TESIS DE LA UNIVERSIDAD
DE ZARAGOZA

2014

23

Iñaki Asier Iturrate Gil

Robot Learning and Control Using Error-Related Cognitive Brain Signals

Departamento

Informática e Ingeniería de Sistemas

Director/es

Mínguez Zafra, Javier
Montesano del Campo, Luis

ISSN 2254-7606

Prensas de la Universidad
Universidad Zaragoza

Tesis Doctoral

ROBOT LEARNING AND CONTROL USING ERROR-RELATED COGNITIVE BRAIN SIGNALS

Autor

Iñaki Asier Iturrate Gil

Director/es

Mínguez Zafra, Javier
Montesano del Campo, Luis

UNIVERSIDAD DE ZARAGOZA
Informática e Ingeniería de Sistemas

2014

Robot learning and control using error-related cognitive brain signals

Iñaki Asier Iturrate Gil

Thesis submitted in fulfillment of the requirements for the degree of Doctor of Philosophy in Computer Science and Systems Engineering

Supervised by Javier Minguez and Luis Montesano

Instituto de Investigación en Ingeniería de Aragón (I3A)
Departamento de Informática e Ingeniería de Sistemas (DIIS)
Escuela de Ingeniería y Arquitectura (EINA)
Universidad de Zaragoza
November, 2013

Acknowledgements

First of all, I would like to thank my advisors Javier Minguez and Luis Montesano for their patience, assistance and guidance during the good and the bad moments. If I had to choose only one of the previous three terms, I would surely choose patience. I strongly hope the professional and personal links forged keep going on in the future.

Carlos, Edu, Mauricio, my present for you is that you begin this paragraph. This work would not have been possible without my thesis brothers (the best anyone could have) Carlos Escolano, Edu López and Mauricio Antelis. They have been my friends long before I started it, they have always been there during it, and I am pretty sure they will be there afterwards. Jason Omedes, our last vcizar mate, I do not forget you either. Thanks for re-shaping the lab and being such a good guy. I am very happy to have met you. And of course, I wanted to thank all the workers from Bitbrain, especially Dani. María López, thank you for always trusting in me and giving me the opportunity to enter in this amazing world.

I am strongly indebted to my advisors in Lausanne José del R. Millán and Ricardo Chavarriaga. Thanks to them, I was able to grow and see things from a different point of view. I could not forget thanking at all the teammates from the CNBI, and specially Andrea Biasiucci and Michele Tavella for taking care of me. Fabiola, I could not forget you either. I am also in debt with Manuel Lopes and Jonathan Grizou, which helped me (and are helping right now) in the completion of my thesis.

My family has always been there when I needed it for giving me support and their shoulder. Dad, you have always been my example to follow. Mom, you are simply the best. My incredible brothers, Iker (and Maria), Joseba (and Miriam), thank you for all your counsel and caring. To Ramón and Elena, the best parents-in-law one could ever wish.

My friends, nothing could be possible without you. TPM (and ladies), this goes for you: Buka and Nahikari (ponme otro que lo he tirado), Gastón (vamos a ver a los ciclos), Ichi and Irene (colectivo viernes), Latorre and Yoli (pirotecnia?), Motilva and Isa (a ver cuándo nos pegamos) and Oscar (y no pocou). Motilva, Ichi, Gastón, por los momentos duros. Popy, Tamara, Miguel, Estefania, César, Ana, you became closer than I would have never imagined. Matthias osaesmamado, mein lieber Deutsch, thank you^n. Next time I will throw your shoe further. Romeo, R, Jarmok, decrack, Morfeo, siempre nos quedará Juslibol. Sinu, Pablo, Raul, you had to be here. To the array of pointers to Informáticos. To the Redima F7 team. Thank you all.

This thesis is for you, María. You are my light in the dark, I love you.

Resumen

Durante los últimos años, el campo de los interfaces cerebro-máquina (BMIs en inglés) ha demostrado cómo humanos y animales son capaces de controlar dispositivos neuroprotésicos directamente de la modulación voluntaria de sus señales cerebrales, tanto en aproximaciones invasivas como no invasivas. Todos estos BMIs comparten un paradigma común, donde el usuario trasmite información relacionada con el control de la neuroprótesis. Esta información se recoge de la actividad cerebral del usuario, para luego ser traducida en comandos de control para el dispositivo. Cuando el dispositivo recibe y ejecuta la orden, el usuario recibe una retroalimentación del rendimiento del sistema, cerrando de esta manera el bucle entre usuario y dispositivo.

La mayoría de los BMIs decodifican parámetros de control de áreas corticales para generar la secuencia de movimientos para la neuroprótesis. Esta aproximación simula al control motor típico, dado que enlaza la actividad neural con el comportamiento o la ejecución motora. La ejecución motora, sin embargo, es el resultado de la actividad combinada del córtex cerebral, áreas subcorticales y la médula espinal. De hecho, numerosos movimientos complejos, desde la manipulación a andar, se tratan principalmente al nivel de la médula espinal, mientras que las áreas corticales simplemente proveen el punto del espacio a alcanzar y el momento de inicio del movimiento.

Esta tesis propone un paradigma BMI alternativo que trata de emular el rol de los niveles subcorticales durante el control motor. El paradigma se basa en señales cerebrales que transportan información cognitiva asociada con procesos de toma de decisiones en movimientos orientados a un objetivo, y cuya implementación de bajo nivel se maneja en niveles subcorticales. A lo largo de la tesis, se presenta el primer paso hacia el desarrollo de este paradigma centrándose en una señal cognitiva específica relacionada con el procesamiento de errores humano: los potenciales de error (ErrPs) medibles mediante electroencefalograma (EEG). En esta propuesta de paradigma, la neuroprótesis ejecuta activamente una tarea de alcance mientras el usuario simplemente monitoriza el rendimiento del dispositivo mediante la evaluación de la calidad de las acciones ejecutadas por el dispositivo. Estas evaluaciones se traducen (gracias a los ErrPs) en retroalimentación para el dispositivo, el cual las usa en un contexto de aprendizaje por refuerzo para mejorar su comportamiento. Esta tesis demuestra por primera vez este paradigma BMI de enseñanza con doce sujetos en tres experimentos en bucle cerrado concluyendo con la operación de un manipulador robótico real.

Como la mayoría de BMIs, el paradigma propuesto requiere una etapa de calibración específica para cada sujeto y tarea. Esta fase, un proceso que requiere mucho tiempo y extenuante para el usuario, dificulta la distribución de los BMIs a aplicaciones fuera del laboratorio. En el caso particular del paradigma propuesto, una fase de calibración para cada tarea es altamente impráctico ya que el tiempo necesario para esta fase se suma al tiempo de aprendizaje de la tarea, retrasando sustancialmente el control final del dispositivo. Así, sería conveniente poder entrenar clasificadores capaces de funcionar independientemente de la tarea de aprendizaje que se esté ejecutando. Esta tesis analiza desde un punto de vista electrofisiológico cómo los potenciales se ven afectados por diferentes tareas ejecutadas por el dispositivo, mostrando cambios principalmente en la latencia la señal; y estudia cómo transferir el clasificador entre tareas de dos maneras: primero, aplicando clasificadores adaptativos del estado del arte, y segundo corrigiendo la latencia entre las señales de dos tareas para poder generalizar entre ambas.

Otro reto importante bajo este paradigma viene del tiempo necesario para aprender la tarea. Debido al bajo ratio de información transferida por minuto del BMI, el sistema tiene una pobre escalabilidad: el tiempo de aprendizaje crece exponencialmente con el tamaño del espacio de aprendizaje, y por tanto resulta impráctico obtener el comportamiento motor óptimo mediante aprendizaje por refuerzo. Sin embargo, este problema puede resolverse explotando la estructura de la tarea de aprendizaje. Por ejemplo, si el número de posiciones a alcanzar es discreto se puede pre-calcular la política óptima para cada posible posición. En esta tesis, se muestra cómo se puede usar la estructura de la tarea dentro del paradigma propuesto para reducir enormemente el tiempo de aprendizaje de la tarea (de diez minutos a apenas medio minuto), mejorando enormemente así la escalabilidad del sistema.

Finalmente, esta tesis muestra cómo, gracias a las lecciones aprendidas en los descubrimientos anteriores, es posible eliminar completamente la etapa de calibración del paradigma propuesto mediante el aprendizaje no supervisado del clasificador al mismo tiempo que se está ejecutando la tarea. La idea fundamental es calcular un conjunto de clasificadores que sigan las restricciones de la tarea anteriormente usadas, para a continuación seleccionar el mejor clasificador del conjunto. De esta manera, esta tesis presenta un BMI *plug-and-play* que sigue el paradigma propuesto, aprende la tarea y el clasificador y finalmente alcanza la posición del espacio deseada por el usuario.

