algorithms for high-dimensional fully nonlinear partial differential equations and second-order backward stochastic differential equations. *Journal of Nonlinear Science*, 29(4):1563–1619, 2019. (cited on Page 44)

James Vere Beck and Kenneth J Arnold. *Parameter estimation in engineering and science*. James Beck, 1977. (cited on Page 81)

R Shekari Beidokhti and Alaeddin Malek. Solving initial-boundary value problems for systems of partial differential equations using neural networks and optimization techniques. *Journal of the Franklin Institute*, 346(9):898–913, 2009. (cited on Page 48)

Mikhail Belkin and Partha Niyogi. Laplacian eigenmaps for dimensionality reduction and data representation. *Neural computation*, 15(6):1373–1396, 2003. (cited on Page 59 and 82)

Peter Benner, Serkan Gugercin, and Karen Willcox. A survey of projection-based model reduction methods for parametric dynamical systems. *SIAM review*, 57(4): 483–531, 2015. (cited on Page 34)

TH Benzinger. Thermodynamics, chemical reactions and molecular biology. *Nature*, 229(5280):100–102, 1971. (cited on Page 315)

Michael E Berens and Alf Giese. "... those left behind." biology and oncology of invasive glioma cells. *Neoplasia*, 1(3):208–219, 1999. (cited on Page 259)

Jens Berg and Kaj Nyström. A unified deep artificial neural network approach to partial differential equations in complex geometries. *Neurocomputing*, 317:28–41, 2018. (cited on Page 48)

Jens Berg and Kaj Nyström. Data-driven discovery of pdes in complex datasets. *Journal of Computational Physics*, 384:239–252, 2019. (cited on Page 54)

James Bergstra, Frédéric Bastien, Olivier Breuleux, Pascal Lamblin, Razvan Pascanu, Olivier Delalleau, Guillaume Desjardins, David Warde-Farley, Ian Goodfellow, Arnaud Bergeron, et al. Theano: Deep learning on gpus with python. In *NIPS 2011, BigLearning Workshop, Granada, Spain*, volume 3, pages 1–48. Citeseer, 2011. (cited on Page 145)

Gal Berkooz, Philip Holmes, and John L Lumley. The proper orthogonal decomposition in the analysis of turbulent flows. *Annual review of fluid mechanics*, 25(1): 539–575, 1993. (cited on Page 34)

David M Berry. The computational turn: Thinking about the digital humanities. *Culture machine*, 12, 2011. (cited on Page 25)

Simone Bersini, Jessie S Jeon, Gabriele Dubini, Chiara Arrigoni, Seok Chung, Joseph L Charest, Matteo Moretti, and Roger D Kamm. A microfluidic 3d in vitro model for specificity of breast cancer metastasis to bone. *Biomaterials*, 35 (8):2454–2461, 2014. (cited on Page 9)

Lorenzo Bertini and Nicoletta Cancrini. The stochastic heat equation: Feynman-kac formula and intermittence. *Journal of statistical Physics*, 78(5):1377–1401, 1995. (cited on Page 44)

Miguel A Bessa, R Bostanabad, Zeliang Liu, A Hu, Daniel W Apley, C Brinson, Wei Chen, and Wing Kam Liu. A framework for data-driven analysis of materials under uncertainty: Countering the curse of dimensionality. *Computer Methods in Applied Mechanics and Engineering*, 320:633–667, 2017. (cited on Page 37)

Philippe Besse, Céline Castets-Renard, Aurélien Garivier, and Jean-Michel Loubes. Can everyday ai be ethical? machine learning algorithm fairness. *Machine Learning Algorithm Fairness (May 20, 2018). Statistiques et Société*, 6(3), 2019. (cited on Page 12)

Bhavana Bhadriraju, Abhinav Narasingam, and Joseph Sang-Il Kwon. Machine learning-based adaptive model identification of systems: Application to a chemical process. *Chemical Engineering Research and Design*, 152:372–383, 2019. (cited on Page 52)

Ilias Bilionis, Nicholas Zabaras, Bledar A Konomi, and Guang Lin. Multi-output separable gaussian process: Towards an efficient, fully bayesian paradigm for uncertainty quantification. *Journal of Computational Physics*, 241:212–239, 2013. (cited on Page 37)

Chris M Bishop. Training with noise is equivalent to tikhonov regularization. *Neural computation*, 7(1):108–116, 1995. (cited on Page 66)

Christopher M Bishop. Pattern recognition. *Machine Learning*, 128:1–58, 2006. (cited on Page 10)

Przemyslaw Bogacki and Lawrence F Shampine. A 3 (2) pair of runge-kutta formulas. *Applied Mathematics Letters*, 2(4):321–325, 1989. (cited on Page 409)

Hamid Bolouri, Lue Ping Zhao, and Eric C Holland. Big data visualization identifies the multidimensional molecular landscape of human gliomas. *Proceedings of the national academy of sciences*, 113(19):5394–5399, 2016. (cited on Page 11)

Javier Bonet, Antonio J Gil, and Richard D Wood. *Nonlinear solid mechanics for finite element analysis: statics*. Cambridge University Press, 2016. (cited on Page 166)

Josh Bongard and Hod Lipson. Automated reverse engineering of nonlinear dynamical systems. *Proceedings of the National Academy of Sciences*, 104(24):9943–9948, 2007. (cited on Page 51)

L Borkowski, C Sorini, and A Chattopadhyay. Recurrent neural network-based multiaxial plasticity model with regularization for physics-informed constraints. *Computers & Structures*, 258:106678, 2022. (cited on Page 36)

Domenico Borzacchiello, José V Aguado, and Francisco Chinesta. Non-intrusive sparse subspace learning for parametrized problems. *Archives of Computational Methods in Engineering*, 26(2):303–326, 2019. (cited on Page 40)

Gert-Jan Both, Subham Choudhury, Pierre Sens, and Remy Kusters. Deepmod: Deep learning for model discovery in noisy data. *Journal of Computational Physics*, 428:109985, 2021. (cited on Page 54)

Heni Boubaker and Nadia Sghaier. Portfolio optimization in the presence of dependent financial returns with long memory: A copula based approach. *Journal of Banking & Finance*, 37(2):361–377, 2013. (cited on Page 235)

Alexandra Boussommier-Calleja, Ran Li, Michelle B Chen, Siew Cheng Wong, and Roger D Kamm. Microfluidics: a new tool for modeling cancer–immune interactions. *Trends in cancer*, 2(1):6–19, 2016. (cited on Page 9)

George EP Box. All models are wrong, but some are useful. *Robustness in Statistics*, 202(1979):549, 1979. (cited on Page 242)

John P Boyd. *Chebyshev and Fourier spectral methods*. Courier Corporation, 2001. (cited on Page 30 and 165)

Daniel J Brat. Glioblastoma: biology, genetics, and behavior. *American Society of Clinical Oncology Educational Book*, 32(1):102–107, 2012. (cited on Page 7, 234, and 285)

Daniel J Brat and Erwin G Van Meir. Vaso-occlusive and prothrombotic mechanisms associated with tumor hypoxia, necrosis, and accelerated growth in glioblastoma. *Laboratory Investigation*, 84(4):397, 2004. (cited on Page 234 and 236)

Daniel J Brat, Amilcar A Castellano-Sanchez, Stephen B Hunter, Marcia Pecot, Cynthia Cohen, Elizabeth H Hammond, Sarojini N Devi, Balveen Kaur, and Erwin G Van Meir. Pseudopalisades in glioblastoma are hypoxic, express extracellular matrix proteases, and are formed by an actively migrating cell population. *Cancer research*, 64(3):920–927, 2004. (cited on Page 7 and 234)

Curt M Breneman, L Catherine Brinson, Linda S Schadler, Bharath Natarajan, Michael Krein, Ke Wu, Lisa Morkowchuk, Yang Li, Hua Deng, and Hongyi Xu. Stalking the materials genome: A data-driven approach to the virtual design of nanostructured polymers. *Advanced Functional Materials*, 23(46):5746–5752, 2013. (cited on Page 70)

Mathias Brieu and François Devries. Micro-mechanical approach and algorithm for the study of damage appearance in elastomer composites. *Composite structures*, 46(4):309–319, 1999. (cited on Page 92)

Andrew Briggs, Taane Clark, Jane Wolstenholme, and Philip Clarke. Missing.... presumed at random: cost-analysis of incomplete data. *Health economics*, 12(5): 377–392, 2003. (cited on Page 102 and 122)

Scott Broderick and Krishna Rajan. Informatics derived materials databases for multifunctional properties. *Science and technology of advanced materials*, 2015. (cited on Page 57)

Michael M Bronstein, Joan Bruna, Yann LeCun, Arthur Szlam, and Pierre Vandergheynst. Geometric deep learning: going beyond euclidean data. *IEEE Signal Processing Magazine*, 34(4):18–42, 2017. (cited on Page 37 and 173)

JEM Brouwers, Bert van Rietbergen, Rik Huiskes, and K Ito. Effects of pth treatment on tibial bone of ovariectomized rats assessed by in vivo micro-ct. *Osteoporosis International*, 20(11):1823–1835, 2009. (cited on Page 127)

J Martin Brown and William R Wilson. Exploiting tumour hypoxia in cancer treatment. *Nature Reviews Cancer*, 4(6):437, 2004. (cited on Page 260 and 262)

Charles G Broyden. A class of methods for solving nonlinear simultaneous equations. *Mathematics of computation*, 19(92):577–593, 1965. (cited on Page 410)

Steven L Brunton and J Nathan Kutz. Methods for data-driven multiscale model discovery for materials. *Journal of Physics: Materials*, 2(4):044002, 2019. (cited on Page 52)

Steven L Brunton, Joshua L Proctor, and J Nathan Kutz. Discovering governing equations from data by sparse identification of nonlinear dynamical systems. *Proceedings of the National Academy of Sciences*, 113(15):3932–3937, 2016. (cited on Page 24, 51, and 53)

Steven L Brunton, Bingni W Brunton, Joshua L Proctor, Eurika Kaiser, and J Nathan Kutz. Chaos as an intermittently forced linear system. *Nature communications*, 8 (1):1–9, 2017. (cited on Page 52)

John Charles Butcher. *Numerical methods for ordinary differential equations*. John Wiley & Sons, 2016. (cited on Page 431)

Helen M Byrne. Dissecting cancer through mathematics: from the cell to the animal model. *Nature Reviews Cancer*, 10(3):221–230, 2010. (cited on Page 10 and 234)

HM Byrne, T Alarcon, MR Owen, SD Webb, and PK Maini. Modelling aspects of cancer dynamics: a review. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 364(1843):1563–1578, 2006. (cited on Page 10, 13, and 321)

Luis A Caffarelli and Juan L Vazquez. Nonlinear porous medium flow with fractional potential pressure. *arXiv preprint arXiv:1001.0410*, 2010. (cited on Page 204)

Jian-Feng Cai, Bin Dong, Stanley Osher, and Zuowei Shen. Image restoration: total variation, wavelet frames, and beyond. *Journal of the American Mathematical Society*, 25(4):1033–1089, 2012. (cited on Page 54 and 170)

P Calabresi and PS Schein. Basic principles and clinical management of cancer. *Medical Oncology, McGraw-Hill: New York*, 33, 1993. (cited on Page 259)

Emmanuel Candes and Terence Tao. The dantzig selector: Statistical estimation when p is much larger than n. *The annals of Statistics*, 35(6):2313–2351, 2007. (cited on Page 45)

Shaosheng Cao, Wei Lu, and Qiongkai Xu. Grarep: Learning graph representations with global structural information. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*, CIKM '15, page 891–900, New York, NY, USA, 2015. Association for Computing Machinery. ISBN 9781450337946. doi: 10.1145/2806416.2806512. URL https://doi.org/10.1145/2806416.2806512. (cited on Page 37)

Wenming Cao, Zhiyue Yan, Zhiquan He, and Zhihai He. A comprehensive survey on geometric deep learning. *IEEE Access*, 8:35929–35949, 2020. (cited on Page 173)

Yang Cao, Shengtai Li, and Linda Petzold. Adjoint sensitivity analysis for differential-algebraic equations: algorithms and software. *Journal of computational and applied mathematics*, 149(1):171–191, 2002. (cited on Page 60)

Yang Cao, Shengtai Li, Linda Petzold, and Radu Serban. Adjoint sensitivity analysis for differential-algebraic equations: The adjoint dae system and its numerical solution. *SIAM journal on scientific computing*, 24(3):1076–1089, 2003. (cited on Page 60)

Kevin Carlberg, Ray Tuminaro, and Paul Boggs. Preserving lagrangian structure in nonlinear model reduction with application to structural dynamics. *SIAM Journal on Scientific Computing*, 37(2):B153–B184, 2015. (cited on Page 50)

Giuseppe Carleo and Matthias Troyer. Solving the quantum many-body problem with artificial neural networks. *Science*, 355(6325):602–606, 2017. (cited on Page 44)

Gunnar Carlsson. Topology and data. *Bulletin of the American Mathematical Society*, 46(2):255–308, 2009. (cited on Page 60)

Peter Carmeliet and Rakesh K Jain. Angiogenesis in cancer and other diseases. *nature*, 407(6801):249–257, 2000. (cited on Page 317)

Pietro Carrara, Laura De Lorenzis, Laurent Stainier, and Michael Ortiz. Data-driven fracture mechanics. *Computer Methods in Applied Mechanics and Engineering*, 372:113390, 2020. (cited on Page 58)

Aude Carreau, Bouchra El Hafny-Rahbi, Agata Matejuk, Catherine Grillon, and Claudine Kieda. Why is the partial oxygen pressure of human tissues a crucial parameter? small molecules and hypoxia. *Journal of cellular and molecular medicine*, 15(6):1239–1253, 2011. (cited on Page 254 and 297)

Alessandra Carriero, Lisa Abela, Andrew A Pitsillides, and Sandra J Shefelbine. Ex vivo determination of bone tissue strains for an in vivo mouse tibial loading model. *Journal of biomechanics*, 47(10):2490–2497, 2014. (cited on Page 127)

DR Carter, PR Blenman, and GS Beaupre. Correlations between mechanical stress history and tissue differentiation in initial fracture healing. *Journal of Orthopaedic Research*, 6(5):736–748, 1988. (cited on Page 126)

Davide Castelvecchi. Can we open the black box of ai? *Nature News*, 538(7623):20, 2016. (cited on Page 12 and 155)

Enrique Castillo, Jose M Gutierrez, and Ali S Hadi. *Expert systems and probabilistic network models*. Springer Science & Business Media, 2012. (cited on Page 33)

Kathryn Chaloner and Isabella Verdinelli. Bayesian experimental design: A review. *Statist. Sci.*, 10(3):273–304, 08 1995. doi: 10.1214/ss/1177009939. URL https://doi.org/10.1214/ss/1177009939. (cited on Page 247)

Kathleen Champion, Bethany Lusch, J Nathan Kutz, and Steven L Brunton. Data-driven discovery of coordinates and governing equations. *Proceedings of the National Academy of Sciences*, 116(45):22445–22451, 2019a. (cited on Page 52)

Kathleen P Champion, Steven L Brunton, and J Nathan Kutz. Discovery of nonlinear multiscale systems: Sampling strategies and embeddings. *SIAM Journal on Applied Dynamical Systems*, 18(1):312–333, 2019b. (cited on Page 52)

Quentin Chan-Wai-Nam, Joseph Mikael, and Xavier Warin. Machine learning for semi linear pdes. *Journal of Scientific Computing*, 79(3):1667–1712, 2019. (cited on Page 44)

Soumyadeep Chatterjee, Sheng Chen, and Arindam Banerjee. Generalized dantzig selector: Application to the k-support norm. *arXiv preprint arXiv:1406.5291*, 2014. (cited on Page 46)

Saifon Chaturantabut and Danny C Sorensen. Nonlinear model reduction via discrete empirical interpolation. *SIAM Journal on Scientific Computing*, 32(5):2737–2764, 2010. (cited on Page 34)

Chun-Teh Chen and Grace X Gu. Generative deep neural networks for inverse materials design using backpropagation and active learning. *Advanced Science*, 7 (5):1902607, 2020. (cited on Page 37)

Jee-Wei Emily Chen, Jan Lumibao, Sarah Leary, Jann N Sarkaria, Andrew J Steelman, H Rex Gaskins, and Brendan AC Harley. Crosstalk between microglia and patient-derived glioblastoma cells inhibit invasion in a three-dimensional gelatin hydrogel model. *Journal of neuroinflammation*, 17(1):1–15, 2020a. (cited on Page 316)

Yuyao Chen, Lu Lu, George Em Karniadakis, and Luca Dal Negro. Physics-informed neural networks for inverse problems in nano-optics and metamaterials. *Optics express*, 28(8):11618–11633, 2020b. (cited on Page 49)

Zhao Chen, Yang Liu, and Hao Sun. Physics-informed learning of governing equations from scarce data. *Nature communications*, 12(1):1–13, 2021. (cited on Page 54 and 64)

Shunfeng Cheng and Michael Pecht. A fusion prognostics method for remaining useful life prediction of electronic products. In *Automation Science and Engineering, 2009. CASE 2009. IEEE International Conference on*, pages 102–107. IEEE, 2009. (cited on Page 70)

Serena PH Chiang, Ramon M Cabrera, and Jeffrey E Segall. Tumor cell intravasation. *American Journal of Physiology-Cell Physiology*, 311(1):C1–C14, 2016. (cited on Page 317)

Francisco Chinesta, Amine Ammar, and Elías Cueto. Proper generalized decomposition of multiscale models. *International journal for numerical methods in engineering*, 83(8-9):1114–1132, 2010a. (cited on Page 40)

Francisco Chinesta, Amine Ammar, and Elías Cueto. Recent advances and new challenges in the use of the proper generalized decomposition for solving multidimensional models. *Archives of Computational methods in Engineering*, 17(4): 327–350, 2010b. (cited on Page 39)

Francisco Chinesta, Pierre Ladeveze, and Elías Cueto. A short review on model order reduction based on proper generalized decomposition. *Archives of Computational Methods in Engineering*, 18(4):395, 2011. (cited on Page 39 and 40)

Francisco Chinesta, Serge Cescotto, Elías Cueto, and Philippe Lorong. *Natural element method for the simulation of structures and processes*. John Wiley & Sons, 2013a. (cited on Page 173)

Francisco Chinesta, Roland Keunings, and Adrien Leygue. *The proper generalized decomposition for advanced numerical simulations: a primer*. Springer Science & Business Media, 2013b. (cited on Page 40)

Francisco Chinesta, Adrien Leygue, Felipe Bordeu, Jose Vicente Aguado, Elías Cueto, David González, Iciar Alfaro, Amine Ammar, and Antonio Huerta. Pgd-based computational vademecum for efficient design, optimization and control. *Archives of Computational Methods in Engineering*, 20(1):31–59, 2013c. (cited on Page 40 and 65)

Francisco Chinesta, Elias Cueto, Emmanuelle Abisset-Chavanne, Jean Louis Duval, and Fouad El Khaldi. Virtual, digital and hybrid twins: a new paradigm in data-based engineering and engineered data. *Archives of computational methods in engineering*, 27(1):105–134, 2020. (cited on Page 56)

William C Cho. Big data for cancer research, 2015. (cited on Page 11)

Óscar Ciaurri, Luz Roncal, Pablo Raúl Stinga, José L Torrea, and Juan Luis Varona. Nonlocal discrete diffusion equations and the fractional discrete laplacian, regularity and applications. *Advances in Mathematics*, 330:688–738, 2018. (cited on Page 203)

Myriam Cilla, Ignacio Pérez-Rey, Miguel Angel Martínez, Estefania Peña, and Javier Martínez. On the use of machine learning techniques for the mechanical characterization of soft biological tissues. *International journal for numerical methods in biomedical engineering*, 34(10):e3121, 2018. (cited on Page 36)

Davide Cirillo and Alfonso Valencia. Big data analytics for personalized medicine. *Current opinion in biotechnology*, 58:161–167, 2019. (cited on Page 317)

LE Claes and CA Heigele. Magnitudes of local stress and strain along bony surfaces predict the course and type of fracture healing. *Journal of biomechanics*, 32(3): 255–266, 1999. (cited on Page 126 and 143)

Cyril F Colebrook and Curt M White. Experiments with fluid friction in roughened pipes. *Proceedings of the Royal Society of London. Series A-Mathematical and Physical Sciences*, 161(906):367–381, 1937. (cited on Page 177)

Cyril Frank Colebrook, T Blench, H Chatley, EH Essex, JR Finniecome, G Lacey, J Williamson, and GG Macdonald. Correspondence. turbulent flow in pipes, with particular reference to the transition region between the smooth and rough pipe laws.(includes plates). *Journal of the Institution of Civil engineers*, 12(8):393–422, 1939. (cited on Page 177)

Sergio Conti, Stefan Müller, and Michael Ortiz. Data-driven problems in elasticity. *Archive for Rational Mechanics and Analysis*, 229(1):79–123, 2018. (cited on Page 58)

Sergio Conti, Stefan Müller, and Michael Ortiz. Data-driven finite elasticity. *Archive for Rational Mechanics and Analysis*, 237(1):1–33, 2020. (cited on Page 58)

Athel Cornish-Bowden. The origins of enzyme kinetics. *FEBS letters*, 587(17): 2725–2730, 2013. (cited on Page 295)

Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine learning*, 20(3):273–297, 1995. (cited on Page 33)

Richard Courant and David Hilbert. *Methods of mathematical physics: partial differential equations*. John Wiley & Sons, 2008. (cited on Page 44)

SC Cowin. Mechanosensation and fluid transport in living bone. *Journal of Musculoskeletal and Neuronal Interactions*, 2(3):256–260, 2002. (cited on Page 126 and 143)

Alison Cozad, Nikolaos V Sahinidis, and David C Miller. Learning surrogate models for simulation-based optimization. *AIChE Journal*, 60(6):2211–2227, 2014. (cited on Page 35 and 163)