Abstract

Over the last years, the field of brain-machine interfaces (BMIs) has demonstrated how humans and animals can operate neuroprosthetic devices directly from a voluntary modulation of their brain signals, using both invasive and non-invasive approaches. All these BMIs share a common paradigm, where the user conveys a variety of information related to the operation of the neuroprosthesis. This information is recorded from the user's brain activity, and translated into control commands to operate a neuroprosthetic device. Whenever the device executes a command, the user receives a feedback of the system performance, closing this way the loop between user and device.

Most BMIs decode control parameters from cortical areas in order to generate the sequence of movements for the neuroprosthesis. This approach resembles normal motor control since it links neural activity to motor behavior or to motor execution. Motor control, however, is the result of the combined activity of the cerebral cortex, subcortical areas and spinal cord. In fact, many elements of skilled movements, from manipulation to walking, are mainly handled at the spinal cord level with cortical areas providing just goals and movement onset.

This thesis proposes an alternative BMI paradigm that tries to emulate the role of subcortical levels during motor control. This paradigm is based on human brain signals that carry cognitive information associated to decision-making processes that arise during goal-directed movements whose low-level implementation is handled at the subcortical and spinal cord level. Throughout the thesis, we present the first step towards developing this paradigm by focusing on a specific cognitive signal related to the user's error processing: the electroencephalogram (EEG) error-related potentials (ErrPs). In this approach, the neuroprosthesis is actively performing a reaching task while the user simply monitors the performance of the device by assessing the quality of its actions. These assessments –via the decoded ErrP signals– are translated into feedback to the device, which uses this information within a reinforcement learning context to improve its behavior. This thesis firstly demonstrates this teaching BMI paradigm in a series of three closed-loop

experiments of increasing complexity involving twelve subjects, concluding with the operation of a real robot manipulator.

As most BMIs, the proposed paradigm requires a calibration phase specific to each subject and task. This phase, which is a tiring and time-consuming operation for the user, hinders the deployment of BMIs out of the lab. In the particular case of the proposed paradigm, it would be impractical to calibrate the system for every task, as this time adds to the task learning time and the device operation would be largely delayed. Thus, it would be convenient to train classifiers able to work irrespectively of the learning task being performed. This thesis analyzes, from an electrophysiology point of view, how the error-related potentials are affected by different tasks performed by a device, showing how the latency of the signals vary depending on the task; and studies how to transfer learning between tasks in two ways: firstly applying state-of-the-art adaptive classifiers, and secondly correcting the latency between the signals of two tasks as a way to generalize a classifier among them.

Another important challenge under this paradigm comes from the time needed to learn the task. Due to the low information transfer rate of a BMI, the system has a poor scalability: The learning time grows exponentially as the learning space increases, and thus it becomes impractical to obtain the optimal motor behavior via reinforcement learning. However, this issue can be solved by exploiting the learning task structure. For instance, a discrete number of positions to reach can be exploited by computing their correspondent optimal motor behaviors. In this thesis, we show how this task structure can be used within the proposed BMI paradigm to greatly reduce the time needed to learn the task (from tens of minutes to even half a minute), greatly improving this way the scalability of the system.

Finally, this thesis show how, thanks to the lessons learned on the previous findings, it is possible to completely remove the calibration phase of the proposed paradigm by learning unsupervisedly the classifier while the task is being executed. The main idea was to compute a set of classifiers based on the task constraints previously used and choosing the most suitable one for the signals. This way, this thesis presents a plug-and-play BMI that follows the described paradigm, learns the task and the classifier and finally reaches the desired position by the user.

Contents

Abbreviations

N_e	Error-related negativity
P_e	Error-related positivity
ACC	Anterior cingulate cortex
BCI	Brain-computer interface
BMI	Brain-machine interface
CAR	Common-average reference
EEG	Electroencephalography
ERN	Error-related negativity
ERP	Event-related potentials
ErrP	Error-related potentials
fMRI	Functional magnetic resonance imaging
FRN	Feedback-related negativity
GA	Grand average
IRL	Inverse reinforcement learning
LP	Low pass
RL	Reinforcement learning
SCI	Spinal cord injury
SMA	Supplementary motor area
SNR	Signal-to-noise ratio
STF	Spatio-temporal filter

1 | Introduction

Brain-machine interfacing (BMI) is an emergent technology developed to provide a communication channel between a human and a device using only brain activity [1]. These systems have been successfully used in four different applications: communication and control, motor restoration, entertainment and games, and motor rehabilitation [2]. The most significant feature that defines a BMI is the method used for the electrophysiological recordings [3]. On one hand, invasive BMIs, such as single-neuron or multiple-neuron recordings, or electrocorticography (ECoG), require surgical procedures as they rely on intracranial electrodes for the signal acquisition, but provide a high quality signal allowing a complex control of different devices. On the other hand, non-invasive BMIs, primarily the electroencephalography (EEG), do not require any surgical intervention but usually suffer from low signal-to-noise ratios. In applications involving motor control, these systems have mainly focused on the operation of neuroprosthetic devices such as mobile robots, robotic wheelchairs or robotic arms [2].

Over the last years, the field of BMI has witnessed impressive demonstrations of how humans and animals can control the operation of the aforementioned neuroprosthetic devices directly from a voluntary modulation of their brain signals, using both invasive and non-invasive approaches [4–14]. All these BMIs share a common approach, namely the control paradigm (see Figure 1.1). In this paradigm, the user conveys a variety of information related to the operation of the neuroprosthesis, ranging from continuous velocity/position [4, 8, 12, 14–19] and muscular activity [11] to rather discrete states such as directions [5, 10, 13], targets [6, 9], hand opening/closing [4, 7], and self-initiation of movements [20–22]. This information is recorded from the user's brain activity, and translated into control commands to operate a neuroprosthetic device. Whenever the device executes a command, the user receives a feedback of the system performance, closing this way the loop between user and device [23].

Most of these BMIs decode control parameters from cortical areas in order to generate the sequence of movements for the neuroprosthesis. This

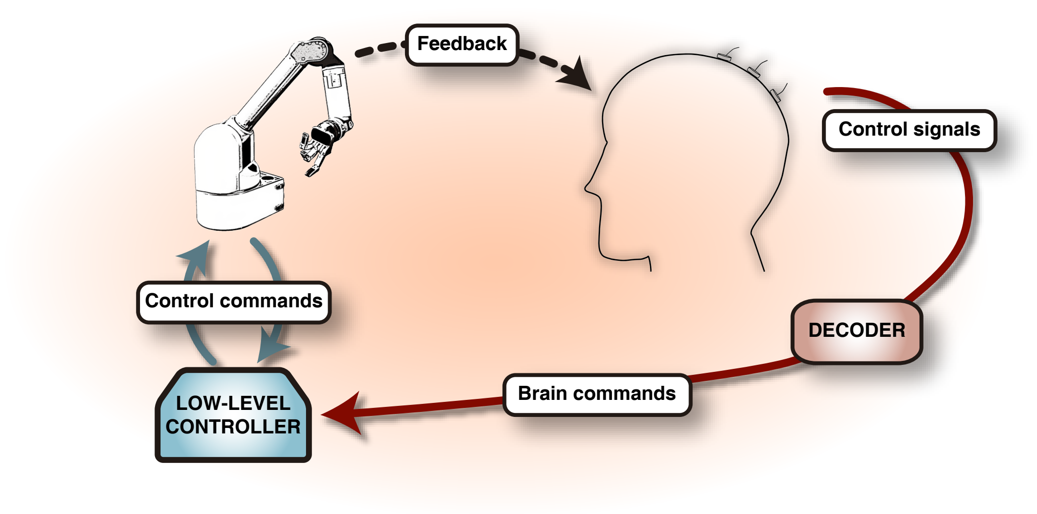


Figure 1.1: Usual ‘control BMI’ approach: the user delivers mental commands that specify the next state of the neuroprosthesis.

approach closely resembles normal motor control in that it links neural activity to motor behavior or to motor execution [24]. Motor control, however, is the result of the combined activity of three different levels: the cerebral cortex, the brainstem, and the spinal cord [24]. The lowest level corresponds to the spinal cord, which handles many elements of basic or stereotypical movements, mainly via rhythmic and oscillatory outputs commonly known as central pattern generators (CPGs) [25, 26]. Following the spinal cord, the brainstem is mainly responsible for the selection, enhancement or variation of the different patterns present on the lower levels [27]; finally, the cerebral cortex represents the highest level, playing a wide range of roles such as sensory-motor transformations, action understanding, and planning and executing of goal-directed and skilled motor tasks [24, 28, 29]. Therefore, the involvement of and communication between these three levels will vary together with the motor task being executed. In fact, many elements of skilled movements, from manipulation to walking, are mainly handled at the spinal cord level with cortical areas providing just goals and movement onset [24,26].