José Crespo, Marcos Latorre, and Francisco Javier Montáns. Wypiwyg hyperelasticity for isotropic, compressible materials. *Computational Mechanics*, 59(1):73–92, 2017. (cited on Page 59)

Elías Cueto, David González, and Icíar Alfaro. *Proper generalized decompositions: an introduction to computer implementation with Matlab*. Springer, 2016. (cited on Page 39)

Xin Cui, Chao Ma, Varshini Vasudevaraja, Jonathan Serrano, Jie Tong, Yansong Peng, Michael Delorenzo, Guomiao Shen, Joshua Frenster, Renee-Tyler Tan Morales, et al. Dissecting the immunosuppressive tumor microenvironments in glioblastoma-on-a-chip for optimized pd-1 immunotherapy. *eLife*, 9:e52253, 2020. (cited on Page 285)

George Cybenko. Approximations by superpositions of a sigmoidal function. *Mathematics of Control, Signals and Systems*, 2:183–192, 1989. (cited on Page 47, 151, and 305)

Marie Dalémat, Michel Coret, Adrien Leygue, and Erwan Verron. Measuring stress field without constitutive equation. *Mechanics of Materials*, 136:103087, 2019. (cited on Page 59)

Magnus Dam, Morten Brøns, Jens Juul Rasmussen, Volker Naulin, and Jan S Hesthaven. Sparse identification of a predator-prey system from simulation data of a convection model. *Physics of Plasmas*, 24(2):022310, 2017. (cited on Page 52)

Bryan C Daniels and Ilya Nemenman. Automated adaptive inference of phenomenological dynamical models. *Nature communications*, 6(1):1–8, 2015. (cited on Page 51)

Jérôme Darbon. On convex finite-dimensional variational methods in imaging sciences and hamilton–jacobi equations. *SIAM Journal on Imaging Sciences*, 8(4):2268–2293, 2015. (cited on Page 44)

Jerome Darbon, Gabriel P Langlois, and Tingwei Meng. Overcoming the curse of dimensionality for some hamilton–jacobi partial differential equations via neural network architectures. *Research in the Mathematical Sciences*, 7(3):1–50, 2020. (cited on Page 44)

Frederica Darema. Dynamic data driven applications systems: A new paradigm for application simulations and measurements. In *International Conference on Computational Science*, pages 662–669. Springer, 2004. (cited on Page 24 and 55)

Silvio Däster, Nunzia Amatruda, Diego Calabrese, Robert Ivanek, Eleonora Turrini, Raoul A Droeser, Paul Zajac, Carmela Fimognari, Giulio C Spagnoli, Giandomenica Iezzi, et al. Induction of hypoxia and necrosis in multicellular tumor spheroids is associated with resistance to chemotherapy treatment. *Oncotarget*, 8(1):1725, 2017. (cited on Page 285)

Alexandru Daşu, Iuliana Toma-Daşu, and Mikael Karlsson. Theoretical simulation of tumour oxygenation and results from acute and chronic hypoxia. *Physics in Medicine & Biology*, 48(17):2829, 2003. (cited on Page 261, 262, and 296)

JP Moitinho de Almeida. A basis for bounding the errors of proper generalised decomposition solutions in solid mechanics. *International Journal for Numerical Methods in Engineering*, 94(10):961–984, 2013. (cited on Page 40)

Roy De Maesschalck, Delphine Jouan-Rimbaud, and Désiré L Massart. The mahalanobis distance. *Chemometrics and intelligent laboratory systems*, 50(1):1–18, 2000. (cited on Page 66, 80, and 245)

David Montes de Oca Zapiain, James A Stewart, and Rémi Dingreville. Accelerating phase-field-based microstructure evolution predictions via surrogate models trained by machine learning methods. *npj Computational Materials*, 7(1):1–11, 2021. (cited on Page 36)

Brian M de Silva, David M Higdon, Steven L Brunton, and J Nathan Kutz. Discovery of physics from data: universal laws and discrepancies. *Frontiers in artificial intelligence*, 3:25, 2020. (cited on Page 56)

Lokenath Debnath. *Nonlinear partial differential equations for scientists and engineers*. Springer Science & Business Media, 2011. (cited on Page 323)

Thomas S Deisboeck and Georgios S Stamatakos. *Multiscale Cancer Modeling*. CRC Press, 2010. (cited on Page 234)

Thomas S Deisboeck, Zhihui Wang, Paul Macklin, and Vittorio Cristini. Multiscale cancer modeling. *Annual review of biomedical engineering*, 13:127–155, 2011. (cited on Page 10)

Stefano Demarta and Alexander J McNeil. The t copula and related copulas. *International statistical review*, 73(1):111–129, 2005. (cited on Page 266)

Arthur P Dempster, Nan M Laird, and Donald B Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the royal statistical society. Series B (methodological)*, pages 1–38, 1977. (cited on Page 103)

Mark W Dewhirst, Timothy W Secomb, Edgardo T Ong, Richard Hsu, and Joseph F Gross. Determination of local oxygen consumption rates in tumors. *Cancer Research*, 54(13):3333–3336, 1994. (cited on Page 261)

Ken Dill and Sarina Bromberg. *Molecular driving forces: statistical thermodynamics in biology, chemistry, physics, and nanoscience*. Garland Science, 2010. (cited on Page 315)

Anne Dirkse, Anna Golebiewska, Thomas Buder, Petr V Nazarov, Arnaud Muller, Suresh Poovathingal, Nicolaas HC Brons, Sonia Leite, Nicolas Sauvageot, Dzjemma Sarkisjan, et al. Stem cell-associated heterogeneity in glioblastoma results from intrinsic tumor plasticity shaped by the microenvironment. *Nature communications*, 10(1):1–16, 2019. (cited on Page 291)

MWMG Dissanayake and Nhan Phan-Thien. Neural-network-based approximations for solving partial differential equations. *communications in Numerical Methods in Engineering*, 10(3):195–201, 1994. (cited on Page 47)

Manfredo Perdigao Do Carmo and J Flaherty Francis. *Riemannian geometry*, volume 115. Birkhäuser Boston, 1992. (cited on Page 111)

Bin Dong, Qingtang Jiang, and Zuowei Shen. Image restoration: Wavelet frame shrinkage, nonlinear evolution pdes, and beyond. *Multiscale Modeling & Simulation*, 15(1):606–660, 2017. (cited on Page 54 and 170)

Sheng Dong, Chong Zhou, Shan-Shan Tao, and Dong-Sheng Xue. Bivariate gumbel distribution based on clayton copula and its application in offshore platform design. *periodical of ocean university of China*, 41(10):117–120, 2011. (cited on Page 235)

John R Dormand and Peter J Prince. A family of embedded runge-kutta formulae. *Journal of computational and applied mathematics*, 6(1):19–26, 1980. (cited on Page 409 and 414)

Filip Karlo Došilović, Mario Brčić, and Nikica Hlupić. Explainable artificial intelligence: A survey. In *2018 41st International convention on information and communication technology, electronics and microelectronics (MIPRO)*, pages 0210–0215. IEEE, 2018. (cited on Page 228)

Vivek Dua. An artificial neural network approximation based decomposition approach for parameter estimation of system of ordinary differential equations. *Computers & chemical engineering*, 35(3):545–553, 2011. (cited on Page 41)

Richard Durrett. *Stochastic calculus: a practical introduction*. CRC press, 2018. (cited on Page 43)

Rasheena Edmondson, Jessica Jenkins Broglie, Audrey F Adcock, and Liju Yang. Three-dimensional cell culture systems and their applications in drug discovery and cell-based biosensors. *Assay and drug development technologies*, 12(4):207–218, 2014. (cited on Page 8)

Sohrab Effati and Morteza Pakdaman. Artificial neural network approach for solving fuzzy differential equations. *Information Sciences*, 180(8):1434–1457, 2010. (cited on Page 47)

Robert Eggersmann, Trenton Kirchdoerfer, Stefanie Reese, Laurent Stainier, and Michael Ortiz. Model-free data-driven inelasticity. *Computer Methods in Applied Mechanics and Engineering*, 350:81–99, 2019. (cited on Page 58)

Robert Eggersmann, Laurent Stainier, Michael Ortiz, and Stefanie Reese. Efficient data structures for model-free data-driven computational mechanics. *Computer Methods in Applied Mechanics and Engineering*, 382:113855, 2021. (cited on Page 58)

R Eils and AE Kriete. Computational systems biology: from molecular mechanisms to disease, 2013. (cited on Page 321)

F El Halabi, D González, JA Sanz-Herrera, and M Doblaré. A pgd-based multiscale formulation for non-linear solid mechanics under small deformations. *Computer Methods in Applied Mechanics and Engineering*, 305:806–826, 2016. (cited on Page 126)

Scott R Eliason. *Maximum likelihood estimation: Logic and practice*. Number 96 in Quantitative Applications in the Social Sciences. Sage, 1993. (cited on Page 81)

GW Ellis, C Yao, Rui Zhao, and Df Penumadu. Stress-strain modeling of sands using artificial neural networks. *Journal of geotechnical engineering*, 121(5):429–435, 1995. (cited on Page 36)

Frank Emmert-Streib, Olli Yli-Harja, and Matthias Dehmer. Explainable artificial intelligence and machine learning: A reality rooted perspective. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 10(6):e1368, 2020. (cited on Page 12 and 155)

Craig K Enders. A primer on maximum likelihood algorithms available for use with missing data. *Structural Equation Modeling*, 8(1):128–141, 2001. (cited on Page 103)

Craig K Enders. *Applied missing data analysis*. Guilford Press, 2010. (cited on Page 103)

Craig K Enders and Deborah L Bandalos. The relative performance of full information maximum likelihood estimation for missing data in structural equation models. *Structural equation modeling*, 8(3):430–457, 2001. (cited on Page 103)

Virginia Espina and Lance A Liotta. What is the malignant nature of human ductal carcinoma in situ? *Nature Reviews Cancer*, 11(1):68–75, 2011. (cited on Page 4)

Randall L Eubank. *Nonparametric regression and spline smoothing*. CRC press, 1999. (cited on Page 33)

F Gaynor Evans and Raúl Vincentelli. Relation of collagen fiber orientation to some mechanical properties of human cortical bone. *Journal of biomechanics*, 2(1):63–71, 1969. (cited on Page 127)

Lawrence C Evans and Panagiotis E Souganidis. Differential games and representation formulas for solutions of hamilton-jacobi-isaacs equations. *Indiana University mathematics journal*, 33(5):773–797, 1984. (cited on Page 44)

B Everett. *An introduction to latent variable models*. Springer Science & Business Media, 2013. (cited on Page 37)

Yanqin Fan. Goodness-of-fit tests for a multivariate distribution by the empirical characteristic function. *Journal of Multivariate Analysis*, 62(1):36–63, 1997. (cited on Page 246)

Rui Fang, David Sondak, Pavlos Protopapas, and Sauro Succi. Neural network models for the anisotropic reynolds stress tensor in turbulent channel flow. *Journal of Turbulence*, 21(9-10):525–543, 2020. (cited on Page 41)

Gregory E Fasshauer and Qi Ye. A kernel-based collocation method for elliptic partial differential equations with random coefficients. In *Monte Carlo and Quasi-Monte Carlo Methods 2012*, pages 331–347. Springer, 2013. (cited on Page 37 and 43)

Frédéric Feyel. Multiscale fe2 elastoviscoplastic analysis of composite structures. *Computational Materials Science*, 16(1-4):344–354, 1999. (cited on Page 126 and 130)

Paul Fischer and Julia Mullen. Filter-based stabilization of spectral element methods. *Comptes Rendus de l'Académie des Sciences-Series I-Mathematics*, 332(3):265–270, 2001. (cited on Page 173)

Ronald Aylmer Fisher. *The design of experiments*. Oliver And Boyd; Edinburgh; London, 1937a. (cited on Page 247)

Ronald Aylmer Fisher. The wave of advance of advantageous genes. *Annals of eugenics*, 7(4):355–369, 1937b. (cited on Page 297)

Moritz Flaschel, Siddhant Kumar, and Laura De Lorenzis. Unsupervised discovery of interpretable hyperelastic constitutive laws. *Computer Methods in Applied Mechanics and Engineering*, 381:113852, 2021. (cited on Page 24, 61, 62, 67, 354, and 370)

Valentina Fodale, Mariaelena Pierobon, Lance Liotta, and Emanuel Petricoin. Mechanism of cell adaptation: when and how do cancer cells develop chemoresistance? *Cancer journal (Sudbury, Mass.)*, 17(2):89, 2011. (cited on Page 3)

Roseanne M Ford and Douglas A Lauffenburger. Analysis of chemotactic bacterial distributions in population migration assays using a mathematical model applicable to steep or shallow attractant gradients. *Bulletin of Mathematical Biology*, 53(5): 721–749, 1991. (cited on Page 258 and 262)

Alexander IJ Forrester and Andy J Keane. Recent advances in surrogate-based optimization. *Progress in aerospace sciences*, 45(1-3):50–79, 2009. (cited on Page 35)

Yousef Ahmed Fouad and Carmen Aanei. Revisiting the hallmarks of cancer. *American journal of cancer research*, 7(5):1016, 2017. (cited on Page 4)

Michalis Frangos, Youssef Marzouk, Karen Willcox, and Bart van Bloemen Waanders. Surrogate and reduced-order modeling: a comparison of approaches for large-scale statistical inverse problems [chapter 7]. In *Large-Scale Inverse Problems and Quantification of Uncertainty*. John Wiley & Sons, 2010. (cited on Page 35)

Till Daniel Frank. *Nonlinear Fokker-Planck equations: fundamentals and applications*. Springer Science & Business Media, 2005. (cited on Page 203)

Carsten Franke and Robert Schaback. Solving partial differential equations by collocation using radial basis functions. *Applied Mathematics and Computation*, 93(1):73–82, 1998. (cited on Page 37 and 43)

Stefania Fresca and Andrea Manzoni. Pod-dl-rom: enhancing deep learning-based reduced order models for nonlinear parametrized pdes by proper orthogonal decomposition. *Computer Methods in Applied Mechanics and Engineering*, 388:114181, 2022. (cited on Page 34)

James P Freyer and Robert M Sutherland. Regulation of growth saturation and development of necrosis in emt6/ro multicellular spheroids by the glucose and oxygen supply. *Cancer Research*, 46(7):3504–3512, 1986. (cited on Page 261)

Hermann B Frieboes, John S Lowengrub, S Wise, X Zheng, Paul Macklin, Elaine L Bearer, and Vittorio Cristini. Computer simulation of glioma growth and morphology. *Neuroimage*, 37:S59–S70, 2007. (cited on Page 259 and 260)

Dinorah Friedmann-Morvinski. Glioblastoma heterogeneity and cancer cell plasticity. *Critical Reviews™ in Oncogenesis*, 19(5), 2014. (cited on Page 291)

Frederick N Fritsch and Ralph E Carlson. Monotone piecewise cubic interpolation. *SIAM Journal on Numerical Analysis*, 17(2):238–246, 1980. (cited on Page 122)

Masaaki Fujii, Akihiko Takahashi, and Masayuki Takahashi. Asymptotic expansion as prior knowledge in deep learning method for high dimensional bsdes. *Asia-Pacific Financial Markets*, 26(3):391–408, 2019. (cited on Page 44)

Kenichi Funamoto, Ioannis K Zervantonakis, Yuchun Liu, Christopher J Ochs, Choong Kim, and Roger D Kamm. A novel microfluidic platform for high-resolution imaging of a three-dimensional cell culture under a controlled hypoxic environment. *Lab on a chip*, 12(22):4855–4863, 2012. (cited on Page 297)

Tomonari Furukawa and Genki Yagawa. Implicit constitutive modelling for viscoplasticity using neural networks. *International Journal for Numerical Methods in Engineering*, 43(2):195–219, 1998. (cited on Page 36)

Jakub Gajewski and Tomasz Sadowski. Sensitivity analysis of crack propagation in pavement bituminous layered structures using a hybrid system integrating artificial neural networks and finite element method. *Computational Materials Science*, 82: 114–117, 2014. (cited on Page 36)

Edgar A Galan, Haoran Zhao, Xukang Wang, Qionghai Dai, Wilhelm TS Huck, and Shaohua Ma. Intelligent microfluidics: The convergence of machine learning and microfluidics in materials science and biomedicine. *Matter*, 3(6):1893–1922, 2020. (cited on Page 11, 356, and 372)

Aurél Galántai. *Projectors and projection methods*, volume 6. Springer Science & Business Media, 2013. (cited on Page 392)

Armin Galetzka, Dimitrios Loukrezis, and Herbert De Gersem. Data-driven solvers for strongly nonlinear material response. *International Journal for Numerical Methods in Engineering*, 122(6):1538–1562, 2021a. (cited on Page 58)

Armin Galetzka, Dimitrios Loukrezis, and Herbert De Gersem. Three-dimensional data-driven magnetostatic field computation using real-world measurement data. *COMPEL-The international journal for computation and mathematics in electrical and electronic engineering*, 2021b. (cited on Page 58)

Lorenzo Galluzzi, Ilio Vitale, Stuart A Aaronson, John M Abrams, Dieter Adam, Patrizia Agostinis, Emad S Alnemri, Lucia Altucci, Ivano Amelio, David W Andrews, et al. Molecular mechanisms of cell death: recommendations of the nomenclature committee on cell death 2018. *Cell Death & Differentiation*, page 1, 2018. (cited on Page 255)

Baskar Ganapathysubramanian and Nicholas Zabaras. Modeling diffusion in random heterogeneous media: Data-driven models, stochastic collocation and the variational multiscale method. *Journal of Computational Physics*, 226(1):326–353, 2007. (cited on Page 92)

Sashikumaar Ganesan and Shangerganesh Lingeshwaran. Galerkin finite element method for cancer invasion mathematical model. *Computers & Mathematics with Applications*, 73(12):2603–2617, 2017. (cited on Page 300)

Shambavi Ganesh, Beliz Utebay, Jeremy Heit, and Ahmet F Coskun. Cellular sociology regulates the hierarchical spatial patterning and organization of cells in organisms. *Open Biology*, 10(12):200300, 2020. (cited on Page 316)

Nicholas Geneva and Nicholas Zabaras. Quantifying model form uncertainty in reynolds-averaged turbulence models with bayesian deep neural networks. *Journal of Computational Physics*, 383:125–147, 2019. (cited on Page 36)

Philip Gerlee and Alexander RA Anderson. An evolutionary hybrid cellular automaton model of solid tumour growth. *Journal of Theoretical Biology*, 246(4):583–603, 2007. (cited on Page 259, 260, 261, and 262)

Philip Gerlee and Sven Nelander. Travelling wave analysis of a mathematical model of glioblastoma growth. *Mathematical biosciences*, 276:75–81, 2016. (cited on Page 322)

Aurélien Géron. *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: Concepts, tools, and techniques to build intelligent systems*. O'Reilly Media, 2019. (cited on Page 145)

J Ghaboussi and DE Sidarta. New nested adaptive neural networks (nann) for constitutive modeling. *Computers and Geotechnics*, 22(1):29–52, 1998. (cited on Page 36)

J Ghaboussi, JH Garrett Jr, and Xiping Wu. Knowledge-based modeling of material behavior with neural networks. *Journal of engineering mechanics*, 117(1):132–153, 1991. (cited on Page 36)

Jamshid Ghaboussi. Advances in neural networks in computational mechanics and engineering. In *Advances of soft computing in engineering*, pages 191–236. Springer, 2010. (cited on Page 36)

F Ghavamian and A Simone. Accelerating multiscale finite element simulations of history-dependent materials using a recurrent neural network. *Computer Methods in Applied Mechanics and Engineering*, 357:112594, 2019. (cited on Page 36)

Ch Ghnatios, F Masson, Antonio Huerta, Adrien Leygue, E Cueto, and Francisco Chinesta. Proper generalized decomposition based dynamic data-driven control of thermal processes. *Computer Methods in Applied Mechanics and Engineering*, 213: 29–41, 2012. (cited on Page 39)

Chady Ghnatios, Emmanuelle Abisset, Amine Ammar, Elías Cueto, Jean-Louis Duval, and Francisco Chinesta. Advanced separated spatial representations for hardly separable domains. *Computer Methods in Applied Mechanics and Engineering*, 354:802–819, 2019a. (cited on Page 34)

Chady Ghnatios, Iciar Alfaro, David González, Francisco Chinesta, and Elias Cueto. Data-driven generic modeling of poroviscoelastic materials. *Entropy*, 21(12):1165, 2019b. (cited on Page 43)

Shyamal Ghosh, Atin Roy, and Subrata Chakraborty. Support vector regression based metamodeling for seismic reliability analysis of structures. *Applied Mathematical Modelling*, 64:584–602, 2018. (cited on Page 35)

A Giese, R Bjerkvig, ME Berens, and M Westphal. Cost of migration: invasion of malignant gliomas and implications for treatment. *Journal of Clinical Oncology*, 21(8):1624–1636, 2003. (cited on Page 259)

Robert A Gingold and Joseph J Monaghan. Smoothed particle hydrodynamics: theory and application to non-spherical stars. *Monthly notices of the royal astronomical society*, 181(3):375–389, 1977. (cited on Page 173)

A Golbabai and Sattar Seifollahi. Radial basis function networks in the numerical solution of linear integro-differential equations. *Applied Mathematics and Computation*, 188(1):427–432, 2007. (cited on Page 43)

A Golbabai, M Mammadov, and Sattar Seifollahi. Solving a system of nonlinear integral equations by an rbf network. *Computers & Mathematics with Applications*, 57(10):1651–1658, 2009. (cited on Page 43)

Herbert Goldstein. *Classical mechanics*. Pearson Education India, 2011. (cited on Page 44)

Benjamin Gompertz. On the nature of the function expressive of the law of human mortality, and on a new mode of determining the value of life contingencies. in a letter to francis baily, esq. frs &c. *Philosophical transactions of the Royal Society of London*, 24(115):513–583, 1825. (cited on Page 239)

David González, Alberto Badias, Icíar Alfaro, Francisco Chinesta, and Elias Cueto. Model order reduction for real-time data assimilation through extended kalman filters. *Computer Methods in Applied Mechanics and Engineering*, 326:679–693, 2017. (cited on Page 56)

David González, Francisco Chinesta, and Elías Cueto. Learning corrections for hyperelastic models from data. *Frontiers in Materials*, 6, feb 2019. ISSN 22968016. doi: 10.3389/fmats.2019.00014. (cited on Page 56 and 164)

David González, Francisco Chinesta, and Elías Cueto. Thermodynamically consistent data-driven computational mechanics. *Continuum Mechanics and Thermodynamics*, 31(1):239–253, 2019. (cited on Page 41, 43, and 164)

David González, Alberto García-González, Francisco Chinesta, and Elías Cueto. A data-driven learning method for constitutive modeling: application to vascular hyperelastic soft tissues. *Materials*, 13(10):2319, 2020. (cited on Page 43)

David González, Francisco Chinesta, and Elías Cueto. Learning non-markovian physics from data. *Journal of Computational Physics*, 428:109982, 2021. (cited on Page 43)

Francisco J. Gonzalez and Maciej Balajewicz. Deep convolutional recurrent autoencoders for learning low-dimensional feature dynamics of fluid systems, 2018. (cited on Page 36)

Raul González-García, Ramiro Rico-Martìnez, and Ioannis G Kevrekidis. Identification of distributed parameter systems: A neural net based approach. *Computers & chemical engineering*, 22:S965–S968, 1998. (cited on Page 41)

Marco Gori, Gabriele Monfardini, and Franco Scarselli. A new model for learning in graph domains. In *Proceedings. 2005 IEEE International Joint Conference on Neural Networks, 2005.*, volume 2, pages 729–734. IEEE, 2005. (cited on Page 37)

Richard J Goss. Hypertrophy versus hyperplasia. *Science*, 153(3744):1615–1620, 1966. (cited on Page 3)

Somdatta Goswami, Cosmin Anitescu, Souvik Chakraborty, and Timon Rabczuk. Transfer learning enhanced physics informed neural network for phase-field modeling of fracture. *Theoretical and Applied Fracture Mechanics*, 106:102447, 2020. (cited on Page 36)

Peter Gould. Letting the data speak for themselves. *Annals of the Association of American Geographers*, 71(2):166–176, 6 1981. (cited on Page 25 and 147)

Thore Graepel. Solving noisy linear operator equations by gaussian processes: Application to ordinary and partial differential equations. In *ICML*, volume 3, pages 234–241, 2003. (cited on Page 39)

John W Graham. Missing data analysis: Making it work in the real world. *Annual review of psychology*, 60:549–576, 2009. (cited on Page 102)

Michael D Graham and Ioannis G Kevrekidis. Alternative approaches to the karhunen-loeve decomposition for model reduction and data analysis. *Computers & chemical engineering*, 20(5):495–506, 1996. (cited on Page 34)

Robert B Gramacy. *Surrogates: Gaussian process modeling, design, and optimization for the applied sciences*. Chapman and Hall/CRC, 2020. (cited on Page 37)

Yves Grandvalet, Stéphane Canu, and Stéphane Boucheron. Noise injection: Theoretical prospects. *Neural Computation*, 9(5):1093–1108, 1997. (cited on Page 194)

Florian R Greten and Sergei I Grivennikov. Inflammation and cancer: triggers, mechanisms, and consequences. *Immunity*, 51(1):27–41, 2019. (cited on Page 317)

Andreas Griewank et al. On automatic differentiation. *Mathematical Programming: recent developments and applications*, 6(6):83–107, 1989. (cited on Page 147)

Corinne E Griguer, Claudia R Oliva, Eric Gobin, Pascale Marcorelles, Dale J Benos, Jack R Lancaster Jr, and G Yancey Gillespie. Cd133 is a marker of bioenergetic stress in human glioma. *PLoS One*, 3(11):e3655, 2008. (cited on Page 261 and 262)

Miroslav Grmela. Why generic? *Journal of non-newtonian fluid mechanics*, 165 (17-18):980–986, 2010. (cited on Page 41 and 164)

Miroslav Grmela and Hans Christian Öttinger. Dynamics and thermodynamics of complex fluids. i. development of a general formalism. *Physical Review E*, 56(6): 6620, 1997. (cited on Page 41)

J Grote, R Süsskind, and P Vaupel. Oxygen diffusivity in tumor tissue (ds-carcinosarcoma) under temperature conditions within the range of 20–40 c. *Pflügers Archiv*, 372(1):37–42, 1977. (cited on Page 261)

David J Guckenberger, Theodorus E de Groot, Alwin MD Wan, David J Beebe, and Edmond WK Young. Micromilling: a method for ultra-rapid prototyping of plastic microfluidic devices. *Lab on a Chip*, 15(11):2364–2378, 2015. (cited on Page 9)

Ruey-Shiang Guh. Robustness of the neural network based control chart pattern recognition system to non-normality. *International Journal of Quality & Reliability Management*, 19(1):97–112, 2002. (cited on Page 152)

Mamikon Gulian, Maziar Raissi, Paris Perdikaris, and George Karniadakis. Machine learning of space-fractional differential equations. *SIAM Journal on Scientific Computing*, 41(4):A2485–A2509, 2019. (cited on Page 39)

Mamikon Gulian, Ari Frankel, and Laura Swiler. Gaussian process regression constrained by boundary value problems. *Computer Methods in Applied Mechanics and Engineering*, 388:114117, 2022. (cited on Page 39)

Antonio Gulli and Sujit Pal. *Deep learning with Keras*. Packt Publishing Ltd, 2017. (cited on Page 145)

Pietro M Gullino, Flora H Grantham, and Anita H Courtney. Utilization of oxygen by transplanted tumors in vivo. *Cancer Research*, 27(6 Part 1):1020–1030, 1967. (cited on Page 261)

David Gunning and David Aha. Darpa's explainable artificial intelligence (xai) program. *AI Magazine*, 40(2):44–58, 2019. (cited on Page 12)

David Gunning, Mark Stefik, Jaesik Choi, Timothy Miller, Simone Stumpf, and Guang-Zhong Yang. Xai—explainable artificial intelligence. *Science Robotics*, 4 (37), 2019. (cited on Page 12)

Hongwei Guo, Xiaoying Zhuang, and Timon Rabczuk. A deep collocation method for the bending analysis of kirchhoff plate. *arXiv preprint arXiv:2102.02617*, 2021. (cited on Page 48)

Anna Gustafsson, Neashan Mathavan, Mikael J Turunen, Jonas Engqvist, Hanifeh Khayyeri, Stephen A Hall, and Hanna Isaksson. Linking multiscale deformation to microstructure in cortical bone using in situ loading, digital image correlation and synchrotron x-ray scattering. *Acta biomaterialia*, 69:323–331, 2018. (cited on Page 127)

Dan Haesloop and Bradley R Holt. A neural network structure for system identification. In *1990 American Control Conference*, pages 2460–2465. IEEE, 1990. (cited on Page 40)

Ehsan Haghighat, Maziar Raissi, Adrian Moure, Hector Gomez, and Ruben Juanes. A physics-informed deep learning framework for inversion and surrogate modeling in solid mechanics. *Computer Methods in Applied Mechanics and Engineering*, 379: 113741, 2021. (cited on Page 49)

Alastair R Hall et al. *Generalized method of moments*. Oxford university press, 2005. (cited on Page 81)

Peter Hall. On kullback-leibler loss and density estimation. *The Annals of Statistics*, pages 1491–1519, 1987. (cited on Page 49)

Ridha Hambli, Abdessalam Chamekh, and Hédi Bel Hadj Salah. Real-time deformation of structure using finite element and neural networks in virtual reality applications. *Finite elements in analysis and design*, 42(11):985–991, 2006. (cited on Page 158)

Jiequn Han, Arnulf Jentzen, and E Weinan. Solving high-dimensional partial differential equations using deep learning. *Proceedings of the National Academy of Sciences*, 115(34):8505–8510, 2018. (cited on Page 44, 65, and 151)

Jiequn Han, Linfeng Zhang, and E Weinan. Solving many-electron schrödinger equation using deep neural networks. *Journal of Computational Physics*, 399: 108929, 2019. (cited on Page 44)

Douglas Hanahan and Robert A Weinberg. The hallmarks of cancer. *cell*, 100(1): 57–70, 2000. (cited on Page 4)

Douglas Hanahan and Robert A Weinberg. Hallmarks of cancer: the next generation. *cell*, 144(5):646–674, 2011. (cited on Page 4 and 6)

Boris Hanin. Universal function approximation by deep neural nets with bounded width and relu activations. *arXiv preprint arXiv:1708.02691*, 2017. (cited on Page 47 and 151)

MM Hansen, T Miron-Shatz, AYS Lau, and C Paton. Big data in science and healthcare: a review of recent literature and perspectives. *Yearbook of medical informatics*, 23(01):21–26, 2014. (cited on Page 11)

Kevin J Harrington. Biology of cancer. *Medicine*, 39(12):689–692, 2011. (cited on Page 4)

HO Hartley and RR Hocking. The analysis of incomplete data. *Biometrics*, pages 783–823, 1971. (cited on Page 103)

Leith Hathout, Benjamin Ellingson, and Whitney Pope. Modeling the efficacy of the extent of surgical resection in the setting of radiation therapy for glioblastoma. *Cancer Science*, 107(8):1110–1116, 2016. (cited on Page 258, 259, 260, and 262)

Haralampos Hatzikirou, Andreas Deutsch, Carlo Schaller, Matthias Simon, and Kristin Swanson. Mathematical modelling of glioblastoma tumour development: a review. *Mathematical Models and Methods in Applied Sciences*, 15(11):1779–1794, 2005. (cited on Page 10 and 234)

Haralampos Hatzikirou, David Basanta, Matthias Simon, K Schaller, and Andreas Deutsch. 'go or grow': the key to the emergence of invasion in tumour progression? *Mathematical medicine and biology: a journal of the IMA*, 29(1):49–65, 2012. (cited on Page 255)

Donald T Haynie. *Biological thermodynamics*. Cambridge University Press, 2001. (cited on Page 315)

Qizhi He and Jiun-Shyan Chen. A physics-constrained data-driven approach based on locally convex reconstruction for noisy database. *Computer Methods in Applied Mechanics and Engineering*, 363:112791, 2020. (cited on Page 58)

Qizhi He, Devin W Laurence, Chung-Hao Lee, and Jiun-Shyan Chen. Manifold learning based data-driven modeling for soft biological tissues. *Journal of Biomechanics*, 117:110124, 2021a. (cited on Page 58)

Shouling He, Konrad Reif, and Rolf Unbehauen. Multilayer neural networks for solving a class of partial differential equations. *Neural networks*, 13(3):385–396, 2000. (cited on Page 43)

Xiaolong He, Qizhi He, and Jiun-Shyan Chen. Deep autoencoders for physics-constrained data-driven nonlinear materials modeling. *Computer Methods in Applied Mechanics and Engineering*, 385:114034, 2021b. (cited on Page 60)

Yousef Heider, Kun Wang, and WaiChing Sun. So (3)-invariance of informed-graph-based deep neural network for anisotropic elastoplastic materials. *Computer Methods in Applied Mechanics and Engineering*, 363:112875, 2020. (cited on Page 41)

Suzana Herculano-Houzel and Roberto Lent. Isotropic fractionator: a simple, rapid method for the quantification of total cell and neuron numbers in the brain. *Journal of Neuroscience*, 25(10):2518–2521, 2005. (cited on Page 260)

Quercus Hernandez, Alberto Badias, David Gonzalez, Francisco Chinesta, and Elias Cueto. Deep learning of thermodynamics-aware reduced-order models from data. *Computer Methods in Applied Mechanics and Engineering*, 379:113763, 2021. (cited on Page 43)

Quercus Hernández, Alberto Badías, David González, Francisco Chinesta, and Elías Cueto. Structure-preserving neural networks. *Journal of Computational Physics*, 426:109950, 2021. (cited on Page 43)

Mireille Hildebrandt. Law as information in the era of data-driven agency. *The Modern Law Review*, 79(1):1–30, 2016. (cited on Page 12)

Shawndra Hill, Foster Provost, and Chris Volinsky. Network-based marketing: Identifying likely adopters via consumer networks. *Statistical Science*, pages 256–276, 2006. (cited on Page 24)

Jessica Hoarau-Véchot, Arash Rafii, Cyril Touboul, and Jennifer Pasquier. Halfway between 2d and animal models: are 3d cultures the ideal tool to study cancer-microenvironment interactions? *International journal of molecular sciences*, 19(1): 181, 2018. (cited on Page 285)

Arthur E Hoerl and Robert W Kennard. Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12(1):55–67, 1970. (cited on Page 45)

Moritz Hoffmann, Christoph Fröhner, and Frank Noé. Reactive sindy: Discovering governing reactions from concentration data. *The Journal of chemical physics*, 150 (2):025101, 2019. (cited on Page 52)

BHJ Hofstee. Non-inverted versus inverted plots in enzyme kinetics. *Nature*, 184 (4695):1296–1298, 1959. (cited on Page 297)

Lasse Holmstrom and Petri Koistinen. Using additive noise in back-propagation training. *IEEE transactions on neural networks*, 3(1):24–38, 1992. (cited on Page 194)

Kurt Hornik. Approximation capabilities of multilayer feedforward networks. *Neural networks*, 4(2):251–257, 1991. (cited on Page 47, 151, and 305)

Le Hou, Dimitris Samaras, Tahsin M Kurc, Yi Gao, and Joel H Saltz. Convnets with smooth adaptive activation functions for regression. *Proceedings of machine learning research*, 54:430, 2017. (cited on Page 224)

Weronika Hryniewska, Przemysław Bombiński, Patryk Szatkowski, Paulina Tomaszewska, Artur Przelaskowski, and Przemysław Biecek. Do not repeat these mistakes–a critical appraisal of applications of explainable artificial intelligence for image based covid-19 detection. *arXiv preprint arXiv:2012.08333*, 2020. (cited on Page 12)

Daniel Z Huang, Kailai Xu, Charbel Farhat, and Eric Darve. Learning constitutive relations from indirect observations using deep neural networks. *Journal of Computational Physics*, 416:109491, 2020. (cited on Page 61, 354, and 370)

Sui Huang and Donald E Ingber. Cell tension, matrix mechanics, and cancer development. *Cancer cell*, 8(3):175–176, 2005. (cited on Page 235)

John J Hunter and Kenneth R Chien. Signaling pathways for cardiac hypertrophy and failure. *New England Journal of Medicine*, 341(17):1276–1283, 1999. (cited on Page 3)

Côme Huré, Huyên Pham, and Xavier Warin. Some machine learning schemes for high-dimensional nonlinear pdes. arxiv. *arXiv preprint arXiv:1902.01599*, 2019. (cited on Page 44)

Martin Hutzenthaler, Arnulf Jentzen, Thomas Kruse, and Tuan Anh Nguyen. A proof that rectified deep neural networks overcome the curse of dimensionality in the numerical approximation of semilinear heat equations. *SN partial differential equations and applications*, 1(2):1–34, 2020. (cited on Page 49)

Rob J Hyndman. Computing and graphing highest density regions. *The American Statistician*, 50(2):120–126, 1996. (cited on Page 246)

Ruben Ibañez, Domenico Borzacchiello, Jose Vicente Aguado, Emmanuelle Abisset-Chavanne, Elias Cueto, Pierre Ladeveze, and Francisco Chinesta. Data-driven non-linear elasticity: constitutive manifold construction and problem discretization. *Computational Mechanics*, 60(5):813–826, 2017. (cited on Page 60)

Rubén Ibanez, Emmanuelle Abisset-Chavanne, Jose Vicente Aguado, David Gonzalez, Elias Cueto, and Francisco Chinesta. A manifold learning approach to data-driven computational elasticity and inelasticity. *Archives of Computational Methods in Engineering*, 25(1):47–57, 2018. (cited on Page 60, 70, 71, 102, and 163)

Rubén Ibáñez, Emmanuelle Abisset-Chavanne, David González, Jean-Louis Duval, Elias Cueto, and Francisco Chinesta. Hybrid constitutive modeling: data-driven learning of corrections to plasticity models. *International Journal of Material Forming*, 12(4):717–725, 2019. (cited on Page 56 and 60)

Sunyoung Im, Jonggeon Lee, and Maenghyo Cho. Surrogate modeling of elasto-plastic problems via long short-term memory neural networks and proper orthogonal decomposition. *Computer Methods in Applied Mechanics and Engineering*, 385: 114030, 2021. (cited on Page 36)

Hanna Isaksson, Olivier Comas, Corrinus C van Donkelaar, Jesus Mediavilla, Wouter Wilson, Rik Huiskes, and Keita Ito. Bone regeneration during distraction osteogenesis: mechano-regulation by shear strain and fluid velocity. *Journal of biomechanics*, 40(9):2002–2011, 2007. (cited on Page 143)

R Ishizuka, S-H Chong, and F Hirata. An integral equation theory for inhomogeneous molecular fluids: The reference interaction site model approach. *The Journal of chemical physics*, 128(3):034504, 2008. (cited on Page 204)

Leon Isserlis. On a formula for the product-moment coefficient of any order of a normal frequency distribution in any number of variables. *Biometrika*, 12(1/2): 134–139, 1918. (cited on Page 396)

DJ J. Toal, Neil W Bressloff, and Andy J Keane. Kriging hyperparameter tuning strategies. *AIAA journal*, 46(5):1240–1252, 2008. (cited on Page 227)

Nicholas E Jackson, Michael A Webb, and Juan J de Pablo. Recent advances in machine learning towards multiscale soft materials design. *Current Opinion in Chemical Engineering*, 23:106–114, 2019. (cited on Page 37)

Rudolf Jaenisch and Adrian Bird. Epigenetic regulation of gene expression: how the genome integrates intrinsic and environmental signals. *Nature genetics*, 33(3): 245–254, 2003. (cited on Page 350, 364, and 381)

Ameya D Jagtap, Ehsan Kharazmi, and George Em Karniadakis. Conservative physics-informed neural networks on discrete domains for conservation laws: Applications to forward and inverse problems. *Computer Methods in Applied Mechanics and Engineering*, 365:113028, 2020. (cited on Page 49)

Gareth M James, Peter Radchenko, and Jinchi Lv. Dasso: connections between the dantzig selector and lasso. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 71(1):127–142, 2009. (cited on Page 45)

Angela M Jarrett, Ernesto ABF Lima, David A Hormuth, Matthew T McKenna, Xinzeng Feng, David A Ekrut, Anna Claudia M Resende, Amy Brock, and Thomas E Yankeelov. Mathematical models of tumor cell proliferation: a review of the literature. *Expert review of anticancer therapy*, 18(12):1271–1286, 2018. (cited on Page 239)

Piotr Jaworski, Fabrizio Durante, Wolfgang Karl Härdle, and Tomasz Rychlik. *Copula theory and its applications: proceedings of the workshop held in Warsaw, 25-26*

September 2009, volume 198. Springer Science & Business Media, 2010. (cited on Page 244)

Randy L Jensen. Brain tumor hypoxia: tumorigenesis, angiogenesis, imaging, pseudoprogression, and as a therapeutic target. *Journal of neuro-oncology*, 92(3):317–335, 2009. (cited on Page 7 and 8)

Jessie S Jeon, Simone Bersini, Mara Gilardi, Gabriele Dubini, Joseph L Charest, Matteo Moretti, and Roger D Kamm. Human 3d vascularized organotypic microfluidic assays to study breast cancer cell extravasation. *Proceedings of the National Academy of Sciences*, 112(1):214–219, 2015. (cited on Page 9)

Weiqi Ji, Weilun Qiu, Zhiyu Shi, Shaowu Pan, and Sili Deng. Stiff-pinn: Physics-informed neural network for stiff chemical kinetics. *The Journal of Physical Chemistry A*, 125(36):8098–8106, 2021. (cited on Page 49)

Xiaowei Jia, Jared Willard, Anuj Karpatne, Jordan Read, Jacob Zwart, Michael Steinbach, and Vipin Kumar. Physics guided rnns for modeling dynamical systems: A case study in simulating lake temperature profiles. In *Proceedings of the 2019 SIAM International Conference on Data Mining*, pages 558–566. SIAM, 2019. (cited on Page 46)

Chiyu Jiang, Jingwei Huang, Karthik Kashinath, Philip Marcus, Matthias Niessner, et al. Spherical cnns on unstructured grids. *arXiv preprint arXiv:1901.02039*, 2019. (cited on Page 37)

Li Jianyu, Luo Siwei, Qi Yingjian, and Huang Yaping. Numerical solution of elliptic partial differential equation using radial basis function neural networks. *Neural Networks*, 16(5-6):729–734, 2003. (cited on Page 43)

Kam-Chuen Jim, C Lee Giles, and Bill G Horne. An analysis of noise in recurrent neural networks: convergence and generalization. *IEEE Transactions on neural networks*, 7(6):1424–1438, 1996. (cited on Page 194)

José A Jiménez-Torres, Stephen L Peery, Kyung E Sung, and David J Beebe. Lumenext: a practical method to pattern luminal structures in ecm gels. *Advanced healthcare materials*, 5(2):198–204, 2016. (cited on Page 9)

Ian Jolliffe. *Principal component analysis*. Wiley Online Library, 2002. (cited on Page 34)

Justin V Joseph, Siobhan Conroy, Kirill Pavlov, Pallavi Sontakke, Tushar Tomar, Ellie Eggens-Meijer, Veerakumar Balasubramaniyan, Michiel Wagemakers, Wilfred FA den Dunnen, and Frank AE Kruyt. Hypoxia enhances migration and invasion in glioblastoma by promoting a mesenchymal shift mediated by the hif1α–zeb1 axis. *Cancer letters*, 359(1):107–116, 2015. (cited on Page 289)