But, could a BMI also mimic this strategy? As mentioned before, some studies have shown the feasibility to decode such a kind of cognitive information associated to voluntary, goal-directed movements [6, 20–22, 30]. Nevertheless, this approach requires that the intelligent controller, emulating the roles of the lower levels of motor control, knows how to (*i*) learn optimal motor executions (i.e. optimal trajectories), and (*ii*) store and execute them as needed.

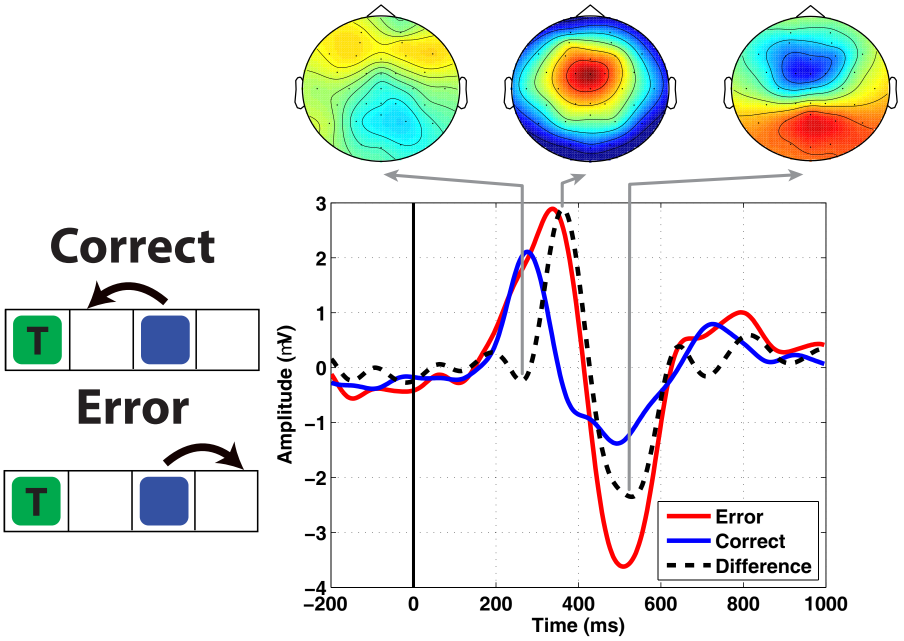


Figure 1.2: (Left) The device objective is to reach the target position (marked with T), and executes either correct or wrong actions while the user assesses them. (Right) Averaged error-related potential generated from the user's assessments, together with their topographic scalp interpolation (red is positive, blue is negative). The difference average (error minus correct averages) are used to study these potentials to remove those effects common to both potentials. This difference average is composed of three main components: a negativity at around 250 ms, and two broader positive and negative peaks at 350 ms and 500 ms respectively.

The main objective of this thesis is to propose and develop an alternative BMI paradigm to neuroprosthetics that, instead of decoding and executing low level control commands, makes use of higher-level cognitive information related to the task being performed. This way, the intelligent controller of the device could use this data as a way of dealing with the lack of information from subcortical levels. In other words, the paradigm tries to emulate the role of subcortical levels during motor control. However, entirely emulating this role would require extracting a large amount of cognitive signals, each one dealing with a specific piece of information about the task. In this thesis, we present the first step towards developing this paradigm by focusing on a specific cognitive signal related to the user's error processing. Specifically, this thesis studies human brain signals that carry cognitive information associated to decision-making processes that arise during goal-directed movements, and whose low-level implementation is handled at the subcortical and spinal cord level. In this approach, the neuroprosthesis is actively performing a task, while the user simply monitors the performance of the neuroprosthesis by assessing the quality of its actions. These assessments are extracted from the user's brain, and translated into feedback signals that help the intelligent controller of the neuroprosthesis how to solve the task. Thus, the development of this new paradigm was guided by its two main modules: the extraction and decoding of error signals from the user's brain, and the use of this information as input for an intelligent controller associated to the device.

Regarding the cognitive signals used as feedback for the paradigm, this thesis focuses on EEG signals related to the error processing system of the brain. In the field of cognitive neuroscience, these event-related potentials (ERP) are usually defined as a negative voltage deflection elicited in the user's brain around 200 ms after noticing his/her expected outcome differed from the actual outcome. Several works have associated them to the dopaminergic neural system [31–35], which is thought to be richly projected into the anterior cingulate cortex (ACC) region of the brain [36]. This cortex region would be in charge of conveying reinforcement learning signals (i.e. rewards) about the ongoing events [31]. In fact, many works have linked the dopamine, reinforcement learning and error processing as a single neural framework [31, 32, 35, 37–41].

A large number of works have demonstrated the existence of these signals in very different situations, always by averaging across hundreds of trials to increase the EEG signal-to-noise ratio: when a subject performs a choice reaction task under time pressure and realizes that he/she has committed an error [42] (the so-called error-related negativity –ERN or N_e–, usually followed by a sharp positivity, P_e); after a user is given feedback about a performed task [43] (the feedback-related negativity, FRN); and when the subject perceives an error committed by another person (observation ERN) [44]. A few years ago, the field of BMI became interested in these signals [45] finding that a variant of them, the error-related potentials (ErrP), were also present when the user delivered an order to the machine and the machine executed another one [46] (interaction ErrP); or when a user observed a device committing an error [47]. In spite of resembling a similar pattern to the ERN and FRN potentials, the error-related potentials are always followed by a sharp positivity (probably the P_e [42]) and by a second negativity, which could be related to a visual semantic mismatch [48] (see Figure 1.2).

Despite the existence on average of these signals in controlled situations, detecting them on single trial poses a harder problem, due to the non-stationary behavior of the EEG and its poor signal-to-noise ratio. Furthermore, the event-related potentials have additional sources of variability that can affect the amplitude or the latency of their components, such as the arousal [49], the probability of occurrence of the expected stimulus [50] or the stimulus evaluation time (i.e., the amount of time required to perceive and categorize a stimulus) [50, 51]. To solve this problem, the BCIs need a calibration phase to train a classifier prior to the device control. Recently, several works have successfully decoded these signals online, and used them to adapt the BMI classifier [52–54] or to prevent the execution of missclassified commands [55–60].

In this thesis, once these signals are decoded thanks to the calibration phase, they can be used as feedback for an intelligent controller. Given that these signals are allegedly related to a human reinforcement learning model, the field of reinforcement learning (RL) [61] provides a natural decision for the development of this module. In reinforcement learning, the goal of the device is to learn a policy mapping from situations to actions that maximize the expected reward [62]. Since its beginnings over twenty years ago, many approaches have been proposed and demonstrated over a wide range of applications, either in discrete spaces [61], continuous spaces under virtual scenarios [63–66], and for learning complex tasks in robotics [67, 68]. Throughout this thesis, the reinforcement learning module played an important role for the development of this paradigm, closing the BMI loop during the device control. However, for this first attempt of the proposed BMI paradigm, the closed loop worked in a different way: rather than sending commands and receving feedback from the device as in the control paradigm, the proposed system sends feedback in both directions (user to device, and device to user), allowing for a constant co-adaptation between the two. Once the paradigm is defined in a general context, it presents the first challenge to resolve: it is necessary to prove the advantages of the paradigm in realistic scenarios, such as the real-time use of a neuroprosthetic device. In this thesis, we show how can be achieve this objective.

A second challenge comes from the fact that, under this paradigm, the device needs a certain amount of time to learn the task being executed. Despite it would be desirable to have a system that is able to learn from the beginning of the experiment, in practice it is also needed for each task a calibration phase to train and detect online the EEG signals used, a time-consuming operation that adds to the task learning time and hinders the deployment of BMIs out of the lab. In the specific case of our paradigm, it would be thus convenient to train classifiers able to work irrespectively of the learning task being performed. Many works have tried to reduce this calibration time in several ways: Adapting the classifier with supervised techniques incorporating labeled examples of subsequent sessions [69, 70] or with unsupervised techniques [71, 72]; initializing the classifier model with data from a pool of subjects [73, 74]; or finding time-invariant features of the EEG to control the BCI [75, 76]. However, it is still unclear whether (and how) the error-related potentials vary across different tasks, and how to create classifiers able to generalize among different tasks. In this thesis, we study transfer learning across different tasks as a way of reducing the calibration time.