Norman P Jouppi, Cliff Young, Nishant Patil, David Patterson, Gaurav Agrawal, Raminder Bajwa, Sarah Bates, Suresh Bhatia, Nan Boden, Al Borchers, et al. In-datacenter performance analysis of a tensor processing unit. In *Proceedings of the 44th annual international symposium on computer architecture*, pages 1–12, 2017. (cited on Page 152)

Sungmoon Jung and Jamshid Ghaboussi. Neural network constitutive model for rate-dependent materials. *Computers & Structures*, 84(15-16):955–963, 2006. (cited on Page 36)

Heikki Junninen, Harri Niska, Kari Tuppurainen, Juhani Ruuskanen, and Mikko Kolehmainen. Methods for imputation of missing values in air quality data sets. *Atmospheric Environment*, 38(18):2895–2907, 2004. (cited on Page 102)

David Kahaner, Cleve Moler, and Stephen Nash. Numerical methods and software. *Englewood Cliffs: Prentice Hall, 1989*, 1989. (cited on Page 122)

Eurika Kaiser, J Nathan Kutz, and Steven L Brunton. Sparse identification of nonlinear dynamics for model predictive control in the low-data limit. *Proceedings of the Royal Society A*, 474(2219):20180335, 2018. (cited on Page 52)

Elias Kalapanidas, Nikolaos Avouris, Marian Craciun, and Daniel Neagu. Machine learning algorithms: a study on noise sensitivity. In *Proc. 1st Balcan Conference in Informatics*, pages 356–365, 2003. (cited on Page 194)

Rudolph Emil Kalman. A new approach to linear filtering and prediction problems. *Journal of basic Engineering*, 82(1):35–45, 1960. (cited on Page 40 and 55)

Yoshihiro Kanno. Data-driven computing in elasticity via kernel regression. *Theoretical and Applied Mechanics Letters*, 8(6):361–365, 2018. (cited on Page 58)

EJ Kansa, H Power, GE Fasshauer, and L Ling. A volumetric integral radial basis function method for time-dependent partial differential equations. i. formulation. *Engineering Analysis with Boundary Elements*, 28(10):1191–1206, 2004. (cited on Page 43)

Alan A Kaptanoglu, Kyle D Morgan, Chris J Hansen, and Steven L Brunton. Physics-constrained, low-dimensional models for magnetohydrodynamics: First-principles and data-driven approaches. *Physical Review E*, 104(1):015206, 2021. (cited on Page 50)

K Karapiperis, L Stainier, M Ortiz, and JE Andrade. Data-driven multiscale modeling in mechanics. *Journal of the Mechanics and Physics of Solids*, 147:104239, 2021a. (cited on Page 58)

Konstantinos Karapiperis, Michael Ortiz, and José E Andrade. Data-driven nonlocal mechanics: Discovering the internal length scales of materials. *Computer Methods in Applied Mechanics and Engineering*, 386:114039, 2021b. (cited on Page 58)

Kari Karhunen. Under lineare methoden in der wahr scheinlichkeitsrechnung. *Annales Academiae Scientiarun Fennicae Series A1: Mathematia Physica*, 47, 1947. (cited on Page 34)

Sadegh Karimpouli and Pejman Tahmasebi. Physics informed machine learning: Seismic wave equation. *Geoscience Frontiers*, 11(6):1993–2001, 2020. (cited on Page 39)

Anuj Karpatne, Gowtham Atluri, James Faghmous, Michael Steinbach, Arindam Banerjee, Auroop Ganguly, Shashi Shekhar, Nagiza Samatova, Ajith Kumar, and Vipin V. Theory-guided data science: A new paradigm for scientific discovery. *IEEE Transactions on Knowledge and Data Engineering*, 29(10):2318–2331, 2017a. doi: 10.1109/TKDE.2017.2720168. (cited on Page 152 and 163)

Anuj Karpatne, Gowtham Atluri, James H Faghmous, Michael Steinbach, Arindam Banerjee, Auroop Ganguly, Shashi Shekhar, Nagiza Samatova, and Vipin Kumar. Theory-guided data science: A new paradigm for scientific discovery from data. *IEEE Transactions on knowledge and data engineering*, 29(10):2318–2331, 2017b. (cited on Page 46)

Anuj Karpatne, William Watkins, Jordan Read, and Vipin Kumar. Physics-guided neural networks (pgnn): An application in lake temperature modeling. *arXiv preprint arXiv:1710.11431*, 2017c. (cited on Page 46, 146, and 158)

Moriah E Katt, Amanda L Placone, Andrew D Wong, Zinnia S Xu, and Peter C Searson. In vitro tumor models: advantages, disadvantages, variables, and selecting the right platform. *Frontiers in bioengineering and biotechnology*, 4:12, 2016. (cited on Page 10 and 321)

Balveen Kaur, Fatima W Khwaja, Eric A Severson, Shannon L Matheny, Daniel J Brat, and Erwin G Van Meir. Hypoxia and the hypoxia-inducible-factor pathway in glioma growth and angiogenesis. *Neuro-oncology*, 7(2):134–153, 2005. (cited on Page 286)

Li Dao Ke, Yue-Xi Shi, Seock-Ah Im, Xiashan Chen, and WK Alfred Yung. The relevance of cell proliferation, vascular endothelial growth factor, and basic fibroblast growth factor production to angiogenesis and tumorigenicity in human glioma cell lines. *Clinical Cancer Research*, 6(6):2562–2572, 2000. (cited on Page 259)

EA Kearsley and LJ Zapas. Some methods of measurement of an elastic strain-energy function of the valanis-landel type. *Journal of Rheology*, 24(4):483–500, 1980. (cited on Page 59)

Marc C Kennedy and Anthony O'Hagan. Predicting the output from a complex computer code when fast approximations are available. *Biometrika*, 87(1):1–13, 2000. (cited on Page 37)

Xavier M Keutgen, Filippo Filicori, Michael J Crowley, Yongchun Wang, Theresa Scognamiglio, Rana Hoda, Daniel Buitrago, David Cooper, Martha A Zeiger, Rasa Zarnegar, et al. A panel of four mirnas accurately differentiates malignant from benign indeterminate thyroid lesions on fine needle aspiration. *Clinical Cancer Research*, 18(7):2032–2038, 2012. (cited on Page 11)

Shahzad Khan. Introduction to machine learning (adaptive computation and machine learning series). *Natural Language Engineering*, 14(01):133–137, 2008. (cited on Page 24)

Boris N Kholodenko. Cell-signalling dynamics in time and space. *Nature reviews Molecular cell biology*, 7(3):165–176, 2006. (cited on Page 315)

Yuehaw Khoo, Jianfeng Lu, and Lexing Ying. Solving for high-dimensional committor functions using artificial neural networks. *Research in the Mathematical Sciences*, 6(1):1–13, 2019. (cited on Page 44)

Yuehaw Khoo, Jianfeng Lu, and Lexing Ying. Solving parametric pde problems with artificial neural networks. *European Journal of Applied Mathematics*, 32(3): 421–435, 2021. (cited on Page 49)

Roger T Kilgore and David B Thompson. Estimating joint flow probabilities at stream confluences by using copulas. *Transportation Research Record*, 2262(1): 200–206, 2011. (cited on Page 235)

Jong-Min Kim, Yoon-Sung Jung, Engin A Sungur, Kap-Hoon Han, Changyi Park, and Insuk Sohn. A copula method for modeling directional dependence of genes. *BMC bioinformatics*, 9(1):225, 2008. (cited on Page 235)

Yangjin Kim, Hyejin Jeon, and Hans Othmer. The role of the tumor microenvironment in glioblastoma: A mathematical model. *IEEE Transactions on Biomedical Engineering*, 64(3):519–527, 2016. (cited on Page 10, 234, and 241)

Youngkyu Kim, Youngsoo Choi, David Widemann, and Tarek Zohdi. A fast and accurate physics-informed neural network reduced order model with shallow masked autoencoder. *Journal of Computational Physics*, page 110841, 2021. (cited on Page 36)

Hiroyuki Kimura, Rod D Braun, Edgardo T Ong, Richard Hsu, Timothy W Secomb, Demetrios Papahadjopoulos, Keelung Hong, and Mark W Dewhirst. Fluctuations in red cell flux in tumor microvessels can lead to transient hypoxia and reoxygenation in tumor parenchyma. *Cancer Research*, 56(23):5522–5528, 1996. (cited on Page 259)

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. (cited on Page 205 and 307)

Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *CoRR*, abs/1609.02907, 2016. URL http://arxiv.org/abs/1609.02907. (cited on Page 37)

Trenton Kirchdoerfer and Michael Ortiz. Data-driven computational mechanics. *Computer Methods in Applied Mechanics and Engineering*, 304:81–101, 2016a. (cited on Page 353 and 369)

Trenton Kirchdoerfer and Michael Ortiz. Data-driven computational mechanics. *Computer Methods in Applied Mechanics and Engineering*, 304:81–101, 2016b. (cited on Page 17, 57, 64, 70, 71, 72, 73, 89, 95, 102, 117, 130, 146, and 164)

Trenton Kirchdoerfer and Michael Ortiz. Data driven computing with noisy material data sets. *arXiv preprint arXiv:1702.01574*, 2017. (cited on Page 58, 130, 146, 353, and 369)

Trenton Kirchdoerfer and Michael Ortiz. Data-driven computing in dynamics. *International Journal for Numerical Methods in Engineering*, 113(11):1697–1710, 2018. (cited on Page 58)

Hiroaki Kitano. Computational systems biology. *Nature*, 420(6912):206, 2002. (cited on Page 321)

Rob Kitchin. Big data and human geography: Opportunities, challenges and risks. *Dialogues in Human Geography*, 3:262–267, 12 2013. doi: doi:10.1177/2043820613513388. (cited on Page 25)

Rob Kitchin. Big data, new epistemologies and paradigm shifts. *Big Data & Society*, 1:1–12, 2014. doi: doi:10.1177/2053951714528481. (cited on Page 25)

Fima C Klebaner. *Introduction to stochastic calculus with applications*. World Scientific Publishing Company, 2012. (cited on Page 43)

Dominic Kohler, Youssef M Marzouk, Johannes Müller, and Utz Wever. A new network approach to bayesian inference in partial differential equations. *International Journal for Numerical Methods in Engineering*, 104(5):313–329, 2015. (cited on Page 36)

Teuvo Kohonen. The self-organizing map. *Proceedings of the IEEE*, 78(9):1464–1480, 1990. (cited on Page 59 and 82)

Erik Kole, Kees Koedijk, and Marno Verbeek. Selecting copulas for risk management. *Journal of Banking & Finance*, 31(8):2405–2423, 2007. (cited on Page 235 and 266)

Bernard O Koopman. Hamiltonian systems and transformation in hilbert space. *Proceedings of the national academy of sciences of the united states of america*, 17 (5):315, 1931. (cited on Page 52)

Nathabandu T Kottegoda and Renzo Rosso. *Applied statistics for civil and environmental engineers*. Blackwell Malden, MA, 2008. (cited on Page 246)

V Kouznetsova, WAM Brekelmans, and FPT Baaijens. An approach to micro-macro modeling of heterogeneous materials. *Computational Mechanics*, 27(1):37–48, 2001. (cited on Page 92)

Varvara Kouznetsova, Marc GD Geers, and WA Marcel Brekelmans. Multi-scale constitutive modelling of heterogeneous materials with a gradient-enhanced computational homogenization scheme. *International journal for numerical methods in engineering*, 54(8):1235–1260, 2002. (cited on Page 126)

Mark A Kramer. Nonlinear principal component analysis using autoassociative neural networks. *AIChE journal*, 37(2):233–243, 1991. (cited on Page 34 and 36)

Andrej Krenker, Andrej Kos, and Janez Bešter. *Introduction to the artificial neural networks*. INTECH Open Access Publisher, 2011. (cited on Page 24)

A Krogh. The anatomy and physiology of capillaries. new haven, 1929. *FREDERICK SMITH AND PEYTON ROUS*, 513, 1922. (cited on Page 261)

MG Kulkarni, K Matouš, and PH Geubelle. Coupled multi-scale cohesive modeling of failure in heterogeneous adhesives. *International Journal for Numerical Methods in Engineering*, 84(8):916–946, 2010. (cited on Page 126)

Vinay Kumar, Abul K Abbas, Nelson Fausto, and Jon C Aster. *Robbins and Cotran pathologic basis of disease, professional edition e-book*. Elsevier health sciences, 2014. (cited on Page 3)

Gitta Kutyniok, Philipp Petersen, Mones Raslan, and Reinhold Schneider. A theoretical analysis of deep neural networks and parametric pdes. *Constructive Approximation*, pages 1–53, 2021. (cited on Page 49)

J Nathan Kutz. Deep learning in fluid dynamics. *Journal of Fluid Mechanics*, 814: 1–4, 2017. (cited on Page 37)

JT-Y Kwok and IW-H Tsang. The pre-image problem in kernel methods. *IEEE transactions on neural networks*, 15(6):1517–1525, 2004. (cited on Page 60)

Pierre Ladevèze. The large time increment method for the analysis of structures with non-linear behavior described by internal variables. *COMPTES RENDUS DE L ACADEMIE DES SCIENCES SERIE II*, 309(11):1095–1099, 1989. (cited on Page 60, 70, and 102)

Pierre Ladeveze and Ludovic Chamoin. On the verification of model reduction methods based on the proper generalized decomposition. *Computer Methods in Applied Mechanics and Engineering*, 200(23-24):2032–2047, 2011. (cited on Page 40)

Pierre Ladevèze, David Néron, and Paul-William Gerbaud. Data-driven computation for history-dependent materials. *Comptes Rendus Mécanique*, 347(11):831–844, 2019. (cited on Page 58)

Isaac E Lagaris, Aristidis Likas, and Dimitrios I Fotiadis. Artificial neural networks for solving ordinary and partial differential equations. *IEEE transactions on neural networks*, 9(5):987–1000, 1998. (cited on Page 47)

Isaac E Lagaris, Aristidis C Likas, and Dimitris G Papageorgiou. Neural-network methods for boundary value problems with irregular boundaries. *IEEE Transactions on Neural Networks*, 11(5):1041–1049, 2000. (cited on Page 47)

John H Lagergren, John T Nardini, Ruth E Baker, Matthew J Simpson, and Kevin B Flores. Biologically-informed neural networks guide mechanistic modeling from sparse experimental data. *PLoS computational biology*, 16(12):e1008462, 2020. (cited on Page 49, 294, 354, and 370)

Zhilu Lai and Satish Nagarajaiah. Sparse structural system identification method for nonlinear dynamic systems with hysteresis/inelastic behavior. *Mechanical Systems and Signal Processing*, 117:813–842, 2019. (cited on Page 52)

Anna Kane Laird. Dynamics of tumour growth. *British Journal of Cancer*, 18(3): 490, 1964. (cited on Page 259)

Remi R Lam, Lior Horesh, Haim Avron, and Karen E Willcox. Should you derive, or let the data drive? an optimization framework for hybrid first-principles data-driven modeling. *arXiv preprint arXiv:1711.04374*, 2017. (cited on Page 56)

Sandra F Lam, Venktesh S Shirure, Yunli E Chu, Alan G Soetikno, and Steven C George. Microfluidic device to attain high spatial and temporal control of oxygen. *PLoS One*, 13(12):e0209574, 2018. (cited on Page 299)

John Denholm Lambert. *Numerical methods for ordinary differential systems: the initial value problem*. John Wiley & Sons, Inc., 1991. (cited on Page 39 and 301)

Hans Petter Langtangen. *Computational partial differential equations: numerical methods and diffpack programming*, volume 2. Springer Berlin, 1999. (cited on Page 30 and 165)

Mathieu Laplante and David M Sabatini. mtor signaling at a glance. *Journal of cell science*, 122(20):3589–3594, 2009. (cited on Page 315)

Daniel T Larose. *Discovering knowledge in data: an introduction to data mining*. John Wiley & Sons, 2014. (cited on Page 24)

S. Larsson and V. Thomee. *Partial Differential Equations with Numerical Methods*. Springer Verlag, Belin-Heidelberg, 2009. (cited on Page 30, 147, and 164)

Marcos Latorre and Francisco J Montáns. Strain-level dependent nonequilibrium anisotropic viscoelasticity: Application to the abdominal muscle. *Journal of biomechanical engineering*, 139(10), 2017a. (cited on Page 59)

Marcos Latorre and Francisco J Montáns. Wypiwyg hyperelasticity without inversion formula: Application to passive ventricular myocardium. *Computers & Structures*, 185:47–58, 2017b. (cited on Page 59)

Marcos Latorre and Francisco J Montáns. Experimental data reduction for hyperelasticity. *Computers & Structures*, 232:105919, 2020. (cited on Page 59)

Marcos Latorre and Francisco Javier Montáns. Extension of the sussman–bathe spline-based hyperelastic model to incompressible transversely isotropic materials. *Computers & Structures*, 122:13–26, 2013. (cited on Page 59 and 70)

Marcos Latorre and Francisco Javier Montáns. What-you-prescribe-is-what-you-get orthotropic hyperelasticity. *Computational Mechanics*, 53(6):1279–1298, 2014. (cited on Page 59, 70, and 164)

Marcos Latorre, Estefanía Peña, and Francisco J Montáns. Determination and finite element validation of the wypiwyg strain energy of superficial fascia from experimental data. *Annals of biomedical engineering*, 45(3):799–810, 2017. (cited on Page 59)

Patrick Laux, Sven Wagner, A Wagner, Jucundus Jacobeit, A Bardossy, and Harald Kunstmann. Modelling daily precipitation features in the volta basin of west africa. *International Journal of Climatology: A Journal of the Royal Meteorological Society*, 29(7):937–954, 2009. (cited on Page 235)

Patrick Laux, S Vogl, W Qiu, Hans Richard Knoche, and Harald Kunstmann. Copula-based statistical refinement of precipitation in rcm simulations over complex terrain. *Hydrology and Earth System Sciences*, 15(7):2401–2419, 2011. (cited on Page 235)

Inna Lavrik, Alexander Golks, and Peter H Krammer. Death receptor signaling. *Journal of cell science*, 118(2):265–267, 2005. (cited on Page 315)

Neil Lawrence and Aapo Hyvärinen. Probabilistic non-linear principal component analysis with gaussian process latent variable models. *Journal of machine learning research*, 6(11), 2005. (cited on Page 37)

BA Le, Julien Yvonnet, and Q-C He. Computational homogenization of nonlinear elastic materials using neural networks. *International Journal for Numerical Methods in Engineering*, 104(12):1061–1084, 2015. (cited on Page 37)

Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521(7553): 436–444, 2015. (cited on Page 11)

Y.C. LeCun. Deep learning hardware: Past, present, and future. In *Proceedings 2019 IEEE International Solid- State Circuits Conference - (ISSCC), San Francisco, CA, USA*, pages 12–19, 2019. doi: doi:10.1109/ISSCC.2019.8662396. (cited on Page 145)

Hyuk Lee and In Seok Kang. Neural algorithm for solving differential equations. *Journal of Computational Physics*, 91(1):110–131, 1990. (cited on Page 47)

John A Lee and Michel Verleysen. *Nonlinear dimensionality reduction*. Springer Science & Business Media, 2007. (cited on Page 24)

Kookjin Lee and Kevin T Carlberg. Model reduction of dynamical systems on nonlinear manifolds using deep convolutional autoencoders. *Journal of Computational Physics*, 404:108973, 2020. (cited on Page 34)

Su Yeon Lee, Min Kyung Ju, Hyun Min Jeon, Eui Kyong Jeong, Yig Ji Lee, Cho Hee Kim, Hye Gyeong Park, Song Iy Han, and Ho Sung Kang. Regulation of tumor progression by programmed necrosis. *Oxidative medicine and cellular longevity*, 2018, 2018. (cited on Page 285)

Tae-Hwy Lee, Halbert White, and Clive WJ Granger. Testing for neglected nonlinearity in time series models: A comparison of neural network methods and alternative tests. *Journal of Econometrics*, 56(3):269–290, 1993. (cited on Page 152)

Kin Fong Lei, Min-Hsien Wu, Che-Wei Hsu, and Yi-Dao Chen. Real-time and non-invasive impedimetric monitoring of cell proliferation and chemosensitivity in a perfusion 3d cell culture microfluidic chip. *Biosensors and Bioelectronics*, 51: 16–21, 2014. (cited on Page 297)

S. Leonelli. Introduction: Making sense of data-driven research in the biological and biomedical sciences. *Studies in History and Philosophy of Science Part C: Studies in History and Philosophy of Biological and Biomedical Sciences*, 43(1):1–3, 2012. ISSN 1369-8486. doi: https://doi.org/10.1016/j.shpsc.2011.10.001. URL https://www.sciencedirect.com/science/article/pii/S136984861100077X. Data-Driven Research in the Biological and Biomedical Sciences On Nature and Normativity: Normativity, Teleology, and Mechanism in Biological Explanation. (cited on Page 25)

Kenneth Levenberg. A method for the solution of certain non-linear problems in least squares. *Quarterly of applied mathematics*, 2(2):164–168, 1944. (cited on Page 214)

Adrien Leygue, Michel Coret, Julien Réthoré, Laurent Stainier, and Erwan Verron. Data-based derivation of material response. *Computer Methods in Applied Mechanics and Engineering*, 331:184–196, 2018. (cited on Page 58, 64, and 146)

Adrien Leygue, Rian Seghir, Julien Réthoré, Michel Coret, Erwan Verron, and Laurent Stainier. Non-parametric material state field extraction from full field measurements. *Computational Mechanics*, 64(2):501–509, 2019. (cited on Page 59)