The third important challenge of the proposed paradigm is related to scalability. Firstly, it is impractical to learn the behavior each time a task has to be performed. Therefore, once optimal motor behaviors are learned, it is necessary to store and execute them on demand similarly to human motor control. In this way, learned behaviors are reused and can be part of more complex behaviors. Indeed, learning more complex behaviors also pose several problems for the proposed paradigm since BMIs have in general low information transfer rates. As the learning time grows exponentially as the task space increases, learning the optimal motor behavior via RL using brain signals may require too much time and effort from the user. Instead, it would be useful to pre-compute optimal motor behaviors rather than learning them from the user's feedback. This pre-computation can be done by exploiting the learning task structure (e.g. using external sensors to analyze the environment and localize tasks/objects of interest). The policies can be computed from scratch or adapted from previously learned policies taking into account the current context. In this thesis, this concept is illustrated using a reaching task with a discrete number of possible final positions. The corresponding optimal policies are computed offline, assuming that an optimal policy is the one that reaches the target following the shortest path. Similar strategies have been exploited by the so-called shared-control systems, where the device does not only execute the decoded commands, but is also involved in performing the task [2] (e.g. by taking into account the environment while reaching a target or avoiding an obstacle [9, 10]). The thesis shows how it is possible to store a collection of motor behaviors, and choose among them via the error potentials the one that most closely resembles the user expectations. In other words, the user can exploit previously learned policies to select which one has to be executed. An alternative interpretation is that by reducing the task space to a set of predefined policies, the time needed to learn and execute the task is drastically reduced (from tens of minutes to even half a minute) and, more importantly, it improves the scalability of the system to larger and more complex tasks.

Finally, given a sufficiently informative task structure, it could be possible not only to learn the task, but also to learn online and unsupervisedly the classifier while the task is being executed [74, 77]. This way we will show how, using all the lessons learned throughout the thesis, we can have a plug-

and-play BMI that follows the described paradigm, learns the task and the classifier and finally reaches the desired position by the user.

1.1 Structure and publications

The contents of the thesis are organized as follows.

Chapter 2 focuses on the first objective of the thesis: the development of an alternative paradigm for BMIs based on reinforcement learning and brain-decoded reward signals. Specifically, this chapter demonstrates how it is possible to learn optimal motor executions using users' brain feedback. This chapter was presented in [78–81].

Chapter 3 performs a deep analysis of the brain signals used for the paradigm: the error-related potentials. In this chapter, we demonstrate how the error-related potentials vary from one experimental protocol to another, and how this impedes the classifier generalization among different tasks. This work has been presented in [82, 83].

Chapter 4 shows how to reduce the calibration time of a given ERP experimental protocol by compensating the ERP latency variations across experiments so as to re-use data from previous experiments. This analysis was performed in two different ERPs: the error-related potentials and the P300 potentials [84]; and presented in [85, 86].

Chapter 5 extends and improves the paradigm presented in chapter 2 exploiting the task structure by choosing among a set of optimal motor behaviors given the user feedback. These findings were presented in [87–89].

Chapter 6 presents a novel way of removing the calibration phase for the proposed paradigm by estimating at the same time the optimal motor behaviors and the EEG meanings, also exploiting the task structure. This work is under review [90].

Finally, **chapter 7** presents the conclusions of this thesis and summarizes some points of future work.

2 | Reinforcement learning using brain signals

2.1 Introduction

In this chapter we introduce and demonstrate the alternative BMI paradigm proposed in the introduction (see Figure 2.1). This paradigm is based on human brain signals that carry cognitive information associated to decision-making processes that arise during goal-directed movements whose low-level implementation is handled at the subcortical and spinal cord level. One of such brain cognitive signals is the error-related potential (ErrP), a time-locked potential elicited from wrong decisions. Such ErrP is not only observed in the grand averages of human electroencephalogram (EEG) signals, but can also be detected in single trials [46, 47, 56, 79, 91].

This BMI paradigm can exploit ErrP to teach an intelligent neuroprosthesis how to acquire a correct control policy to achieve user's desired goals through reinforcement learning (RL) [61]. Figure 2.1B illustrates this 'teaching BMI' paradigm. The user simply monitors the performance of the neuroprosthesis, assessing the quality of its actions. ErrP develops in subjects' EEG whenever the neuroprosthesis executes an action that does not match their expectations. This ErrP is decoded online and acts as a reward signal for a reinforcement learning algorithm that trains the neuroprosthesis controller.

Here we demonstrate this teaching BMI paradigm in a series of three closed-loop experiments of increasing complexity involving twelve subjects. These experiments range from controlling the movements of a 1D cursor, to a simulated robot, and, finally, a real arm robot —both robots working in a 2D space. In all cases, the subject monitors the performance of an initially random controller while the online decoding of ErrP enables the RL algorithm to rapidly acquire (quasi) optimal policies to reach desired targets (see Fig. 2.1C). Furthermore, the simulated or real brain-controlled device is operational since the very first trial, improving performance steadily over time and always reaching the desired target. Also, experiments with the real arm robot show that initial control policies learned for a training target can be easily re-acquired for other targets.

These experiments illustrate a number of appealing properties associated to the use of ErrP as a brain signal to operate a BMI. First, like for other event-related potentials, user's training time is minimal —a calibration session is enough to model the user's ErrP decoder. Second, this paradigm makes it possible to achieve tasks in a user-specific manner —the learned control policy depends on the individual user's assessment. Third, single trial decoding of ErrP does not need to be perfect to maneuver a neuroprosthesis —it suffices that the ErrP decoder performs statistically above random to learn the control policy. Finally, and perhaps more importantly, ErrP is rather independent of the task (e.g., target or action type) —making control of neuroprostheses scalable to more complex tasks since the learning burden is on the robot side.

2.2 Methods

2.2.1 Subjects and data recording

Twelve volunteers (four females) aged between 23 and 34 years old participated in each of the three experiments. EEG signals were recorded using a gTec system with 16 electrodes. The electrodes were located at Fz, FC3, FC1, FCz, FC2, FC4, C3, C1, Cz, C2, C4, CP3, CP1, CPz, CP2, and CP4 according to the 10/10 international system. The ground was placed on the forehead (AFz) and the reference on the left earlobe. EEG was digitized at 256 Hz, power-line notch filtered at 50 Hz, and band-pass filtered at [1,10] Hz.

2.2.2 Experimental setup

Each experiment lasted $\sim$ 2.5 hours and was recorded in a different day. The time elapsed between two consecutive experiments was 17.58 ± 10.09 days. In all experiments, subjects were instructed to monitor the device while it tried to reach a target (only known by subject) and to assess whether the device actions were correct or incorrect. They were also asked to restrict eye movements and blink at specific resting periods. Each experiment was divided into two phases: training and reaching. Each phase was composed of several runs, each run consisting of 100 actions performed by the device.

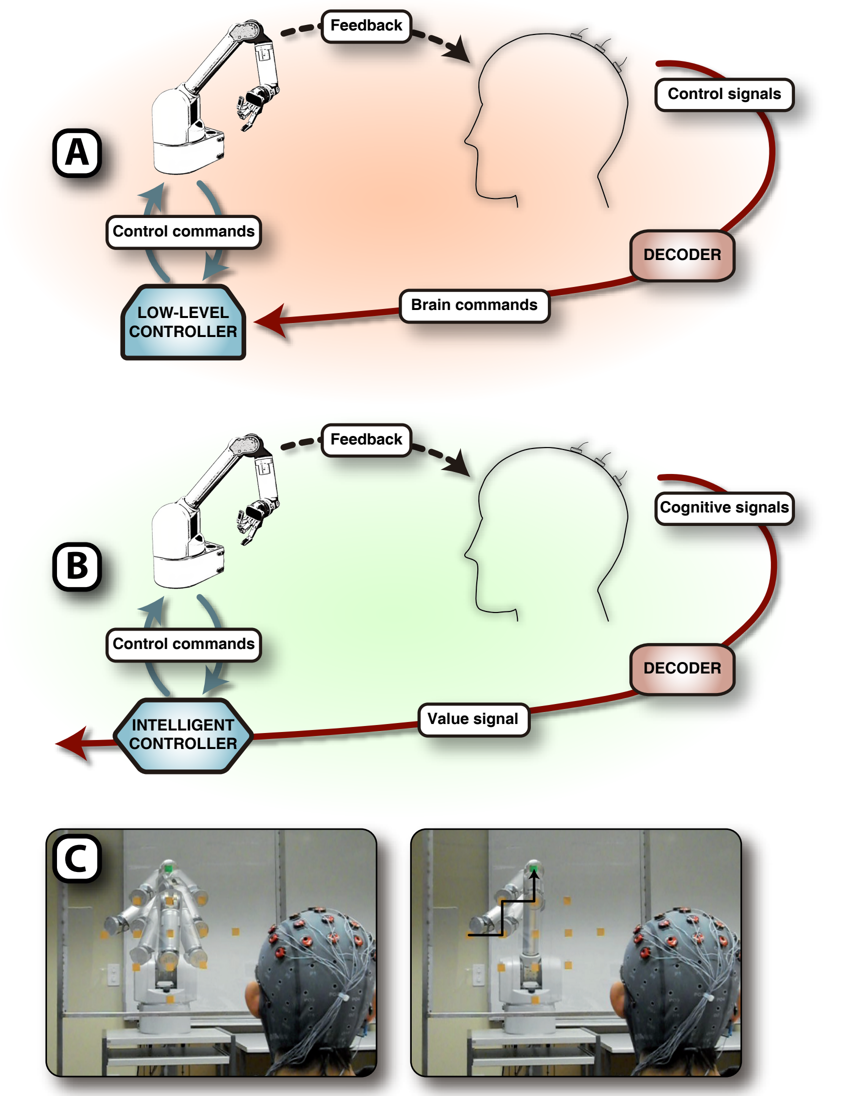


Figure 2.1: *[A]* Usual ‘control BMI’ approach: the user delivers mental commands that specify the next state of the neuroprosthesis. *[B]* The novel ‘teaching BMI’ paradigm: the user assesses the actions performed by the neuroprosthetic device as good or bad. This assessment is decoded from the user’s brain signals, and employed as a reward for a reinforcement learning algorithm embedded in the neuroprosthesis controller. *[C]* Demonstration of the ‘teaching BMI’ paradigm with a real arm robot during a reaching task. *[C, Left]* Initially, the robot explores the environment by performing random actions, while it receives feedback from the user’s brain signals. *[C, Right]* After a short number of exploratory actions, the robot learns how to reach the target through reinforcement learning using the ErrP as a reward signal. See Supplementary Information for a video showing the whole process.

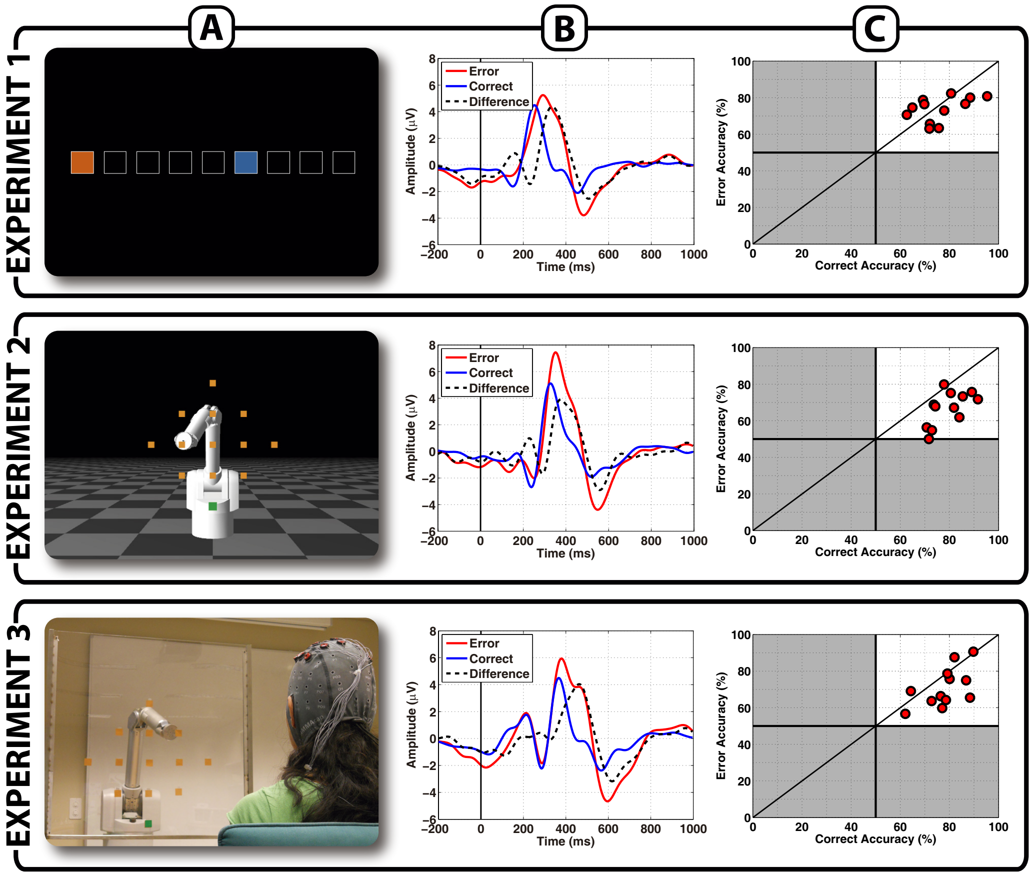


Figure 2.2: *[A]* Experimental setup. In experiment 1, the device (blue square) can perform two possible actions (move one position to the left or to the right) across 9 possible states (positions) to reach a target (red square) located either at the left- or right-most state. In experiments 2 and 3, the robot moves across 13 states (orange squares). At each state, it can perform four actions (move one position to the left, right, up, or down) to reach a target (green square) located at the left-, right-, up- or down-most state. *[B]* Grand average potentials (error, correct and error minus correct) at channel FCz, over all subjects ($N = 12$). $t = 0$ ms is the moment when the device starts the action. *[C]* ErrP online classification accuracy. The x-axis and y-axis represent the correct and error accuracies, respectively. Each dot corresponds to a single run performed by each subject.

During each run the target location remained fixed and, whenever the device reached that location, its position was randomly reset to a location at least two positions away from the target. The target location was randomly chosen between runs.

The objective of the training phase was to build a classifier able to detect the error potentials. In the initial runs, the device performed erroneous actions with a fixed probability (20%). After each run, all the collected data was used to train the ErrP classifier. Once the decoding accuracy was above 80%, or four runs were performed, an additional run was executed where the output of the classifier was used to adapt the device controller using RL (see below). Thus, in this RL run the error probability was variable. If the accuracy in this RL run was below random, the classifier was retrained with additional RL runs until the criterion was reached. All subjects needed one RL run of this additional training for at least one experiment, and a maximum of 3 RL runs were needed for one subject in one experiment. The duration of the entire training phase was always shorter than 30 minutes.

In the reaching phase we assessed the performance of the proposed paradigm. In this phase, the information decoded from the EEG (indicating whether the subject considered the action as correct or erroneous) was used as a reward signal to learn the state-action policy through RL. One run was performed per target location (2 runs in the case of experiment 1, and 4 runs for experiments 2 and 3). In the last two experiments we also tested the generalization capabilities of the proposed approach by including target locations that were not used in the training phase. In all RL runs, the device controller was initialized to a random state-action policy. In the reaching task we also computed the learning rate of the control policy, calculated as the number of correctly learned actions across the number of actions in each run. At each time, only those states already visited were considered in the estimation.

Experiment 1: Virtual Moving Square [47] (Fig. 2.2A, Top). Participants faced a computer screen showing a horizontal grid with nine different positions (states), including one blue moving square (device), and one red square (target position). The blue square could execute two actions: move one position to the left or to the right. When the device was at the boundaries (i.e., at the left- or right-most states), actions that moved the square out of the state space were not allowed. The time between two consecutive actions was random and within the range [1.7,3.0] s. During the whole experiment, the target was either the left-most or right-most state.