Shanwu Li, Eurika Kaiser, Shujin Laima, Hui Li, Steven L Brunton, and J Nathan Kutz. Discovering time-varying aerodynamics of a prototype bridge by sparse identification of nonlinear dynamical systems. *Physical Review E*, 100(2):022220, 2019. (cited on Page 52)

Wei Li, Martin Z Bazant, and Juner Zhu. A physics-guided neural network framework for elastic plates: Comparison of governing equations-based and energy-based approaches. *Computer Methods in Applied Mechanics and Engineering*, 383: 113933, 2021. (cited on Page 48 and 49)

Guanghui Liang and K Chandrashekhara. Neural network based constitutive model for elastomeric foams. *Engineering structures*, 30(7):2002–2011, 2008. (cited on Page 36)

Yulei Liao and Pingbing Ming. Deep nitsche method: Deep ritz method with essential boundary conditions. *arXiv preprint arXiv:1912.01309*, 2019. (cited on Page 48)

Juan Lin, Shuwei Zhou, and Hongwei Guo. A deep collocation method for heat transfer in porous media: Verification from the finite element method. *Journal of Energy Storage*, 28:101280, 2020. (cited on Page 48)

Hans Lineweaver and Dean Burk. The determination of enzyme dissociation constants. *Journal of the American chemical society*, 56(3):658–666, 1934. (cited on Page 297)

Julia Ling, Andrew Kurzawski, and Jeremy Templeton. Reynolds averaged turbulence modelling using deep neural networks with embedded invariance. *Journal of Fluid Mechanics*, 807:155–166, 2016. (cited on Page 37 and 41)

Geert Litjens, Thijs Kooi, Babak Ehteshami Bejnordi, Arnaud Arindra Adiyoso Setio, Francesco Ciompi, Mohsen Ghafoorian, Jeroen Awm Van Der Laak, Bram Van Ginneken, and Clara I Sánchez. A survey on deep learning in medical image analysis. *Medical image analysis*, 42:60–88, 2017. (cited on Page 36)

Roderick JA Little and Donald B Rubin. Bayes and multiple imputation. *Statistical Analysis with Missing Data, Second Edition*, pages 200–220, 2002. (cited on Page 102)

Dehao Liu and Yan Wang. Multi-fidelity physics-constrained neural network and its application in materials modeling. *Journal of Mechanical Design*, 141(12), 2019. (cited on Page 49)

Jin Liu, Kai Wang, Shuangge Ma, and Jian Huang. Accounting for linkage disequilibrium in genome-wide association studies: a penalized regression method. *Statistics and its interface*, 6(1):99, 2013. (cited on Page 46)

Minliang Liu, Liang Liang, and Wei Sun. Estimation of in vivo constitutive parameters of the aortic wall using a machine learning approach. *Computer methods in applied mechanics and engineering*, 347:201–217, 2019. (cited on Page 36)

Xin Liu, Fei Tao, and Wenbin Yu. A neural network enhanced system for learning nonlinear constitutive relation of fiber reinforced composites. In *AIAA Scitech 2020 Forum*, page 0396, 2020. (cited on Page 36)

Postma EO LJP and Herik HJ Van Den. Dimensionality reduction: A comparative review. *Tech. Rrep*, 2007. (cited on Page 24, 35, and 82)

Michel Loève. Probability theory ii. *Graduate texts in mathematics*, 46:15, 1978. (cited on Page 34)

Meghan Logun, Wujun Zhao, Leidong Mao, and Lohitash Karumbaiah. Microfluidics in malignant glioma research and precision medicine. *Advanced biosystems*, 2(5): 1700221, 2018. (cited on Page 285)

Jean-Christophe Loiseau and Steven L Brunton. Constrained sparse galerkin regression. *Journal of Fluid Mechanics*, 838:42–67, 2018. (cited on Page 52)

Jean-Christophe Loiseau, Bernd R Noack, and Steven L Brunton. Sparse reduced-order modelling: sensor-based dynamics to full-state estimation. *Journal of Fluid Mechanics*, 844:459–490, 2018. (cited on Page 52)

Zichao Long, Yiping Lu, Xianzhong Ma, and Bin Dong. Pde-net: Learning pdes from data. In *International Conference on Machine Learning*, pages 3208–3216. PMLR, 2018. (cited on Page 54)

Zichao Long, Yiping Lu, and Bin Dong. Pde-net 2.0: Learning pdes from data with a numeric-symbolic hybrid deep network. *Journal of Computational Physics*, 399: 108925, 2019. (cited on Page 54)

Elena Lopez, David Gonzalez, JV Aguado, Emmanuelle Abisset-Chavanne, Elias Cueto, Christophe Binetruy, and Francisco Chinesta. A manifold learning approach for integrated computational materials engineering. *Archives of Computational Methods in Engineering*, 25(1):59–68, 2018. (cited on Page 60 and 163)

Jianfeng Lu, Yulong Lu, and Min Wang. A priori generalization analysis of the deep ritz method for solving high dimensional elliptic equations. *arXiv preprint arXiv:2101.01708*, 2021a. (cited on Page 48)

Lu Lu, Xuhui Meng, Zhiping Mao, and George Em Karniadakis. Deepxde: A deep learning library for solving differential equations. *SIAM Review*, 63(1):208–228, 2021b. (cited on Page 49 and 146)

Xiaoxin Lu, Dimitris G Giovanis, Julien Yvonnet, Vissarion Papadopoulos, Fabrice Detrez, and Jinbo Bai. A data-driven computational homogenization method based on neural networks for the nonlinear anisotropic electrical response of graphene/polymer nanocomposites. *Computational Mechanics*, 64(2):307–321, 2019. (cited on Page 146)

Xin Lu and Yibin Kang. Hypoxia and hypoxia-inducible factors: master regulators of metastasis. *Clinical cancer research*, 16(24):5928–5935, 2010. (cited on Page 254 and 297)

Zhou Lu, Hongming Pu, Feicheng Wang, Zhiqiang Hu, and Liwei Wang. The expressive power of neural networks: A view from the width. In *Advances in neural information processing systems*, pages 6231–6239, 2017. (cited on Page 47 and 151)

David J Lucia, Philip S Beran, and Walter A Silva. Reduced-order modeling: new approaches for computational physics. *Progress in aerospace sciences*, 40(1-2): 51–117, 2004. (cited on Page 34)

Bethany Lusch, J Nathan Kutz, and Steven L Brunton. Deep learning for universal linear embeddings of nonlinear dynamics. *Nature communications*, 9(1):1–10, 2018. (cited on Page 52)

Juan Ma, Jie Zhang, Liangjie Li, Peter Wriggers, and Shahab Sahraee. Random homogenization analysis for heterogeneous materials with full randomness and correlation in microstructure based on finite element method and monte-carlo method. *Computational Mechanics*, 54(6):1395–1414, 2014. (cited on Page 92)

Juan Ma, Shahab Sahraee, Peter Wriggers, and Laura De Lorenzis. Stochastic multi-scale homogenization analysis of heterogeneous materials under finite deformations with full uncertainty in the microstructure. *Computational Mechanics*, 55(5): 819–835, 2015. (cited on Page 92)

Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(Nov):2579–2605, 2008. (cited on Page 60 and 82)

David JC MacKay. A practical bayesian framework for backpropagation networks. *Neural computation*, 4(3):448–472, 1992. (cited on Page 36)

David JC MacKay et al. Introduction to gaussian processes. *NATO ASI series F computer and systems sciences*, 168:133–166, 1998. (cited on Page 37)

Dougal Maclaurin, David Duvenaud, and Ryan Adams. Gradient-based hyperparameter optimization through reversible learning. In *International Conference on Machine Learning*, pages 2113–2122, 2015. (cited on Page 227)

Jim Magiera, Deep Ray, Jan S Hesthaven, and Christian Rohde. Constraint-aware neural networks for riemann problems. *Journal of Computational Physics*, 409: 109345, 2020. (cited on Page 46)

Martin Magill, Faisal Qureshi, and Hendrick W de Haan. Neural networks trained to solve differential equations learn general representations. *arXiv preprint arXiv:1807.00042*, 2018. (cited on Page 49)

Gianluca Maguolo and Loris Nanni. A critic evaluation of methods for covid-19 automatic detection from x-ray images. *Information Fusion*, 76:1–7, 2021. (cited on Page 12)

Nam Mai-Duy. Solving high order ordinary differential equations with radial basis function networks. *International Journal for Numerical Methods in Engineering*, 62(6):824–852, 2005. (cited on Page 43)

Nam Mai-Duy and Thanh Tran-Cong. Numerical solution of differential equations using multiquadric radial basis function networks. *Neural networks*, 14(2):185–199, 2001. (cited on Page 43)

Nam Mai-Duy and Thanh Tran-Cong. Mesh-free radial basis function network methods with domain decomposition for approximation of functions and numerical solution of poisson's equations. *Engineering Analysis with Boundary Elements*, 26 (2):133–156, 2002. (cited on Page 43)

Nam Mai-Duy and Thanh Tran-Cong. A cartesian-grid collocation method based on radial-basis-function networks for solving pdes in irregular domains. *Numerical Methods for Partial Differential Equations: An International Journal*, 23(5):1192–1210, 2007. (cited on Page 43)

Alaeddin Malek and R Shekari Beidokhti. Numerical solution for high order differential equations using a hybrid neural network—optimization method. *Applied Mathematics and Computation*, 183(1):260–271, 2006. (cited on Page 47)

Timothy Maly and Linda R Petzold. Numerical methods and software for sensitivity analysis of differential-algebraic systems. *Applied Numerical Mathematics*, 20(1-2): 57–79, 1996. (cited on Page 60)

Ankita Mangal and Elizabeth A Holm. Applied machine learning to predict stress hotspots i: Face centered cubic materials. *International Journal of Plasticity*, 111: 122–134, 2018. (cited on Page 36)

Ankita Mangal and Elizabeth A Holm. Applied machine learning to predict stress hotspots ii: Hexagonal close packed materials. *International Journal of Plasticity*, 114:1–14, 2019. (cited on Page 36)

Niall M Mangan, Steven L Brunton, Joshua L Proctor, and J Nathan Kutz. Inferring biological networks by sparse identification of nonlinear dynamics. *IEEE Transactions on Molecular, Biological and Multi-Scale Communications*, 2(1):52–63, 2016. (cited on Page 52)

Niall M Mangan, J Nathan Kutz, Steven L Brunton, and Joshua L Proctor. Model selection for dynamical systems via sparse regression and information criteria. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 473(2204):20170009, 2017. (cited on Page 52)

Niall M Mangan, Travis Askham, Steven L Brunton, J Nathan Kutz, and Joshua L Proctor. Model selection for hybrid dynamical systems via sparse regression. *Proceedings of the Royal Society A*, 475(2223):20180534, 2019. (cited on Page 52)

Ivana Manini, Federica Caponnetto, Anna Bartolini, Tamara Ius, Laura Mariuzzi, Carla Di Loreto, Antonio Paolo Beltrami, and Daniela Cesselli. Role of microenvironment in glioma invasion: what we learned from in vitro models. *International journal of molecular sciences*, 19(1):147, 2018. (cited on Page 285)

James Manyika, Michael Chui, Brad Brown, Jacques Bughin, Richard Dobbs, Charles Roxburgh, and Angela Hung. *Big data: The next frontier for innovation, competition, and productivity*. McKinsey Global Institute Reports, 2011. (cited on Page 24)

Andrea Manzoni, Alfio Quarteroni, and Gianluigi Rozza. Computational reduction for parametrized pdes: strategies and applications. *Milan Journal of Mathematics*, 80(2):283–309, 2012. (cited on Page 34)

Zhiping Mao, Ameya D Jagtap, and George Em Karniadakis. Physics-informed neural networks for high-speed flows. *Computer Methods in Applied Mechanics and Engineering*, 360:112789, 2020. (cited on Page 49 and 146)

Stephen Marsland. *Machine learning: an algorithmic perspective*. Chapman and Hall/CRC, 2011. (cited on Page 12)

Alicia Martínez-González, Gabriel F. Calvo, Luis A. Pérez Romasanta, and Víctor M. Pérez-García. Hypoxic Cell Waves Around Necrotic Cores in Glioblastoma: A Biomathematical Model and Its Therapeutic Implications. *Bulletin of Mathematical Biology*, 74(12):2875–2896, 2012. ISSN 00928240. doi: 10.1007/s11538-012-9786-1. (cited on Page 250, 256, 258, 259, 260, 261, 262, 287, 354, and 370)

Alicia Martínez-González, Mario Durán-Prado, Gabriel F. Calvo, Francisco J. Alcaín, Luis A. Pérez-Romasanta, and Víctor M. Pérez-García. Combined therapies of antithrombotics and antioxidants delay in silico brain tumour progression. *Mathematical Medicine and Biology*, 32(3):239–262, 2015. ISSN 14778602. doi: 10.1093/imammb/dqu002. (cited on Page 258, 259, 260, 261, 262, and 287)

Filippo Masi, Ioannis Stefanou, Paolo Vannucci, and Victor Maffi-Berthier. Thermodynamics-based artificial neural networks for constitutive modeling. *Journal of the Mechanics and Physics of Solids*, 147:104277, 2021. (cited on Page 41)

José M Matías, Manuel Febrero-Bande, Wenceslao González-Manteiga, and Juan Carlos Reboredo. Boosting garch and neural networks for the prediction of heteroskedastic time series. *Mathematical and Computer Modelling*, 51(3-4):256–271, 2010. (cited on Page 152)

Karel Matouš, Marc GD Geers, Varvara G Kouznetsova, and Andrew Gillman. A review of predictive nonlinear theories for multiscale modeling of heterogeneous materials. *Journal of Computational Physics*, 330:192–220, 2017. (cited on Page 126)

Alexandre Mauroy, Igor Mezić, and Yoshihiko Susuki. *The Koopman Operator in Systems and Control: Concepts, Methodologies, and Applications*, volume 484. Springer Nature, 2020. (cited on Page 52)

Micheal L Mavrovouniotis and S Chang. Hierarchical neural networks. *Computers & chemical engineering*, 16(4):347–369, 1992. (cited on Page 40)

Michael McAleer, Marcelo C Medeiros, and Daniel Slottje. A neural network demand system with heteroskedastic errors. *Journal of Econometrics*, 147(2):359–371, 2008. (cited on Page 152)

Michael T McCann, Kyong Hwan Jin, and Michael Unser. Convolutional neural networks for inverse problems in imaging: A review. *IEEE Signal Processing Magazine*, 34(6):85–95, 2017. (cited on Page 147)

Kevin Stanley McFall and James Robert Mahan. Artificial neural network method for solution of boundary value problems with exact satisfaction of arbitrary boundary conditions. *IEEE Transactions on Neural Networks*, 20(8):1221–1233, 2009. (cited on Page 47)

AJ McNeil and R Frey. P. embrechts (2005). quantitative risk management: Concepts, techniques, and tools, 2017. (cited on Page 235)

Andrew J Meade Jr and Alvaro A Fernandez. The numerical solution of linear ordinary differential equations by feedforward neural networks. *Mathematical and Computer Modelling*, 19(12):1–25, 1994a. (cited on Page 47)

Andrew J Meade Jr and Alvaro A Fernandez. Solution of nonlinear ordinary differential equations by feedforward neural networks. *Mathematical and Computer Modelling*, 20(9):19–44, 1994b. (cited on Page 47)

Patrick Mehlen and Alain Puisieux. Metastasis: a question of life or death. *Nature reviews cancer*, 6(6):449–458, 2006. (cited on Page 4)

Alton Meister. *Advances in Enzymology and Related Areas of Molecular Biology, Volume 17*. Advances in Enzymology - and Related Areas of Molecular Biology. by Interscience Publishers, Inc., 1956. ISBN 9780470647462,9780470122624. URL http://gen.lib.rus.ec/book/index.php?md5=8c34c4973d8bb6feed125599b81d6bb7. (cited on Page 255)

Attilio Meucci. A new breed of copulas for risk and portfolio management. *Risk*, 24 (9):122–126, 2011. (cited on Page 235)

Christian Miehe and CG Bayreuther. On multiscale fe analyses of heterogeneous structures: from homogenization to multigrid solvers. *International Journal for Numerical Methods in Engineering*, 71(10):1135–1180, 2007. (cited on Page 126)

Christian Miehe, Jan Schotte, and Jörg Schröder. Computational micro–macro transitions and overall moduli in the analysis of polycrystals at large strains. *Computational Materials Science*, 16(1):372–382, 1999. (cited on Page 92)

Marc P Mignolet, Adam Przekop, Stephen A Rizzi, and S Michael Spottswood. A review of indirect/non-intrusive reduced order modeling of nonlinear geometric structures. *Journal of Sound and Vibration*, 332(10):2437–2460, 2013. (cited on Page 34)

Pedro M Milani, Julia Ling, and John K Eaton. On the generality of tensor basis neural networks for turbulent scalar flux modeling. *International Communications in Heat and Mass Transfer*, 128:105626, 2021. (cited on Page 41)

Michele Milano and Petros Koumoutsakos. Neural network modeling for near wall turbulent flow. *Journal of Computational Physics*, 182(1):1–26, 2002. (cited on Page 36)

Mar Miñano and Francisco J Montáns. Wypiwyg damage mechanics for soft materials: A data-driven approach. *Archives of Computational Methods in Engineering*, 25 (1):165–193, 2018. (cited on Page 59)

George S Misyris, Andreas Venzke, and Spyros Chatzivasileiadis. Physics-informed neural networks for power systems. In *2020 IEEE Power & Energy Society General Meeting (PESGM)*, pages 1–5. IEEE, 2020. (cited on Page 49)

Melanie Mitchell. *An introduction to genetic algorithms*. MIT press, 1998. (cited on Page 33)

Tom M Mitchell. Machine learning. 1997. *Burr Ridge, IL: McGraw Hill*, 45(37): 870–877, 1997. (cited on Page 24)

Shaoxing Mo, Yinhao Zhu, Nicholas Zabaras, Xiaoqing Shi, and Jichun Wu. Deep convolutional encoder-decoder networks for uncertainty quantification of dynamic multiphase flow in heterogeneous media. *Water Resources Research*, 55(1):703–728, 2019. (cited on Page 36)

Saeed Mohammadzadeh and Emma Lejeune. Predicting mechanically driven full-field quantities of interest with deep learning-based metamodels. *Extreme Mechanics Letters*, page 101566, 2021. (cited on Page 36)

Subhasish Mohanty, Rikki Teale, Aditi Chattopadhyay, Pedro Peralta, and Christina Willhauck. Mixed gaussian process and state-space approach for fatigue crack growth prediction. In *International workshop on structural heath monitoring*, volume 2, pages 1108–1115, 2007. (cited on Page 70)

Ana Rita Monteiro, Richard Hill, Geoffrey J Pilkington, and Patrícia A Madureira. The role of hypoxia in glioblastoma invasion. *Cells*, 6(4):45, 2017. (cited on Page 286 and 299)

Francisco Montero-Chacón, José A Sanz-Herrera, and Manuel Doblaré. Computational multiscale solvers for continuum approaches. *Materials*, 12(5):691, 2019. (cited on Page 126)

J Mora-Macías, A Pajares, P Miranda, J Domínguez, and E Reina-Romo. Mechanical characterization via nanoindentation of the woven bone developed during bone transport. *Journal of the mechanical behavior of biomedical materials*, 74:236–244, 2017. (cited on Page 127)

Juan Mora-Macías, J Ayensa-Jiménez, Esther Reina-Romo, Mohamed H Doweidar, J Domínguez, M Doblaré, and José Antonio Sanz-Herrera. A multiscale data-driven

approach for bone tissue biomechanics. *Computer Methods in Applied Mechanics and Engineering*, 368:113136, 2020. (cited on Page 17, 20, 120, and 133)

Philippos Mordohai and Gérard Medioni. Dimensionality estimation, manifold learning and function approximation using tensor voting. *Journal of Machine Learning Research*, 11(1), 2010. (cited on Page 58)

Bobak Mosadegh, Wajeeh Saadi, Shur-Jen Wang, and Noo Li Jeon. Epidermal growth factor promotes breast cancer cell chemotaxis in cxcl12 gradients. *Biotechnology and bioengineering*, 100(6):1205–1213, 2008. (cited on Page 9)

B Moya, D Gonzalez, I Alfaro, F Chinesta, and E Cueto. Learning slosh dynamics by means of data. *Computational Mechanics*, pages 1–13, 2019. (cited on Page 43 and 164)

Beatriz Moya, Iciar Alfaro, David Gonzalez, Francisco Chinesta, and Elías Cueto. Physically sound, self-learning digital twins for sloshing fluids. *PLoS One*, 15(6): e0234569, 2020a. (cited on Page 43, 57, 356, and 372)

Beatriz Moya, Alberto Badías, Icíar Alfaro, Francisco Chinesta, and Elías Cueto. Digital twins that learn and correct themselves. *International Journal for Numerical Methods in Engineering*, 2020b. (cited on Page 56)

M Mozaffar, R Bostanabad, W Chen, K Ehmann, Jian Cao, and MA Bessa. Deep learning predicts path-dependent plasticity. *Proceedings of the National Academy of Sciences*, 116(52):26414–26420, 2019. (cited on Page 36)

Joakim Munkhammar and Joakim Widén. An autocorrelation-based copula model for generating realistic clear-sky index time-series. *Solar Energy*, 158:9–19, 2017. (cited on Page 235)

Nikhil Muralidhar, Mohammad Raihanul Islam, Manish Marwah, Anuj Karpatne, and Naren Ramakrishnan. Incorporating prior domain knowledge into deep neural networks. In *2018 IEEE international conference on big data (big data)*, pages 36–45. IEEE, 2018. (cited on Page 46)

Ahmed Musah-Eroje and Sue Watson. Adaptive changes of glioblastoma cells following exposure to hypoxic (1% oxygen) tumour microenvironment. *International journal of molecular sciences*, 20(9):2091, 2019. (cited on Page 289)