Experiment 2: Simulated Robotic Arm (Fig. 2.2A, Middle). Subjects faced a computer screen displaying a virtual robot (device). We simulated a Barrett whole arm manipulator (WAM) with 7 degrees of freedom using the RobotToolkit framework (http://lasa.epfl.ch/RobotToolKit). The robot could place its end-effector at 13 different positions represented by orange squares (states), with one position in green (target). It could perform four actions: moving one position to the left, right, up, or down. As before, when the device was at a boundary state, actions that moved the robot out of the state space were not allowed. In contrast to the first experiment, the robot's movements between two states were continuous, lasting $\sim$ 500 ms. The time between two consecutive actions was random within the range [2.5,4.0] s. For the training phase, the targets were the up- and down-most positions. For the reaching phase, the up-, down-, left-, and right-most positions were tested as targets.

Experiment 3: Real Robotic Arm (Fig. 2.2A, Bottom). This experiment followed the same design as experiment 2 but involving a real robotic arm (Barret WAM). The robot was two meters away from the user and was pointing at states in a Plexiglas transparent panel between the two. The distance between two neighbor states in the panel was 15 cm.

2.2.3 Event-related potentials

The EEG waveform, amplitude and latency of the time-locked grand average potentials for both conditions (erroneous and correct actions performed by the device) were characterized using the signals from the training runs where the probability of errors was constant. Statistical differences were evaluated using a 3 (frontal, central or centro-parietal electrode locations) x 3 (left, midline or right locations) x 3 (experiments) within-subjects ANOVA on the peak amplitudes and latencies of the difference ERP for all electrodes (see Supplementary Information).

Although the experimental protocols may induce eye movements, these were not likely correlated to the erroneous or correct actions. Indeed, EEG trials for each condition (error and correct) include movements in all possible directions, thus reducing the possibility that the classifier accuracy was influenced by saccadic movements. See Supplementary Information for a more complete analysis supporting that ocular artifacts do not affect ErrP decoding.

2.2.4 ErrP classifier

The features for the classifier were obtained from a subject-specific spatio-temporal filter [80]. For each training trial, a common-average-reference filter was applied. Then, eight fronto-central channels (Fz, FCz, Cz, CPz, FC1, FC2, C1, and C2) within a time window of [200,800] ms were downsampled to 64 Hz and concatenated to form a vector x of 312 features per trial. These vectors were normalized and decorrelated using principal component analysis (PCA). A feature selection process based on the r^2 score then retained the f-most discriminant features, using a five-ten-fold cross validation. The f features of all labeled trials were used to train a linear classifier (linear discriminant analysis, LDA), which mapped the input feature vector x^f of a given trial into a binary output $y \in$ {-1,+1}. On average, 36 features were selected across all subjects (± 13).

We report the online single-trial accuracies during the reaching task. To assess the statistical significance of the obtained accuracies, we compute the chance levels according to the available number of trials using the binomial cumulative distribution [30]. The estimated chance levels at $\alpha = 0.05$ were 56% for Experiment 1 and 54% for Experiments 2 and 3.

2.2.5 Reinforcement Learning with ErrPs

The RL strategy [61] was modeled by a Markov decision process, denoted by the tuple $\{S, A, r, \gamma\}$ with S the state space (the possible positions of the device), and A the action space (the possible actions of the device). The reward function $r : S \times A \rightarrow \mathcal{R}$ represented the goodness of the executed action at a given state. The goal of RL was to obtain a policy $\pi : S \rightarrow A$ mapping the state space into the action space (i.e., which action had to be performed at each state) so as to maximize the expected return $R = \sum_{k=0}^{\infty} \gamma^k r_{k+1}$ at time k. The RL implementation was the Q-learning iterative algorithm [61]:

$$\begin{aligned} Q_{k+1}(s_k, a_k) = &Q_k(s_k, a_k) + \\ &\alpha\Big[r_{k+1}(s_k, a_k) + \gamma \max_{a' \in A} Q_k(s_{k+1}, a') - Q_k(s_k, a_k)\Big] \end{aligned}$$

where k is the current step, γ is a discount factor, and α is the learning rate (for the designed experiments, γ and α were set empirically to 0.4 and 0.1, respectively). During the iterative process at time k, the device executed an action a_k from state s_k to state s_{k+1}, receiving a reward $r_{k+1}(s_k, a_k)$. This reward was used to update the reinforcement learning policy after each action. All the Q-values were set to zero at the beginning of each run ($k = 0$). At the end of the run, the final policy π was computed as the policy that, at each state s, always followed the action a' with the maximum Q-value, $\pi = \arg\max_{a' \in A} Q^{\pi}(s, a')$.

An ε-greedy policy was used to select the next action a_k to be executed at each step k of the iterative process. This policy selected the action with highest Q-value (best action) for (100$-\varepsilon$)% of the times in the state s_k, while a random action was selected the remaining times. The experiments started with a completely exploratory behavior (ε =100%), and every time an exploratory action was chosen ε was decreased by a small factor (5%) until reaching a minimum value (20%) to always maintain a small percentage of exploration.

The output of the ErrP classifier at each step k was used as the reward function r_{k+1}: after an action was executed, the reward was -1 or +1 depending on whether or not an ErrP was detected.

2.3 Results

Figure 2.2A summarizes the three experiments executed by 12 subjects. Subjects' task was to monitor the performance of a neuroprosthetic device that had to reach a target. This target was known by the subject, but not by the neuroprosthesis. Online operation of the closed-loop system was tested on several runs where the goal is to repeatedly reach a fixed target location from different starting points. At the beginning of each run the device controller was initialized to a random behavior (i.e., equiprobable actions for all states). After each action the controller was updated by means of RL based on the online decoding of the ErrPs. Each of the runs had a different target location which remained constant during its entire length (100 device actions). Whenever the device reached the target, it was randomly reset to a new location. On average (for all runs and targets), subjects reached the target 12, 13 and 13 times for experiments 1, 2 and 3, respectively.

2.3.1 Analysis of error-related potentials

We analyzed variations in the grand average EEG potentials for both conditions (see Fig. 2B and Fig. 2.3). To this end we performed a statistical analysis on the difference ERP (error minus correct condition) for all electrodes in the time window [-200, 1000] ms, t =0 ms being the instant when the device starts to move. Only signals from the training runs —having a constant error-rate (20%)— are used in this analysis. These runs yielded between 200 and 400 trials for each subject.

A 3 (brain area: frontal, central or centro-parietal electrode locations) x 3 (left, midline or right locations) x 3 (experiments) within-subjects ANOVA was performed on the peak amplitudes and latencies of the difference average [50]. Each group is summarized in Table 2.1. When needed, the Geisser-Greenhouse correction was applied to assure sphericity. Pairwise post-hoc

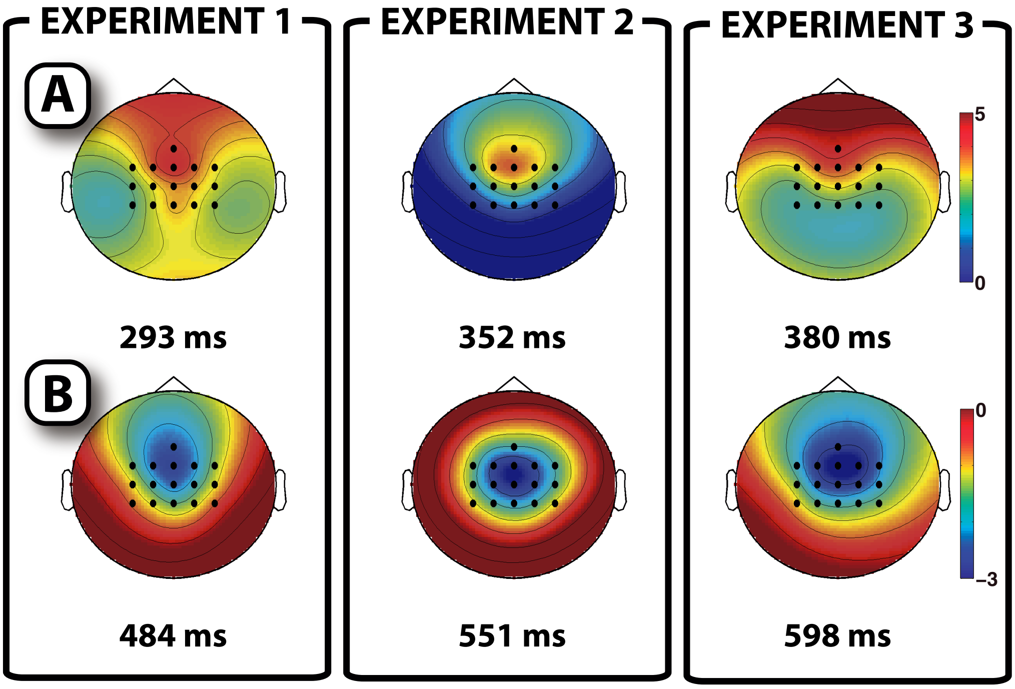


Figure 2.3: Topographic interpolation of the most prominent (A) positive peak and (B) negative peak of the difference average for each experiment, together with the time point of each peak (in milliseconds).

tests with the Bonferroni correction were computed to determine the differences between pairs of experiments.

Table 2.1: Electrode locations used as factors for the ERP statistical analysis. Each row corresponds to the brain areas, while columns correspond to the laterality.

	Left	**Midline**	**Right**
Frontal	FC3, FC1	Fz, FCz	FC2, FC4
Central	C3, C1	Cz	C2, C4
Centro-parietal	CP3, CP1	CPz	CP2, CP4

We mainly found significant effects on the latency but not the amplitude of the difference potential. The type of experiment significantly affected the latencies of both the positive ($F_{2,22} = 41.594, p = 3 \times 10^{-8}$) and negative peaks of the difference ERP ($F_{2,22} = 7.522, p = 0.003$). The brain area also affected the latencies of the positive peak ($F_{1.32,14.55} = 14.175, p = 0.001$) but not the negative one $F_{2,22} = 0.911, p = 0.417$). Similarly, the hemisphere affected the latency of the positive peak ($F_{2,22} = 5.279, p = 0.013$), but not the latency of the negative one ($F_{2,22} = 1.711, p = 0.204$). No significant interactions were found.

For the positive peak latencies, post-hoc pairwise tests revealed significant differences between experiments 1 and 2 ($p = 0.0001$), and between experiments 1 and 3 ($p = 0.0001$), but not between experiments 2 and 3 ($p = 0.068$). For the negative peak latencies, there were significant differences between experiments 1 and 3 ($p = 0.009$), but not between experiments 1 and 2 ($p = 0.357$) nor experiments 2 and 3 ($p = 0.122$).

In contrast, the amplitude of the positive and negative peaks were not significantly affected by the experiment ($F_{2,22} = 0.124, p = 0.884$ and $F_{2,22} = 2.304, p = 0.123$, respectively) nor the brain area ($F_{1.19,13.08} = 1.227, p = 0.737$ and $F_{1.32,14.47} = 0.071, p = 0.857$). The laterality significantly affected the positive peak amplitude ($F_{2,22} = 4.556, p = 0.022$), but not the negative peak ($F_{2,22} = 3.425, p = 0.051$). As in the case of the peak latencies, no significant interactions were found.

2.3.2 Analysis of ocular artifacts

We assessed the possibility of EEG signal contamination by movement-related saccades. In this study, we compute the grand average ERPs (correct and

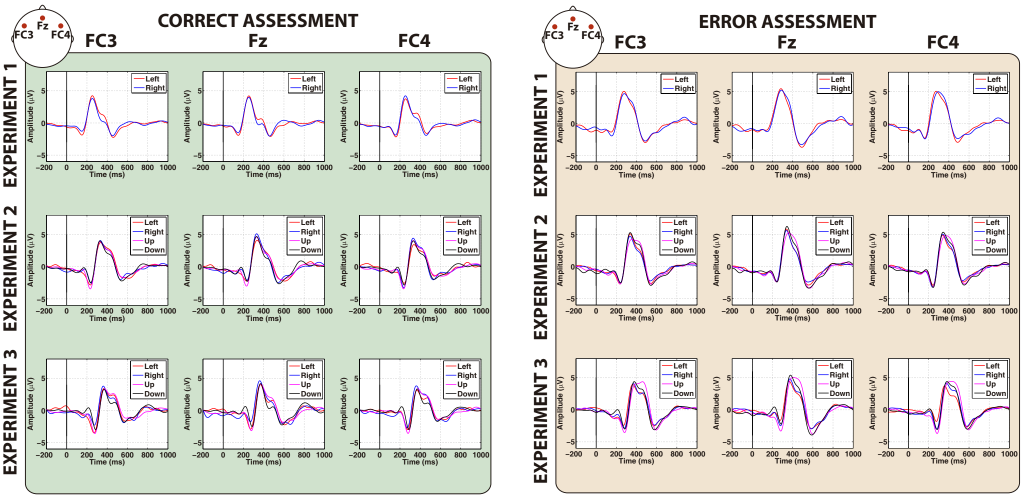


Figure 2.4: Grand averages in channels FC3, Fz, and FC4 for correct and error assessments (left and right respectively) of each movement direction (left, right, up or down), averaged for all subjects ($N = 12$). Rows correspond to each experiment. The results show that the averaged signals of the same assessment (either correct or error) are very similar across all directions, reducing the possibility of having a systematic influence on the ErrP classification process.

error) of all channels separately for each different action (moving left, right, up, or down). No substantial differences were found among these ERPs, suggesting little influence of eye movements. This is illustrated in Figure 2.4 that show the averages of the three most frontal electrodes: FC3, Fz and FC4. As can be seen, the differences among assessments (correct or error) were larger than the differences among directions (left, right, up or down). This is consistent with previous studies that found no influence of this type for experiment 1 [46, 47].

To evaluate the existence of statistical differences due to both assessments and movement directions, we performed 2 (factor assessment: error or correct) x 4 (factor movement direction: left, right, up or down) within-subjects ANOVAs on the values of the most prominent positive and negative peak amplitudes of the grand averages (note that for experiment 1 the ANOVA was 2 x 2 since there were only two possible movement directions). When needed, the Geisser-Greenhouse correction was applied to assure sphericity. The assessment and direction main effects and the assessment x direction interaction were studied.

Regarding the main effects, statistical differences were found for the assessment for all the experiments, for the positive ($F_{1,11} = 17.277, p = 0.002$, $F_{1,11} = 15.567, p = 0.002$, and $F_{1,11} = 14.202, p = 0.003$ for experiments 1 to 3) and negative ($F_{1,11} = 10.087, p = 0.009$, $F_{1,11} = 14.658, p = 0.003$, and $F_{1,11} = 11.581, p = 0.006$) peaks. On the contrary, no significant differences were found for the direction main effect ($p > 0.1$). Regarding the assessment x direction interaction, significant differences were found during experiment 2 ($F_{3,33} = 3.721, p = 0.02$ and $F_{3,33} = 3.903, p = 0.02$ for the positive and negative peak); and during experiment 3 for the negative peak ($F_{3,33} = 3.461, p = 0.03$) but not for the rest of the cases ($p > 0.35$). These results indicated that the largest differences on the potentials were due to the different assessments (error / correct), whereas the movement directions of the device affected less the potentials.

To further discard the influence of artifacts on the assessments, data used to train the classifier included all possible movements for each class, thus reducing the possibility that classification was biased by their directions. For instance, during experiment 1, both targets are used for the classifier training, thus the error and correct assessments are not likely to be correlated with left or right eye movements. Moreover, results obtained in the generalization test for experiments 2 and 3 further support the fact that classification depends on the movement evaluation and not on its direction. Indeed, the training set contained samples where the target locations were Up and Down, while the BMI was tested on targets Left and Right. Finally, to test whether the trained classifiers discriminated different directions rather than assessments, we computed for each subject the accuracy of decoding the different pairs of movement directions (e.g. left versus right, up versus left, ...) from a fixed assessment (either correct or erroneous) with the same features and classifier used during the experiments. The mean accuracies obtained were of 52.16±5.22, 50.07±5.07, and 49.48±6.41 for experiments 1 to 3, and thus did not reach the chance levels of 56%, 54% and 54% (see 'ErrP classifier' in Materials and Methods), indicating that the classifier was not trained to distinguish saccadic artifacts, but user's assessments.

2.3.3 Event-related potentials and single-trial classification

Figure 2.2B shows the grand average potential on electrode FCz elicited by the user's assessment of both types of movements (correct and erroneous), as well as the difference ERP (error minus correct). As reported by previous studies [47, 78], the difference potential in the three experiments is mainly composed of two prominent positive and negative peaks at around 300 and 500 ms, respectively.

A within-subject statistical analysis shows significant effects of the experiments on the latency but not the amplitude of the difference ERPs (see Supplementary Information). Overall, this suggests that these signals reflect a common phenomenon (i.e., error-processing) across the three experiments, where the complexity of the experimental protocol mainly affects the temporal characteristics of the brain response.

Online single-trial decoding of the ErrPs (error vs. correct) is shown in Fig. 2.2C. Classifier accuracy was comparable for all experiments. Performance was above chance level (except for one subject in experiment 2) —a necessary condition when using stochastic rewards for reinforcement learning [61]. This demonstrates the feasibility of extracting useful reward-related signals from human brain activity.