Mohammad Amin Nabian and Hadi Meidani. A deep neural network surrogate for high-dimensional random partial differential equations. *arXiv preprint arXiv:1806.02957*, 2018. (cited on Page 49)

Mohammad Amin Nabian and Hadi Meidani. Physics-driven regularization of deep neural networks for enhanced engineering design and analysis. *Journal of Computing and Information Science in Engineering*, 20(1):011006, 2020. (cited on Page 49)

Frank Naets, Daniel De Gregoriis, and Wim Desmet. Multi-expansion modal reduction: A pragmatic semi–a priori model order reduction approach for nonlinear

structural dynamics. *International Journal for Numerical Methods in Engineering*, 118(13):765–782, 2019. (cited on Page 40)

Anika Nagelkerke, Johan Bussink, Alan E Rowan, and Paul N Span. The mechanical microenvironment in cancer: How physics affects tumours. In *Seminars in cancer biology*, volume 35, pages 62–70. Elsevier, 2015. (cited on Page 4 and 235)

Michikazu Nakai and Weiming Ke. Review of the methods for handling missing data in longitudinal data analysis. *International Journal of Mathematical Analysis*, 5 (1):1–13, 2011. (cited on Page 102 and 122)

Jwa-Min Nam, C Shad Thaxton, and Chad A Mirkin. Nanoparticle-based bio-bar codes for the ultrasensitive detection of proteins. *science*, 301(5641):1884–1886, 2003. (cited on Page 9)

Sritama Nath and Gayathri R Devi. Three-dimensional culture systems in cancer research: Focus on tumor spheroid model. *Pharmacology & therapeutics*, 163: 94–108, 2016. (cited on Page 8)

B Nayroles, G Touzot, and P38504470764 Villon. Generalizing the finite element method: diffuse approximation and diffuse elements. *Computational mechanics*, 10(5):307–318, 1992. (cited on Page 173)

Radford M Neal. *Bayesian learning for neural networks*, volume 118. Springer Science & Business Media, 2012. (cited on Page 37)

Federico Negri, Andrea Manzoni, and David Amsallem. Efficient model reduction of parametrized systems by matrix discrete empirical interpolation. *Journal of Computational Physics*, 303:431–454, 2015. (cited on Page 34)

David Néron and Pierre Ladevèze. Proper generalized decomposition for multiscale and multiphysics problems. *Archives of Computational Methods in Engineering*, 17(4):351–372, 2010. (cited on Page 40)

Duong Nguyen, Said Ouala, Lucas Drumetz, and Ronan Fablet. Em-like learning chaotic dynamics from noisy and partial observations. *arXiv preprint arXiv:1903.10335*, 2019. (cited on Page 52)

Lu Trong Khiem Nguyen and Marc-André Keip. A data-driven approach to nonlinear elasticity. *Computers & Structures*, 194:97–115, 2018. (cited on Page 58)

Lu Trong Khiem Nguyen, Matthias Rambausek, and Marc-André Keip. Variational framework for distance-minimizing method in data-driven computational mechanics. *Computer Methods in Applied Mechanics and Engineering*, 365:112898, 2020. (cited on Page 58)

Ngoc C Nguyen and Jaime Peraire. An efficient reduced-order modeling approach for non-linear parametrized partial differential equations. *International Journal for Numerical Methods in Engineering*, 76(1):27–55, 2008. (cited on Page 34)

TK Nguyen, Olivier Carpentier, Francine Monchau, Feng Chai, Jean-Christophe Hornez, and Ph Hivart. Numerical optimization of cell colonization modelling inside scaffold for perfusion bioreactor: A multiscale model. *Medical engineering & physics*, 57:40–50, 2018. (cited on Page 126)

Vinh Phu Nguyen, Martijn Stroeven, and Lambertus Johannes Sluys. Multiscale continuous and discontinuous modeling of heterogeneous materials: a review on recent developments. *Journal of Multiscale Modelling*, 3(04):229–270, 2011. (cited on Page 126)

Vien Minh Nguyen-Thanh, Xiaoying Zhuang, and Timon Rabczuk. A deep energy method for finite deformation hyperelasticity. *European Journal of Mechanics-A/Solids*, 80:103874, 2020. (cited on Page 48)

Michael A Nielsen. *Neural networks and deep learning*, volume 2018. Determination press San Francisco, CA, USA:, 2015. (cited on Page 32 and 182)

Stefanos Nikolopoulos, Ioannis Kalogeris, and Vissarion Papadopoulos. Non-intrusive surrogate modeling for parametrized time-dependent pdes using convolutional autoencoders. *arXiv preprint arXiv:2101.05555*, 2021. (cited on Page 36)

Siamak Niroomandi, Icíar Alfaro, David González, Elías Cueto, and Francisco Chinesta. Model order reduction in hyperelasticity: a proper generalized decomposition approach. *International Journal for Numerical Methods in Engineering*, 96 (3):129–149, 2013. (cited on Page 40)

Bernd R Noack, Konstantin Afanasiev, MAREK MORZYŃSKI, Gilead Tadmor, and Frank Thiele. A hierarchy of low-dimensional models for the transient and post-transient cylinder wake. *Journal of Fluid Mechanics*, 497:335–363, 2003. (cited on Page 49)

Oameed Noakoasteen, Shu Wang, Zhen Peng, and Christos Christodoulou. Physics-informed deep neural networks for transient electromagnetic analysis. *IEEE Open Journal of Antennas and Propagation*, 1:404–412, 2020. (cited on Page 49)

Garry P Nolan. What's wrong with drug screening today. *Nature chemical biology*, 3 (4):187–191, 2007. (cited on Page 8)

Takahiro Oike, Yoshiyuki Suzuki, Ken-ichi Sugawara, Katsuyuki Shirai, Shin-ei Noda, Tomoaki Tamaki, Masaya Nagaishi, Hideaki Yokoo, Yoichi Nakazato, and Takashi Nakano. Radiotherapy plus concomitant adjuvant temozolomide for glioblastoma: Japanese mono-institutional results. *PloS one*, 8(11):e78943, 2013. (cited on Page 7 and 234)

Atsuya Oishi and Genki Yagawa. Computational mechanics enhanced by deep learning. *Computer Methods in Applied Mechanics and Engineering*, 327:327–351, 2017. (cited on Page 34)

Peggy L Olive, Charlene Vikse, and Martin J Trotter. Measurement of oxygen diffusion distance in tumor cubes using a fluorescent hypoxia probe. *International Journal of Radiation Oncology* Biology* Physics*, 22(3):397–402, 1992. (cited on Page 261)

Ana Isabel Oliveira, Sandra I Anjo, Joana Vieira de Castro, Sofia C Serra, António J Salgado, Bruno Manadas, and Bruno M Costa. Crosstalk between glial and glioblastoma cells triggers the "go-or-grow" phenotype of tumor cells. *Cell Communication and Signaling*, 15(1):1–12, 2017. (cited on Page 316)

Frank Olver. *Asymptotics and special functions*. CRC Press, 1997. (cited on Page 336)

A Wayne Orr, Brian P Helmke, Brett R Blackman, and Martin A Schwartz. Mechanisms of mechanotransduction. *Developmental cell*, 10(1):11–20, 2006. (cited on Page 4)

Robert Osserman. *A survey of minimal surfaces*. Courier Corporation, 2013. (cited on Page 203)

Quinn T Ostrom, Haley Gittleman, Paul Farah, Annie Ondracek, Yanwen Chen, Yingli Wolinsky, Nancy E Stroup, Carol Kruchko, and Jill S Barnholtz-Sloan. Cbtrus statistical report: Primary brain and central nervous system tumors diagnosed in the united states in 2006-2010. *Neuro-oncology*, 15(suppl_2):ii1–ii56, 2013. (cited on Page 7)

Hans Christian Öttinger. *Beyond equilibrium thermodynamics*. John Wiley & Sons, 2005. (cited on Page 41)

Hans Christian Öttinger and Miroslav Grmela. Dynamics and thermodynamics of complex fluids. ii. illustrations of a general formalism. *Physical Review E*, 56(6): 6633, 1997. (cited on Page 164)

Dennis Overbye. Can a computer devise a theory of everything? https://www.nytimes.com/2020/11/23/science/artificial-intelligence-ai-physics-theory.html, November 2020. Accessed: 2021-11-21. (cited on Page 12)

Houman Owhadi. Bayesian numerical homogenization. *Multiscale Modeling & Simulation*, 13(3):812–828, 2015. (cited on Page 37)

Amy L Paguirigan and David J Beebe. Microfluidics meet cell biology: bridging the gap by validation and application of microscale techniques for cell biological assays. *BioEssays*, 30(9):811–821, 2008. (cited on Page 285)

Guofei Pang, Lu Lu, and George Em Karniadakis. fpinns: Fractional physics-informed neural networks. *SIAM Journal on Scientific Computing*, 41(4):A2603–A2626, 2019. (cited on Page 49)

Nicolas Papernot, Patrick McDaniel, Ian Goodfellow, Somesh Jha, Z Berkay Celik, and Ananthram Swami. Practical black-box attacks against machine learning. In *Proceedings of the 2017 ACM on Asia conference on computer and communications security*, pages 506–519, 2017. (cited on Page 155)

Athanasios Papoulis. *Probability & statistics*, volume 2. Prentice-Hall Englewood Cliffs, 1990. (cited on Page 392)

Etienne Pardoux and Shige Peng. Adapted solution of a backward stochastic differential equation. *Systems & Control Letters*, 14(1):55–61, 1990. (cited on Page 44)

Eric J Parish and Karthik Duraisamy. A paradigm for data-driven predictive modeling using field inversion and machine learning. *Journal of Computational Physics*, 305: 758–774, 2016. (cited on Page 26)

HM Park and DH Cho. The use of the karhunen-loeve decomposition for the modeling of distributed parameter systems. *Chemical Engineering Science*, 51(1):81–98, 1996. (cited on Page 34)

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. In *Advances in neural information processing systems*, pages 8026–8037, 2019. (cited on Page 145)

Aalpen A Patel, Edward T Gawlinski, Susan K Lemieux, and Robert A Gatenby. A cellular automaton model of early tumor growth and invasion: the effects of native tissue vascularity and increased anaerobic tumor metabolism. *Journal of Theoretical Biology*, 213(3):315–331, 2001. (cited on Page 256)

KK Pathak, Sanjay Panthi, and N Ramakrishnan. Application of neural network in sheet metal bending process. *Defence Science Journal*, 55(2):125, 2005. (cited on Page 158)

Benjamin Peherstorfer and Karen Willcox. Dynamic data-driven reduced-order models. *Computer Methods in Applied Mechanics and Engineering*, 291:21–41, 2015. (cited on Page 55)

Benjamin Peherstorfer and Karen Willcox. Data-driven operator inference for nonintrusive projection-based model reduction. *Computer Methods in Applied Mechanics and Engineering*, 306:196–215, 2016. (cited on Page 51 and 56)

Chao-Ying Joanne Peng, Michael Harwell, Show-Mann Liou, Lee H Ehman, et al. Advances in missing data methods and implications for educational research. *Real data analysis*, 3178, 2006. (cited on Page 102)

Marina Pérez-Aliacar, Mohamed H Doweidar, Manuel Doblaré, and Jacobo Ayensa-Jiménez. Predicting cell behaviour parameters from glioblastoma on a chip images. a deep learning approach. *Computers in Biology and Medicine*, page 104547, 2021. (cited on Page 37, 293, 299, and 322)

Víctor M Pérez-García, Gabriel F Calvo, Juan Belmonte-Beitia, David Diego, and Luis Pérez-Romasanta. Bright solitary waves in malignant gliomas. *Physical Review E*, 84(2):021921, 2011. (cited on Page 322)

Micha Pfeiffer, Carina Riediger, Jürgen Weitz, and Stefanie Speidel. Learning soft tissue behavior of organs for surgical navigation with convolutional neural networks. *International journal of computer assisted radiology and surgery*, 14(7):1147–1155, 2019. (cited on Page 36)

Hoang Pham. Recent studies in software reliability engineering. In *Handbook of Reliability Engineering*, pages 285–302. Springer, 2003. (cited on Page 235)

Renzo Phellan, Bahe Hachem, Julien Clin, Jean-Marc Mac-Thiong, and Luc Duong. Real-time biomechanics using the finite element method and machine learning: Review and perspective. *Medical Physics*, 48(1):7–18, 2021. (cited on Page 36)

Allan Pinkus. Approximation theory of the mlp model in neural networks. *Acta numerica*, 8:143–195, 1999. (cited on Page 47 and 151)

Auriane Platzer, Adrien Leygue, Laurent Stainier, and Michael Ortiz. Finite element solver for data-driven finite strain elasticity. *Computer Methods in Applied Mechanics and Engineering*, 379:113756, 2021. (cited on Page 58)

Tomaso Poggio, Hrushikesh Mhaskar, Lorenzo Rosasco, Brando Miranda, and Qianli Liao. Why and when can deep-but not shallow-networks avoid the curse of dimensionality: a review. *International Journal of Automation and Computing*, 14 (5):503–519, 2017. (cited on Page 229)

Brian W Pogue, Julia A O'Hara, Carmen M Wilmot, Keith D Paulsen, and Harold M Swartz. Estimation of oxygen distribution in rif-1 tumors by diffusion model-based interpretation of pimonidazole hypoxia and eppendorf measurements. *Radiation Research*, 155(1):15–25, 2001. (cited on Page 255 and 261)

Tony Pourmohamad. Surrogates: Gaussian process modeling, design, and optimization for the applied sciences, 2021. (cited on Page 37)

Joshua L Proctor, Steven L Brunton, and J Nathan Kutz. Dynamic mode decomposition with control. *SIAM Journal on Applied Dynamical Systems*, 15(1):142–161, 2016. (cited on Page 56)

Dimitris C Psichogios and Lyle H Ungar. A hybrid neural network-first principles approach to process modeling. *AIChE Journal*, 38(10):1499–1511, 1992. (cited on Page 40)

P. Pujari, M. Sewak, and R. Karim. *Practical Convolutional Neural Network Models*. Packt Publishing, 2018. ISBN ISBN:1788392302. (cited on Page 147)

Tong Qin, Kailiang Wu, and Dongbin Xiu. Data driven governing equations approximation using deep neural networks. *Journal of Computational Physics*, 395: 620–635, 2019. (cited on Page 44)

Markus Quade, Markus Abel, Kamran Shafi, Robert K Niven, and Bernd R Noack. Prediction of dynamical systems by symbolic regression. *Physical Review E*, 94(1): 012214, 2016. (cited on Page 51)

Daniela F Quail and Johanna A Joyce. Microenvironmental regulation of tumor progression and metastasis. *Nature medicine*, 19(11):1423–1437, 2013. (cited on Page 4)

Wullianallur Raghupathi and Viju Raghupathi. Big data analytics in healthcare: promise and potential. *Health Information Science and Systems*, 2(1):1, 2014. (cited on Page 24)

Maziar Raissi. Deep hidden physics models: Deep learning of nonlinear partial differential equations. *The Journal of Machine Learning Research*, 19(1):932–955, 2018. (cited on Page 52 and 146)

Maziar Raissi and George Em Karniadakis. Hidden physics models: Machine learning of nonlinear partial differential equations. *Journal of Computational Physics*, 357: 125–141, 2018. (cited on Page 39 and 49)

Maziar Raissi, Paris Perdikaris, and George Em Karniadakis. Inferring solutions of differential equations using noisy multi-fidelity data. *Journal of Computational Physics*, 335:736–746, 2017a. (cited on Page 37 and 39)

Maziar Raissi, Paris Perdikaris, and George Em Karniadakis. Machine learning of linear differential equations using gaussian processes. *Journal of Computational Physics*, 348:683–693, 2017b. (cited on Page 37 and 39)

Maziar Raissi, Paris Perdikaris, and George Em Karniadakis. Multistep neural networks for data-driven discovery of nonlinear dynamical systems. *arXiv preprint arXiv:1801.01236*, 2018a. (cited on Page 54)

Maziar Raissi, Paris Perdikaris, and George Em Karniadakis. Numerical gaussian processes for time-dependent and nonlinear partial differential equations. *SIAM Journal on Scientific Computing*, 40(1):A172–A198, 2018b. (cited on Page 39)

Maziar Raissi, Paris Perdikaris, and George E Karniadakis. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics*, 378:686–707, 2019. (cited on Page 48, 146, 147, 152, 158, 161, 163, 173, 294, and 299)

Maziar Raissi, Alireza Yazdani, and George Em Karniadakis. Hidden fluid mechanics: Learning velocity and pressure fields from flow visualizations. *Science*, 367(6481): 1026–1030, 2020. (cited on Page 146)

Krishna Rajan. Materials informatics. *Materials Today*, 8(10):38–45, 2005. (cited on Page 57 and 70)

Krishna Rajan. Materials informatics: The materials "gene" and big data. *Annual Review of Materials Research*, 45:153–169, 2015. (cited on Page 57)

Saroja Ramanujan, Alain Pluen, Trevor D McKee, Edward B Brown, Yves Boucher, and Rakesh K Jain. Diffusion and convection in collagen gels: implications for transport in the tumor interstitium. *Biophysical journal*, 83(3):1650–1660, 2002. (cited on Page 9)

Yulian P Ramirez, Jessica L Weatherbee, Richard T Wheelhouse, and Alonzo H Ross. Glioblastoma multiforme therapy and mechanisms of resistance. *Pharmaceuticals*, 6(12):1475–1506, 2013. (cited on Page 285)

Muruhan Rathinam and Linda R Petzold. A new look at proper orthogonal decomposition. *SIAM Journal on Numerical Analysis*, 41(5):1893–1925, 2003. (cited on Page 34)

Waseem Rawat and Zenghui Wang. Deep convolutional neural networks for image classification: A comprehensive review. *Neural computation*, 29(9):2352–2449, 2017. (cited on Page 147)

Muhammad Imran Razzak, Saeeda Naz, and Ahmad Zaib. Deep learning for medical image processing: Overview, challenges and the future. *Classification in BioApps*, pages 323–350, 2018. (cited on Page 36)

E Reina-Romo and JA Sanz-Herrera. Multiscale simulation of particle-reinforced elastic–plastic adhesives at small strains. *Computer methods in applied mechanics and engineering*, 200(25-28):2211–2222, 2011. (cited on Page 126)

Katarzyna A Rejniak. *Systems Biology of Tumor Microenvironment: Quantitative Modeling and Simulations*, volume 936. Springer, 2016. (cited on Page 10 and 234)

Brandon Reyes, Amanda A Howard, Paris Perdikaris, and Alexandre M Tartakovsky. Learning unknown physics of non-newtonian fluids. *Physical Review Fluids*, 6(7): 073301, 2021. (cited on Page 49)

Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. " why should i trust you?" explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1135–1144, 2016. (cited on Page 12)

FJ Richards. A flexible growth function for empirical use. *Journal of experimental Botany*, 10(2):290–301, 1959. (cited on Page 239)

Stephen Riffle and Rashmi S Hegde. Modeling tumor cell adaptations to hypoxia in multicellular tumor spheroids. *Journal of Experimental & Clinical Cancer Research*, 36(1):102, 2017. (cited on Page 285)

Jason Riordon, Dušan Sovilj, Scott Sanner, David Sinton, and Edmond WK Young. Deep learning with microfluidics for biotechnology. *Trends in biotechnology*, 37(3): 310–324, 2019. (cited on Page 11)

Mercedes A Rivero, Robert T Tranquillo, Helen M Buettner, and Douglas A Lauffenburger. Transport models for chemotactic cell populations based on individual cell behavior. *Chemical Engineering Science*, 44(12):2881–2897, 1989. (cited on Page 259)

IBCM Rocha, Pierre Kerfriden, and FP van der Meer. On-the-fly construction of surrogate constitutive models for concurrent multiscale mechanical analysis through probabilistic machine learning. *Journal of Computational Physics: X*, 9: 100083, 2021. (cited on Page 37)

J Leonel Rocha and Sandra M Aleixo. An extension of gompertzian growth dynamics: Weibull and fréchet models. *Mathematical Biosciences & Engineering*, 10(2):379, 2013. (cited on Page 239)

Rockhill Rockne, JK Rockhill, M Mrugala, AM Spence, I Kalet, K Hendrickson, A Lai, T Cloughesy, EC Alvord Jr, and KR Swanson. Predicting the efficacy of radiotherapy in individual glioblastoma patients in vivo: a mathematical modeling approach. *Physics in Medicine & Biology*, 55(12):3271, 2010. (cited on Page 258, 259, 260, and 262)

Lior Rokach and Oded Z Maimon. *Data mining with decision trees: theory and applications*, volume 69. World scientific, 2007. (cited on Page 32)

David Rolnick, Andreas Veit, Serge Belongie, and Nir Shavit. Deep learning is robust to massive label noise. *arXiv preprint arXiv:1705.10694*, 2017. (cited on Page 194)

Ignacio Romero. Thermodynamically consistent time-stepping algorithms for non-linear thermomechanical systems. *International journal for numerical methods in engineering*, 79(6):706–732, 2009. (cited on Page 43)

Ignacio Romero. Algorithms for coupled problems that preserve symmetries and the laws of thermodynamics: Part i: Monolithic integrators and their application to finite strain thermoelasticity. *Computer Methods in Applied Mechanics and Engineering*, 199(25-28):1841–1858, 2010a. (cited on Page 43)

Ignacio Romero. Algorithms for coupled problems that preserve symmetries and the laws of thermodynamics: Part ii: Fractional step methods. *Computer Methods in Applied Mechanics and Engineering*, 199(33-36):2235–2248, 2010b. (cited on Page 43)

Xabier Romero, Marcos Latorre, and Francisco J Montáns. Determination of the wypiwyg strain energy density of skin through finite element analysis of the experiments on circular specimens. *Finite Elements in Analysis and Design*, 134: 1–15, 2017. (cited on Page 59)

Yuan Rong, Donald L Durden, Erwin G Van Meir, and Daniel J Brat. 'pseudopalisading'necrosis in glioblastoma: a familiar morphologic feature that links vascular pathology, hypoxia, and angiogenesis. *Journal of Neuropathology & Experimental Neurology*, 65(6):529–539, 2006. (cited on Page 234 and 317)

Xavier Ros-Oton. Nonlocal elliptic equations in bounded domains: a survey. *Publicacions matematiques*, pages 3–26, 2016. (cited on Page 172)

Frank Rosenblatt. The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological review*, 65(6):386, 1958. (cited on Page 24 and 153)

Sam T Roweis and Lawrence K Saul. Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290(5500):2323–2326, 2000. (cited on Page 59 and 82)

Clarence W Rowley and Scott TM Dawson. Model reduction for flow analysis and control. *Annual Review of Fluid Mechanics*, 49:387–417, 2017. (cited on Page 34 and 56)

Gianluigi Rozza, Dinh Bao Phuong Huynh, and Anthony T Patera. Reduced basis approximation and a posteriori error estimation for affinely parametrized elliptic

coercive partial differential equations. *Archives of Computational Methods in Engineering*, 15(3):229–275, 2008. (cited on Page 34)

V. Ruas. *Numerical Methods for Partial Differential Equations: An Introduction*. John Wiley and Sons Ltd., Chichester, West Sussex, United Kingdom, 2016. ISBN ISBN: 978-1-119-11136-8. (cited on Page 147)

Donald B Rubin. Inference and missing data. *Biometrika*, 63(3):581–592, 1976. (cited on Page 102)

Donald B Rubin. Multiple imputation after 18+ years. *Journal of the American statistical Association*, 91(434):473–489, 1996. (cited on Page 102)

Keith Rudd and Silvia Ferrari. A constrained integration (cint) approach to solving partial differential equations using artificial neural networks. *Neurocomputing*, 155: 277–285, 2015. (cited on Page 48 and 173)

Keith Rudd, Gianluca Di Muro, and Silvia Ferrari. A constrained backpropagation approach for the adaptive solution of partial differential equations. *IEEE transactions on neural networks and learning systems*, 25(3):571–584, 2013. (cited on Page 48)

Samuel Rudy, Alessandro Alla, Steven L Brunton, and J Nathan Kutz. Data-driven identification of parametric partial differential equations. *SIAM Journal on Applied Dynamical Systems*, 18(2):643–660, 2019a. (cited on Page 52)

Samuel H Rudy, Steven L Brunton, Joshua L Proctor, and J Nathan Kutz. Data-driven discovery of partial differential equations. *Science Advances*, 3(4):e1602614, 2017. (cited on Page 52)

Samuel H Rudy, J Nathan Kutz, and Steven L Brunton. Deep learning of dynamics and signal-noise decomposition with time-stepping constraints. *Journal of Computational Physics*, 396:483–506, 2019b. (cited on Page 56)

D Ryckelynck, L Hermanns, F Chinesta, and E Alarcon. An efficient 'a priori'model reduction for boundary element models. *Engineering Analysis with Boundary Elements*, 29(8):796–801, 2005. (cited on Page 40)

David Ryckelynck, Francisco Chinesta, Elías Cueto, and Amine Ammar. On the" a priori" model reduction: Overview and recent developments. *Archives of Computational methods in Engineering*, 13(1):91–128, 2006. (cited on Page 40)

Eric K Sackmann, Anna L Fulton, and David J Beebe. The present and future role of microfluidics in biomedical research. *Nature*, 507(7491):181–189, 2014. (cited on Page 9)

Michael G Safonov and RY1000665 Chiang. A schur method for balanced-truncation model reduction. *IEEE Transactions on automatic control*, 34(7):729–733, 1989. (cited on Page 34)

Francisco Sahli Costabal, Yibo Yang, Paris Perdikaris, Daniel E Hurtado, and Ellen Kuhl. Physics-informed neural networks for cardiac activation mapping. *Frontiers in Physics*, 8:42, 2020. (cited on Page 49)

S Sakata, F Ashida, and T Kojima. Stochastic homogenization analysis for thermal expansion coefficients of fiber reinforced composites using the equivalent inclusion method with perturbation-based approach. *Computers & structures*, 88(7):458–466, 2010. (cited on Page 92)

AK Md Ehsanes Saleh, Mohammad Arashi, and BM Golam Kibria. *Theory of ridge regression estimation with applications*, volume 285. John Wiley & Sons, 2019. (cited on Page 45)

Esteban Samaniego, Cosmin Anitescu, Somdatta Goswami, Vien Minh Nguyen-Thanh, Hongwei Guo, Khader Hamdia, X Zhuang, and T Rabczuk. An energy approach to the solution of partial differential equations in computational mechanics via machine learning: Concepts, implementation and applications. *Computer Methods in Applied Mechanics and Engineering*, 362:112790, 2020. (cited on Page 48)

Wojciech Samek, Thomas Wiegand, and Klaus-Robert Müller. Explainable artificial intelligence: Understanding, visualizing and interpreting deep learning models. *arXiv preprint arXiv:1708.08296*, 2017. (cited on Page 155 and 228)

Wojciech Samek, Grégoire Montavon, Andrea Vedaldi, Lars Kai Hansen, and Klaus-Robert Müller. *Explainable AI: interpreting, explaining and visualizing deep learning*, volume 11700. Springer Nature, 2019. (cited on Page 12)

Abel Sancarlos, Morgan Cameron, Andreas Abel, Elias Cueto, Jean-Louis Duval, and Francisco Chinesta. From rom of electrochemistry to ai-based battery digital and hybrid twin. *Archives of Computational Methods in Engineering*, 28(3):979–1015, 2021. (cited on Page 57)

Alvaro Sanchez-Gonzalez, Nicolas Heess, Jost Tobias Springenberg, Josh Merel, Martin Riedmiller, Raia Hadsell, and Peter Battaglia. Graph networks as learnable physics engines for inference and control. In *International Conference on Machine Learning*, pages 4470–4479. PMLR, 2018. (cited on Page 37)

Thomas J Santner, Brian J Williams, William I Notz, and Brain J Williams. *The design and analysis of computer experiments*, volume 1. Springer, 2003. (cited on Page 37)

JA Sanz-Herrera, JM García-Aznar, and M Doblaré. Micro–macro numerical modelling of bone regeneration in tissue engineering. *Computer Methods in Applied Mechanics and Engineering*, 197(33-40):3092–3107, 2008. (cited on Page 126)

JA Sanz-Herrera, JM García-Aznar, and M Doblaré. On scaffold designing for bone regeneration: a computational multiscale approach. *Acta Biomaterialia*, 5(1): 219–229, 2009. (cited on Page 126)

José A Sanz-Herrera, Juan Mora-Macías, J Ayensa-Jiménez, E Reina-Romo, Mohamed H Doweidar, J Domínguez, and M Doblaré. Data-driven computational

simulation in bone mechanics. *Annals of Biomedical Engineering*, 49(1):407–419, 2021. (cited on Page 20 and 146)

Simo Särkkä. Linear operators and stochastic partial differential equations in gaussian process regression. In *International Conference on Artificial Neural Networks*, pages 151–158. Springer, 2011. (cited on Page 39)

Warren S Sarle. Neural networks and statistical models. In *Proceedings of the Nineteenth Annual SAS Users Group International Conference, April, 1994*. Citeseer, 1994. (cited on Page 152)

Jack W Scannell, Alex Blanckley, Helen Boldon, and Brian Warrington. Diagnosing the decline in pharmaceutical r&d efficiency. *Nature reviews Drug discovery*, 11(3): 191–200, 2012. (cited on Page 8 and 9)

Hayden Schaeffer. Learning partial differential equations via data discovery and sparse optimization. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 473(2197):20160446, 2017. (cited on Page 52)

Hayden Schaeffer and Scott G McCalla. Sparse model selection via integral terms. *Physical Review E*, 96(2):023302, 2017. (cited on Page 51 and 52)

Hayden Schaeffer, Russel Caflisch, Cory D Hauck, and Stanley Osher. Sparse dynamics for partial differential equations. *Proceedings of the National Academy of Sciences*, 110(17):6634–6639, 2013. (cited on Page 48)

Hayden Schaeffer, Giang Tran, and Rachel Ward. Extracting sparse high-dimensional dynamics from limited data. *SIAM Journal on Applied Mathematics*, 78(6):3279–3295, 2018. (cited on Page 52)

Joseph L Schafer. *Analysis of incomplete multivariate data*. CRC press, 1997. (cited on Page 102 and 123)

Joseph L Schafer. Multiple imputation: a primer. *Statistical methods in medical research*, 8(1):3–15, 1999. (cited on Page 102)

Joseph L Schafer and John W Graham. Missing data: our view of the state of the art. *Psychological methods*, 7(2):147, 2002. (cited on Page 102)

A Scharmga, Michiel Peters, A van Tubergen, J Van Den Bergh, Jeroen de Jong, D Loeffen, B van Rietbergen, R Weijers, and PPMM Geusens. Visual detection of cortical breaks in hand joints: reliability and validity of high-resolution peripheral quantitative ct compared to microct. *BMC musculoskeletal disorders*, 17(1):1–7, 2016. (cited on Page 127)

Nathaniel Schenker and Jeremy MG Taylor. Partially parametric techniques for multiple imputation. *Computational statistics & data analysis*, 22(4):425–446, 1996. (cited on Page 102)

Michael Schlegel and Bernd R Noack. On long-term boundedness of galerkin models. *Journal of Fluid Mechanics*, 765:325–352, 2015. (cited on Page 50)

Peter J Schmid. Application of the dynamic mode decomposition to experimental data. *Experiments in fluids*, 50(4):1123–1130, 2011. (cited on Page 56)

Michael Schmidt and Hod Lipson. Distilling free-form natural laws from experimental data. *science*, 324(5923):81–85, 2009. (cited on Page 24 and 51)

Christian Schoelzel and Petra Friederichs. Multivariate non-normally distributed random variables in climate research–introduction to the copula approach. *Nonlinear Processes in Geophysics*, 15(5):761–772, 2008. (cited on Page 235)

Bernhard Schölkopf, Alexander Smola, and Klaus-Robert Müller. Nonlinear component analysis as a kernel eigenvalue problem. *Neural computation*, 10(5):1299–1319, 1998. (cited on Page 59 and 82)

LM Schwartz. Atrophy and programmed cell death of skeletal muscle. *Cell Death & Differentiation*, 15(7):1163–1169, 2008. (cited on Page 3)

Elizabeth Scribner, Olivier Saut, Paula Province, Asim Bag, Thierry Colin, and Hassan M Fathallah-Shaykh. Effects of anti-angiogenesis on glioblastoma growth and migration: model to clinical predictions. *PLoS One*, 9(12):e115018, 2014. (cited on Page 234)

TW Secomb, R Hsu, MW Dewhirst, B Klitzman, and JF Gross. Analysis of oxygen transport to tumor tissue by microvascular networks. *International Journal of Radiation Oncology* Biology* Physics*, 25(3):481–489, 1993. (cited on Page 255)

Richard Semaan, Pradeep Kumar, Marco Burnazzi, Gilles Tissot, Laurent Cordier, and Bernd R Noack. Reduced-order modelling of the flow around a high-lift configuration with unsteady coanda blowing. *Journal of Fluid Mechanics*, 800: 72–110, 2016. (cited on Page 49)

Ataman Sendoel and Michael O Hengartner. Apoptotic cell death under hypoxia. *Physiology*, 29(3):168–176, 2014. (cited on Page 255)

Saptarshi Sengupta, Sanchita Basak, Pallabi Saikia, Sayak Paul, Vasilios Tsalavoutis, Frederick Atiah, Vadlamani Ravi, and Alan Peters. A review of deep learning with special emphasis on architectures, applications and recent trends. *Knowledge-Based Systems*, 194:105596, 2020. (cited on Page 153)

Christoph Settgast, Geralf Hütter, Meinhard Kuna, and Martin Abendroth. A hybrid approach to simulate the homogenized irreversible elastic–plastic deformations and damage of foams by neural networks. *International Journal of Plasticity*, 126: 102624, 2020. (cited on Page 36)

Sagi Shaier, Maziar Raissi, and Padmanabhan Seshaiyer. Data-driven approaches for predicting spread of infectious diseases through dinns: Disease informed neural networks. *arXiv preprint arXiv:2110.05445*, 2021. (cited on Page 49)

Lawrence F Shampine and Mark W Reichelt. The matlab ode suite. *SIAM journal on scientific computing*, 18(1):1–22, 1997. (cited on Page 414)

Lawrence F Shampine, Mark W Reichelt, and Jacek A Kierzenka. Solving index-1 daes in matlab and simulink. *SIAM Review*, 41(3):538–552, 1999. (cited on Page 412)

Claude Elwood Shannon. A mathematical theory of communication. *Bell system technical journal*, 27(3):379–423, 1948. (cited on Page 247)

Dinggang Shen, Guorong Wu, and Heung-Il Suk. Deep learning in medical image analysis. *Annual review of biomedical engineering*, 19:221–248, 2017. (cited on Page 36)

Yuelin Shen, K Chandrashekhara, WF Breig, and LR Oliver. Finite element analysis of v-ribbed belts using neural network based hyperelastic material model. *International Journal of Non-Linear Mechanics*, 40(6):875–890, 2005. (cited on Page 36)

Andrea Shergalis, Armand Bankhead, Urarika Luesakul, Nongnuj Muangsin, and Nouri Neamati. Current challenges and opportunities in treating glioblastoma. *Pharmacological reviews*, 70(3):412–445, 2018. (cited on Page 285)

Jack Sherman and Winifred J Morrison. Adjustment of an inverse matrix corresponding to a change in one element of a given matrix. *The Annals of Mathematical Statistics*, 21(1):124–127, 1950. (cited on Page 410)

Yoojin Shin, Sewoon Han, Jessie S Jeon, Kyoko Yamamoto, Ioannis K Zervantonakis, Ryo Sudo, Roger D Kamm, and Seok Chung. Microfluidic assay for simultaneous culture of multiple cell types on surfaces or within hydrogels. *Nature protocols*, 7 (7):1247–1259, 2012. (cited on Page 9)

Yazdan Shirvany, Mohsen Hayati, and Rostam Moradian. Numerical solution of the nonlinear schrodinger equation by feedforward neural networks. *Communications in Nonlinear Science and Numerical Simulation*, 13(10):2132–2145, 2008. (cited on Page 43)

Ajay Shrestha and Ausif Mahmood. Review of deep learning algorithms and architectures. *IEEE Access*, 7:53040–53065, 2019. (cited on Page 153)

Ravid Shwartz-Ziv and Naftali Tishby. Opening the black box of deep neural networks via information. *ArXiv*, abs/1703.00810, 2017. (cited on Page 25 and 146)

Rebecca L Siegel, Kimberly D Miller, and Ahmedin Jemal. Cancer statistics, 2016. *CA: a cancer journal for clinicians*, 66(1):7–30, 2016. (cited on Page 4)

David Silver, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, et al. Mastering the game of go with deep neural networks and tree search. *Nature*, 529(7587):484–489, 2016. (cited on Page 24)

JC Simo, N Tarnow, and M Doblare. Non-linear dynamics of three-dimensional rods: Exact energy and momentum conserving algorithms. *International Journal for Numerical Methods in Engineering*, 38(9):1431–1473, 1995. (cited on Page 111)

Noah Simon, Jerome Friedman, Trevor Hastie, and Robert Tibshirani. A sparse-group lasso. *Journal of computational and graphical statistics*, 22(2):231–245, 2013. (cited on Page 46)

Anand Pratap Singh, Shivaji Medida, and Karthik Duraisamy. Machine-learning-augmented predictive modeling of turbulent separated flows over airfoils. *AIAA journal*, 55(7):2215–2227, 2017. (cited on Page 57)

Justin Sirignano and Konstantinos Spiliopoulos. Dgm: A deep learning algorithm for solving partial differential equations. *Journal of computational physics*, 375: 1339–1364, 2018. (cited on Page 47)

Robert D Skeel and Martin Berzins. A method for the spatial discretization of parabolic equations in one space variable. *SIAM journal on scientific and statistical computing*, 11(1):1–32, 1990. (cited on Page 306, 411, and 415)

M Sklar. Fonctions de repartition an dimensions et leurs marges. *Publ. inst. statist. univ. Paris*, 8:229–231, 1959. (cited on Page 244)

Marina Skurichina, Sarunas Raudys, and Robert PW Duin. K-nearest neighbors directed noise injection in multilayer perceptron training. *IEEE Transactions on Neural Networks*, 11(2):504–511, 2000. (cited on Page 194)

Benjamin Skuse. The third pillar. *Physics World*, 32(3):40, 2019. (cited on Page 26)

Jonathan MW Slack and David Tosh. Transdifferentiation and metaplasia—switching cell types. *Current opinion in genetics & development*, 11(5):581–586, 2001. (cited on Page 3)

Jasper Snoek, Hugo Larochelle, and Ryan P Adams. Practical bayesian optimization of machine learning algorithms. In *Advances in neural information processing systems*, pages 2951–2959, 2012. (cited on Page 227)

S Solari and MA Losada. Non-stationary wave height climate modeling and simulation. *Journal of Geophysical Research: Oceans*, 116(C9), 2011. (cited on Page 235)

David Sondak and Pavlos Protopapas. Learning a reduced basis of dynamical systems using an autoencoder. *Physical Review E*, 104(3):034202, 2021. (cited on Page 36)

Erika Spissu, Abdul Rawoof Pinjari, Ram M Pendyala, and Chandra R Bhat. A copula-based joint multinomial discrete–continuous model of vehicle type choice and miles of travel. *Transportation*, 36(4):403–422, 2009. (cited on Page 235)

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958, 2014. (cited on Page 66)

Laurent Stainier, Adrien Leygue, and Michael Ortiz. Model-free data-driven methods in mechanics: material data identification and solvers. *Computational Mechanics*, 64(2):381–393, 2019. (cited on Page 59 and 146)

Tracy L Stepien, Erica M Rutter, and Yang Kuang. Traveling waves of a go-or-grow model of glioma growth. *SIAM Journal on Applied Mathematics*, 78(3):1778–1801, 2018. (cited on Page 322)

Russell Stewart and Stefano Ermon. Label-free supervision of neural networks with physics and domain knowledge. In *Thirty-First AAAI Conference on Artificial Intelligence*, 2017. (cited on Page 46)

Brian Stramer and Roberto Mayor. Mechanisms and in vivo functions of contact inhibition of locomotion. *Nature reviews Molecular cell biology*, 18(1):43–55, 2017. (cited on Page 254)

Daniel Strigl, Klaus Kofler, and Stefan Podlipnig. Performance and scalability of gpu-based convolutional neural networks. In *2010 18th Euromicro Conference on Parallel, Distributed and Network-based Processing*, pages 317–324. IEEE, 2010. (cited on Page 152)

Cecilia Suarez, Felipe Maglietti, Mario Colonna, Karina Breitburd, and Guillermo Marshall. Mathematical modeling of human glioma growth based on brain topological structures: study of two clinical cases. *PLoS One*, 7(6):e39616, 2012. (cited on Page 234)

Sainbayar Sukhbaatar, Rob Fergus, et al. Learning multiagent communication with backpropagation. *Advances in neural information processing systems*, 29:2244–2252, 2016. (cited on Page 37)

Natarajan Sukumar, Brian Moran, and Ted Belytschko. The natural element method in solid mechanics. *International journal for numerical methods in engineering*, 43 (5):839–887, 1998. (cited on Page 173)

Luning Sun, Han Gao, Shaowu Pan, and Jian-Xun Wang. Surrogate modeling for fluid flows based on physics-constrained deep learning without simulation data. *Computer Methods in Applied Mechanics and Engineering*, 361:112732, 2020. (cited on Page 49)

Hyuna Sung, Jacques Ferlay, Rebecca L Siegel, Mathieu Laversanne, Isabelle Soerjomataram, Ahmedin Jemal, and Freddie Bray. Global cancer statistics 2020: Globocan estimates of incidence and mortality worldwide for 36 cancers in 185 countries. *CA: a cancer journal for clinicians*, 71(3):209–249, 2021. (cited on Page 4, 5, and 7)

Pierre M Suquet. Elements of homogenization for inelastic solid mechanics, homogenization techniques for composite media. *Lecture notes in physics*, 272:193, 1985. (cited on Page 133)

Gerald Jay Sussman and Jack Wisdom. *Structure and interpretation of classical mechanics*. Mit Press, 2015. (cited on Page 103)

Theodore Sussman and Klaus-Jürgen Bathe. A model of incompressible isotropic hyperelastic material behavior using spline interpolations of tension–compression test data. *Communications in numerical methods in engineering*, 25(1):53–63, 2009. (cited on Page 59, 70, and 164)

Kenji Suzuki. *Artificial neural networks: methodological advances and biomedical applications*. InTech, 2011. (cited on Page 24)

Kristin R Swanson, EC Alvord, and JD Murray. A quantitative model for differential motility of gliomas in grey and white matter. *Cell Proliferation*, 33(5):317–329, 2000. (cited on Page 259)

Kristin R. Swanson, Russell C. Rockne, Jonathan Claridge, Mark A. Chaplain, Ellsworth C. Alvord, and Alexander R.A. Anderson. Quantifying the role of angiogenesis in malignant progression of gliomas: In silico modeling integrates imaging and histology. *Cancer Research*, 71(24):7366–7375, 2011. ISSN 0008-5472. doi: 10.1158/0008-5472.CAN-11-1399. URL http://cancerres.aacrjournals.org/content/71/24/7366. (cited on Page 234 and 287)

Pavel Sztefek, Maximilien Vanleene, Robin Olsson, Rebecca Collinson, Andrew A Pitsillides, and Sandra Shefelbine. Using digital image correlation to determine bone surface strains during loading and after adaptation of the mouse tibia. *Journal of biomechanics*, 43(4):599–605, 2010. (cited on Page 59 and 127)

Barbara G Tabachnick, Linda S Fidell, and Steven J Osterlind. *Using multivariate statistics*. Allyn and Bacon Boston, 2001. (cited on Page 103)

Mohammad Tabatabai, David Keith Williams, and Zoran Bursac. Hyperbolastic growth models: theory and application. *Theoretical Biology and Medical Modelling*, 2(1):1–13, 2005. (cited on Page 239)

Kuangshin Tai, Ming Dao, Subra Suresh, Ahmet Palazoglu, and Christine Ortiz. Nanoscale heterogeneity promotes energy dissipation in bone. *Nature materials*, 6 (6):454–462, 2007. (cited on Page 127)

Pei-Sung Tang. On the rate of oxygen consumption by tissues and lower organisms as a function of oxygen tension. *The Quarterly Review of Biology*, 8(3):260–274, 1933. (cited on Page 256 and 295)

Ian F Tannock. Oxygen diffusion and the distribution of cellular radiosensitivity in tumours. *The British journal of radiology*, 45(535):515–524, 1972. (cited on Page 255, 261, and 296)

Fang Fang Tao, Xia Xiao, Kin Fong Lei, and I-Chi Lee. Based cell culture microfluidic system. *BioChip Journal*, 9(2):97–104, 2015. (cited on Page 297)

Albert Tarantola. *Inverse problem theory and methods for model parameter estimation*. SIAM, 2005. (cited on Page 60)

Alexandre M Tartakovsky, Carlos Ortiz Marrero, Paris Perdikaris, Guzel D Tartakovsky, and David Barajas-Solano. Learning parameters and constitutive relationships with physics informed deep neural networks. *arXiv preprint arXiv:1808.03398*, 2018. (cited on Page 60 and 67)

Z Tatárová, JP Abbuehl, S Maerkl, and J Huelsken. Microfluidic co-culture platform to quantify chemotaxis of primary stem cells. *Lab on a Chip*, 16(10):1934–1945, 2016. (cited on Page 9)

Gregory H Teichert, AR Natarajan, A Van der Ven, and Krishna Garikipati. Machine learning materials physics: Integrable deep neural networks enable scale bridging by learning free energy functions. *Computer Methods in Applied Mechanics and Engineering*, 353:201–216, 2019. (cited on Page 36)

Joshua B Tenenbaum, Vin De Silva, and John C Langford. A global geometric framework for nonlinear dimensionality reduction. *science*, 290(5500):2319–2323, 2000. (cited on Page 59 and 82)

Kenjiro Terada and Noboru Kikuchi. A class of general algorithms for multi-scale analyses of heterogeneous media. *Computer methods in applied mechanics and engineering*, 190(40-41):5427–5464, 2001. (cited on Page 126)

Jules Thibault and Bernard PA Grandjean. A neural network methodology for heat transfer data analysis. *International Journal of Heat and Mass Transfer*, 34(8): 2063–2070, 1991. (cited on Page 158)

RH Thomlinson and LH Gray. The histological structure of some human lung cancers and the possible implications for radiotherapy. *British Journal of Cancer*, 9(4): 539, 1955. (cited on Page 261)

Michael L Thompson and Mark A Kramer. Modeling chemical processes using prior knowledge and neural networks. *AIChE Journal*, 40(8):1328–1340, 1994. (cited on Page 41)

MS Thompson, H Schell, J Lienau, and GN Duda. Digital image correlation: a technique for determining local mechanical conditions within early bone callus. *Medical engineering & physics*, 29(7):820–823, 2007. (cited on Page 127)

Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, 58(1):267–288, 1996. (cited on Page 45)

Jane S Tjia and Prabhas V Moghe. Regulation of cell motility on polymer substrates via" dynamic," cell internalizable, ligand microinterfaces. *Tissue Engineering*, 8(2): 247–261, 2002. (cited on Page 258)

Giang Tran and Rachel Ward. Exact recovery of chaotic systems from highly corrupted data. *Multiscale Modeling & Simulation*, 15(3):1108–1129, 2017. (cited on Page 52)

Rohit K Tripathy and Ilias Bilionis. Deep uq: Learning deep neural network surrogate models for high dimensional uncertainty quantification. *Journal of computational physics*, 375:565–588, 2018. (cited on Page 35 and 36)

Anastasios Tsoularis and James Wallace. Analysis of logistic growth models. *Mathematical biosciences*, 179(1):21–55, 2002. (cited on Page 295)

Ioannis G Tsoulos, Dimitris Gavrilis, and Euripidis Glavas. Solving differential equations with constructed neural networks. *Neurocomputing*, 72(10-12):2385–2391, 2009. (cited on Page 43)

Jonathan H Tu. *Dynamic mode decomposition: Theory and applications*. PhD thesis, Princeton University, 2013. (cited on Page 56)

Silviu-Marian Udrescu and Max Tegmark. Ai feynman: A physics-inspired method for symbolic regression. *Science Advances*, 6(16):eaay2631, 2020. (cited on Page 24)

Jörg F Unger and Carsten Könke. Neural networks as material models within a multiscale approach. *Computers & structures*, 87(19-20):1177–1186, 2009. (cited on Page 36)

Laurens Van Der Maaten, Eric Postma, Jaap Van den Herik, et al. Dimensionality reduction: a comparative. *J Mach Learn Res*, 10(66-71):13, 2009. (cited on Page 35)

B Ph van Milligen, V Tribaldos, and JA Jiménez. Neural network differential equation and plasma equilibrium solver. *Physical review letters*, 75(20):3594, 1995. (cited on Page 47)

Eric Vanden-Eijnden et al. Transition-path theory and path-finding algorithms for the study of rare events. *Annual review of physical chemistry*, 61:391–420, 2010. (cited on Page 44)

Vladimir Vapnik. *The nature of statistical learning theory*. Springer science & business media, 2013. (cited on Page 24)

Peter Vaupel, Friedrich Kallinowski, and Paul Okunieff. Blood flow, oxygen and nutrient supply, and metabolic microenvironment of human tumors: a review. *Cancer Research*, 49(23):6449–6465, 1989. (cited on Page 259, 260, 261, and 262)

Maria Vinci, Sharon Gowan, Frances Boxall, Lisa Patterson, Miriam Zimmermann, Cara Lomas, Marta Mendiola, David Hardisson, Suzanne A Eccles, et al. Advances in establishment and analysis of three-dimensional tumor spheroid-based functional assays for target validation and drug evaluation. *BMC biology*, 10(1):1–21, 2012. (cited on Page 297)

Francisco G Vital-Lopez, Antonios Armaou, Michelle Hutnik, and Costas D Maranas. Modeling the effect of chemotaxis on glioblastoma tumor progression. *AIChE journal*, 57(3):778–792, 2011. (cited on Page 259, 260, 261, 262, and 287)

Nikolaos N Vlassis and WaiChing Sun. Sobolev training of thermodynamic-informed neural networks for interpretable elasto-plasticity models with level set hardening. *Computer Methods in Applied Mechanics and Engineering*, 377:113695, 2021. (cited on Page 41)

Nikolaos N Vlassis, Ran Ma, and WaiChing Sun. Geometric deep learning for computational mechanics part i: Anisotropic hyperelasticity. *Computer Methods in Applied Mechanics and Engineering*, 371:113299, 2020. (cited on Page 37)

Klaus Von Der Mark, Verena Gauss, Helga Von Der Mark, and Peter Müller. Relationship between cell shape and type of collagen synthesised as chondrocytes lose their cartilage phenotype in culture. *Nature*, 267(5611):531–532, 1977. (cited on Page 9)

H Voss, MJ Bünner, and Markus Abel. Identification of continuous, spatiotemporal systems. *Physical Review E*, 57(3):2820, 1998. (cited on Page 51)

Nicholas Wagner and James M Rondinelli. Theory-guided machine learning in materials science. *Frontiers in Materials*, 3:28, 2016. (cited on Page 46)

Matt P Wand and M Chris Jones. *Kernel smoothing*. CRC press, 1994. (cited on Page 244)

Chengyang Wang, Zhenyu Tang, Yu Zhao, Rui Yao, Lingsong Li, and Wei Sun. Three-dimensional in vitro cancer models: a short review. *Biofabrication*, 6(2): 022001, 2014. (cited on Page 8)

Christina H Wang, Jason K Rockhill, Maciej Mrugala, Danielle L Peacock, Albert Lai, Katy Jusenius, Joanna M Wardlaw, Timothy Cloughesy, Alexander M Spence, Russ Rockne, et al. Prognostic significance of growth kinetics in newly diagnosed glioblastomas revealed by combining serial imaging with a novel biomathematical model. *Cancer Research*, 69(23):9133–9140, 2009. (cited on Page 258, 259, and 262)

Jian-Xun Wang, Jin-Long Wu, and Heng Xiao. Physics-informed machine learning approach for reconstructing reynolds stress modeling discrepancies based on dns data. *Physical Review Fluids*, 2(3):034603, 2017. (cited on Page 57)

Jikun Wang, Tianjiao Li, Fan Cui, Chung-Yuen Hui, Jingjie Yeo, and Alan T Zehnder. Metamodeling of constitutive model using gaussian process machine learning. *Journal of the Mechanics and Physics of Solids*, page 104532, 2021a. (cited on Page 36)

Kun Wang and WaiChing Sun. A multiscale multi-permeability poroplasticity model linked by recursive homogenizations and deep learning. *Computer Methods in Applied Mechanics and Engineering*, 334:337–380, 2018. (cited on Page 36 and 37)

Nanzhe Wang, Dongxiao Zhang, Haibin Chang, and Heng Li. Deep learning of subsurface flow via theory-guided neural network. *Journal of Hydrology*, 584: 124700, 2020. (cited on Page 49)

Nanzhe Wang, Haibin Chang, and Dongxiao Zhang. Theory-guided auto-encoder for surrogate construction and inverse modeling. *Computer Methods in Applied Mechanics and Engineering*, 385:114037, 2021b. (cited on Page 49)

Sifan Wang, Yujun Teng, and Paris Perdikaris. Understanding and mitigating gradient flow pathologies in physics-informed neural networks. *SIAM Journal on Scientific Computing*, 43(5):A3055–A3081, 2021c. (cited on Page 49)

Sifan Wang, Xinling Yu, and Paris Perdikaris. When and why pinns fail to train: A neural tangent kernel perspective. *Journal of Computational Physics*, 449:110768, 2022. (cited on Page 49)

Zhenlin Wang, Xun Huan, and Krishna Garikipati. Variational system identification of the partial differential equations governing the physics of pattern-formation: Inference under varying fidelity and noise. *Computer Methods in Applied Mechanics and Engineering*, 356:44–74, 2019. (cited on Page 55)

E Weinan, Jiequn Han, and Arnulf Jentzen. Deep learning-based numerical methods for high-dimensional parabolic partial differential equations and backward stochastic differential equations. *Communications in Mathematics and Statistics*, 5(4): 349–380, 2017. (cited on Page 44)

Sara Weis, Jianhua Cui, Leo Barnes, and David Cheresh. Endothelial barrier disruption by vegf-mediated src activity potentiates tumor cell extravasation and metastasis. *The Journal of cell biology*, 167(2):223–229, 2004. (cited on Page 317)

Julius Weisbach. Lehrbuch der ingenieur-und maschinen. *Mechanik*, 1:895, 1845. (cited on Page 176)

Hassler Whitney. The singularities of a smooth n-manifold in (2n-1)-space. *Annals of Mathematics*, pages 247–293, 1944. (cited on Page 73)

Karen Willcox and Jaime Peraire. Balanced model reduction via the proper orthogonal decomposition. *AIAA journal*, 40(11):2323–2330, 2002. (cited on Page 34)

Christopher K Williams and Carl Edward Rasmussen. *Gaussian processes for machine learning*. MIT press Cambridge, MA, 2006. (cited on Page 37)

Gardner Stewart Williams and Allen Hazen. *Hydraulic Tables: The Elements of Gagings and the Friction of Water Flowing in Pipes, Aqueducts, Sewers, Etc. as Determined by the Hazen and Williams Formula and the Flow of Water Over Sharp-edged and Irregular Weirs, and the Quantity Discharged, as Determined by Bazin's Formula and Experimental Investigations Upon Large Models*. J. Wiley & sons, 1908. (cited on Page 176)

David F Wilson. Quantifying the role of oxygen pressure in tissue function. *American Journal of Physiology-Heart and Circulatory Physiology*, 294(1):H11–H13, 2008. (cited on Page 259)

D Wirtz, N Karajan, and B Haasdonk. Surrogate modeling of multiscale models using kernel methods. *International Journal for Numerical Methods in Engineering*, 101 (1):1–28, 2015. (cited on Page 37)

Danli Wu and Patricia Yotnda. Induction and testing of hypoxia in cell culture. *JoVE (Journal of Visualized Experiments)*, (54):e2899, 2011. (cited on Page 9)

Ling Wu, Nanda Gopala Kilingar, Ludovic Noels, et al. A recurrent neural network-accelerated multi-scale model for elasto-plastic heterogeneous materials subjected to random cyclic and non-proportional loading paths. *Computer Methods in Applied Mechanics and Engineering*, 369:113234, 2020a. (cited on Page 36 and 37)

Zhe Wu, David Rincon, and Panagiotis D Christofides. Process structure-based recurrent neural network modeling for model predictive control of nonlinear processes. *Journal of Process Control*, 89:74–84, 2020b. (cited on Page 41)

Heng Xiao, J-L Wu, J-X Wang, Rui Sun, and CJ Roy. Quantifying and reducing model-form uncertainties in reynolds-averaged navier–stokes simulations: A data-driven, physics-informed bayesian approach. *Journal of Computational Physics*, 324:115–136, 2016. (cited on Page 57)

SP Xiao and Ted Belytschko. A bridging domain method for coupling continua with molecular dynamics. *Computer methods in applied mechanics and engineering*, 193(17):1645–1669, 2004. (cited on Page 25)

Yang Xiao, Dongjoo Kim, Burak Dura, Kerou Zhang, Runchen Yan, Huamin Li, Edward Han, Joshua Ip, Pan Zou, Jun Liu, et al. Ex vivo dynamics of human glioblastoma cells in a microvasculature-on-a-chip system correlates with tumor heterogeneity and subtypes. *Advanced Science*, 6(8):1801531, 2019. (cited on Page 285)

Yinjiao Xing, Qiang Miao, K-L Tsui, and Michael Pecht. Prognostics and health monitoring for lithium-ion battery. In *Intelligence and Security Informatics (ISI), 2011 IEEE International Conference on*, pages 242–247. IEEE, 2011. (cited on Page 70)

Dongbin Xiu and George Em Karniadakis. The wiener–askey polynomial chaos for stochastic differential equations. *SIAM journal on scientific computing*, 24(2): 619–644, 2002. (cited on Page 49)

Feiyu Xu, Hans Uszkoreit, Yangzhou Du, Wei Fan, Dongyan Zhao, and Jun Zhu. Explainable ai: A brief survey on history, research areas, approaches and challenges. In *CCF international conference on natural language processing and Chinese computing*, pages 563–574. Springer, 2019. (cited on Page 25 and 146)

Kailai Xu, Daniel Z Huang, and Eric Darve. Learning constitutive relations using symmetric positive definite neural networks. *Journal of Computational Physics*, 428:110072, 2021. (cited on Page 41)

X Frank Xu and Xi Chen. Stochastic homogenization of random elastic multi-phase composites and size quantification of representative volume element. *Mechanics of Materials*, 41(2):174–186, 2009. (cited on Page 92)

Ying Xue. An overview of overfitting and its solutions. *J. Phys. Conf. Ser*, 1168: 022022, 6 2019. (cited on Page 25 and 146)

Neha Yadav, Anupam Yadav, Manoj Kumar, et al. *An introduction to neural network methods for differential equations*. Springer, 2015. (cited on Page 47, 152, and 163)

Genki Yagawa and H Okuda. Neural networks in computational mechanics. *Archives of Computational Methods in Engineering*, 3(4):435–512, 1996. (cited on Page 36)

Shibo Yan, Xi Zou, Mohammad Ilkhani, and Arthur Jones. An efficient multiscale surrogate modelling framework for composite materials considering progressive damage based on artificial neural networks. *Composites Part B: Engineering*, 194: 108014, 2020. (cited on Page 37)

Liu Yang, Xuhui Meng, and George Em Karniadakis. B-pinns: Bayesian physics-informed neural networks for forward and inverse pde problems with noisy data. *Journal of Computational Physics*, 425:109913, 2021. (cited on Page 49)

XIA Yang, Suhaib Zafar, J-X Wang, and Heng Xiao. Predictive large-eddy-simulation wall modeling via physics-informed neural networks. *Physical Review Fluids*, 4(3): 034602, 2019. (cited on Page 49)

Yibo Yang and Paris Perdikaris. Adversarial uncertainty quantification in physics-informed neural networks. *Journal of Computational Physics*, 394:136–152, 2019. (cited on Page 49)

Alireza Yazdani, Lu Lu, Maziar Raissi, and George Em Karniadakis. Systems biology informed deep learning for inferring parameters and hidden dynamics. *PLoS computational biology*, 16(11):e1007575, 2020. (cited on Page 49 and 146)

Hee-Gyeong Yi, Young Hun Jeong, Yona Kim, Yeong-Jin Choi, Hyo Eun Moon, Sung Hye Park, Kyung Shin Kang, Mihyeon Bae, Jinah Jang, Hyewon Youn, et al. A bioprinted human-glioblastoma-on-a-chip for the identification of patient-specific responses to chemoradiotherapy. *Nature biomedical engineering*, 3(7):509–519, 2019. (cited on Page 285)

Hyeon-Joong Yoo. Deep convolution neural networks in computer vision: a review. *IEIE Transactions on Smart Processing & Computing*, 4(1):35–43, 2015. (cited on Page 147)

Bing Yu et al. The deep ritz method: a deep learning-based numerical algorithm for solving variational problems. *arXiv preprint arXiv:1710.00211*, 2017. (cited on Page 48)

Ming Yuan and Yi Lin. Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 68(1):49–67, 2006. (cited on Page 46)

Zheng Yuan and Jacob Fish. Toward realization of computational homogenization in practice. *International Journal for Numerical Methods in Engineering*, 73(3): 361–380, 2008. (cited on Page 126)

Ma Yunquan and Fu Yun. Manifold learning theory and applications, 2011. (cited on Page 24)

Eleftherios C Zachmanoglou and Dale W Thoe. *Introduction to partial differential equations with applications*. Courier Corporation, 1986. (cited on Page 323 and 324)

Ioannis K Zervantonakis, Shannon K Hughes-Alford, Joseph L Charest, John S Condeelis, Frank B Gertler, and Roger D Kamm. Three-dimensional microfluidic model for tumor cell intravasation and endothelial barrier function. *Proceedings of the National Academy of Sciences*, 109(34):13515–13520, 2012. (cited on Page 9)

Dongkun Zhang, Ling Guo, and George Em Karniadakis. Learning in modal space: Solving time-dependent stochastic pdes using physics-informed neural networks. *SIAM Journal on Scientific Computing*, 42(2):A639–A665, 2020. (cited on Page 49)

Linan Zhang and Hayden Schaeffer. On the convergence of the sindy algorithm. *Multiscale Modeling & Simulation*, 17(3):948–972, 2019. (cited on Page 52)

Ning Zhang, Shui-Long Shen, Annan Zhou, and Yin-Fu Jin. Application of lstm approach for modelling stress–strain behaviour of soil. *Applied Soft Computing*, 100:106959, 2021. (cited on Page 36)

Hongbo Zhao, Brian D Storey, Richard D Braatz, and Martin Z Bazant. Learning the physics of pattern formation from images. *Physical review letters*, 124(6):060201, 2020. (cited on Page 60)

Yinhao Zhu and Nicholas Zabaras. Bayesian deep convolutional encoder–decoder networks for surrogate modeling and uncertainty quantification. *Journal of Computational Physics*, 366:415–447, 2018. (cited on Page 36)

Yinhao Zhu, Nicholas Zabaras, Phaedon-Stelios Koutsourelakis, and Paris Perdikaris. Physics-constrained deep learning for high-dimensional surrogate modeling and uncertainty quantification without labeled data. *Journal of Computational Physics*, 394:56–81, 2019. (cited on Page 49)

Olgierd Cecil Zienkiewicz, Robert Leroy Taylor, Perumal Nithiarasu, and JZ Zhu. *The finite element method*, volume 3. McGraw-hill London, 1977. (cited on Page 30 and 165)

Markus Zimmermann and Johannes Edler von Hoessle. Computing solution spaces for robust design. *International Journal for Numerical Methods in Engineering*, 94 (3):290–307, 2013. (cited on Page 36)

Helene Zirath, Mario Rothbauer, Sarah Spitz, Barbara Bachmann, Christian Jordan, Bernhard Müller, Josef Ehgartner, Eleni Priglinger, Severin Mühleder, Heinz Redl, et al. Every breath you take: non-invasive real-time oxygen biosensing in two-and three-dimensional microfluidic cell models. *Frontiers in physiology*, 9:815, 2018. (cited on Page 299)

Tarek I Zohdi and Peter Wriggers. *An introduction to computational micromechanics*. Springer Science & Business Media, 2008. (cited on Page 92)

Yajie Zou and Yunlong Zhang. A copula-based approach to accommodate the dependence among microscopic traffic variables. *Transportation Research Part C: Emerging Technologies*, 70:53–68, 2016. (cited on Page 235)

I herewith assure that I wrote the present thesis independently, that the thesis has not been partially or fully submitted as graded academic work and that I have used no other means than the ones indicated. I have indicated all parts of the work in which sources are used according to their wording or to their meaning.

Zaragoza, January 18, 2022