2.3.4 Reaching task: Learning to reach the training targets

Figure 2.5A summarizes the reaching performance when the closed-loop system is tested on the same target locations used for training the ErrP classifier. It shows the optimal action per state (arrow) and the number of subjects that successfully learned those actions (dark red corresponds to all subjects). In experiment 1, the optimal actions were learned in 94% of the cases indicating that the users were able to teach the device a quasi-optimal control strategy.

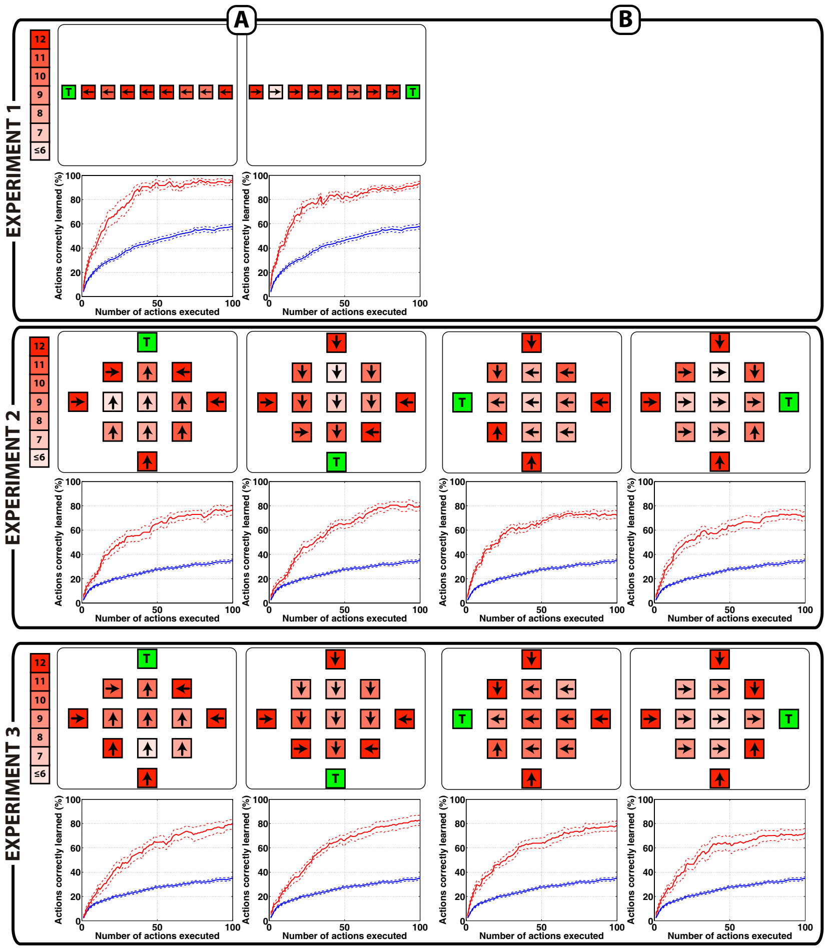


Figure 2.5: Online reinforcement learning. *[A]* Reaching the same targets used to train the ErrP classifier (Left and Right locations for experiment 1, and Up and Down for experiments 2 and 3). *[B]* Reaching new targets (Left and Right for experiments 2 and 3). For each experiment the top rows in [A] and [B] show, for each state, the optimal action (arrows), and the number of subjects that correctly learned that action (color coded, dark red corresponds to all subjects). Target locations are denoted by green squares. The bottom row shows, in red traces, the percentage of correctly performed actions (mean±SEM, straight±dashed lines, $N = 12$ subjects) as a function of the number of performed actions within a run (100 actions, starting from a random device controller). Blue traces correspond to chance performance level calculated using random rewards (100 repetitions). *[C]* Percentage of correctly performed actions (y-axis) versus the online ErrP decoding accuracy (x-axis) at the end of each run, for the training (red circles) and new targets (green squares). The straight lines show the best-fitting regression line for training (red) and new (green) targets, and each point in the plot corresponds to the mean accuracy of one subject.

In experiment 2, where a more complex state-action space has to be learned, the performance decreased. Nonetheless, 85% of the trajectory motion was learned. Interestingly, no major differences were found between the performance in experiments 2 and 3. The correct actions were learned for 87% of the states in experiment 3.

For all experiments and runs we observe a steady acquisition of a suitable control policy. As shown in Fig. 2.5A(bottom), the device quickly learns the correct action after visiting a state (see Methods). For experiment 1, the number of actions correctly learned was always significantly above chance level (one-tailed unpaired t-tests, $p < 0.05$). The number of actions correctly learned consistently increased as more actions were performed (correlation between time and actions learned of $r = 0.70, p < 1 \times 10^{-8}$). For experiment 2 and 3, the convergence was slower due to the higher number of states and actions. However, after 9 and 6 actions (for experiment 2 and 3, respectively) the results were significantly above the chance level (one-tailed unpaired t-tests, $p < 0.05$). In both experiments performance does not appear to have reached asymptotic values after 100 actions. This suggests that given more time, the device would have been able to learn the entire policy. There was also a high correlation between time and learned actions ($r = 0.79, p < 1 \times 10^{-8}$ for both experiments).

The number of actions correctly learned was significantly correlated with the mean performance of the ErrP classifier (Fig. 2.5C). In particular, for experiments 2 and 3 ($r = 0.75, p = 2{\times}10^{-5}$ and $r = 0.76, p = 1{\times}10^{-5}$, respectively). A lower correlation was found in experiment 1 ($r = 0.13, p = 0.54$), indicating that learning the 1D policy is less sensitive to misclassification of the error-related signal.

2.3.5 Reaching task: Learning to reach new targets

The system was also able to rapidly learn suitable policies to reach target locations not used during the training phase (Fig. 2.5B). For both experiments 2 and 3, no significant differences in the number of correctly performed actions were observed with respect to the training targets (paired t-test, $p = 0.26$ and $p = 0.65$, respectively). In experiments 2 and 3, 81% and 85% of the actions were learned, respectively. This shows that the ErrP does not depend on targets, as the ErrP classifier does not need to be retrained for unseen targets.

As before, the number of actions correctly learned across time for the new targets (Fig 2.5B) was above chance level after the first movements (one-tailed unpaired t-tests, $p < 0.05$), and increased along the entire run. Furthermore, the correlation between time and the number of correctly learned actions was also very high ($r = 0.70, p < 1 \times 10^{-8}$ for both experiments). When comparing the acquisition of the control policy across time between training and new targets, the learning rate was significantly equal for more than 19 and 8 actions for experiments 2 and 3, respectively (two-tailed unpaired t-tests, $p > 0.05$).

As before, the number of actions correctly learned across time for the new targets (Fig 2.5B) was above chance level after the first movements (one-tailed unpaired t-tests, $p < 0.05$), and increased along the entire run. Furthermore, the correlation between time and the number of correctly learned actions was also very high ($r = 0.74, p = 3 \times 10^{-5}$ and $r = 0.78, p = 8 \times 10^{-6}$ for experiments 2 and 3, respectively). When comparing the acquisition of the control policy across time between training and new targets, the learning rate was significantly equal for more than 19 and 8 actions for experiments 2 and 3, respectively (two-tailed unpaired t-tests, $p > 0.05$).

There was also a high correlation between the mean classifier accuracy and the number of actions correctly learned (Fig. 2.5C) ($r = 0.74, p < 1 \times 10^{-5}$ and $r = 0.78, p < 1 \times 10^{-5}$ for experiments 2 and 3, respectively), with no significant differences between the accuracies of training and new targets (paired t-test, $p = 0.90$ and $p = 0.85$ for experiments 2 and 3, respectively).

2.4 Discussion

This paper describes an alternative and complementary BMI paradigm to neuroprosthetics that decodes cognitive brain signals associated to decision-making processes relevant for achieving goals. This new approach is demonstrated with a BMI where the subject monitors the robot actions, and it decodes the brain correlate of the subject's assessment of those decisions (erroneous or correct). This cognitive signal is exploited by an RL algorithm to learn which is the sequence of actions that solves a task.

In the experiments reported here, it is assumed that the neuroprosthesis owner wishes to initiate a voluntary, goal-directed movement whose low-level execution is delegated to subcortical, spinal cord and musculoskeletal structures. In our case, this lower level of motor control is emulated by an intelligent controller. But, where do the movement goal and onset come from? This kind of information can be decoded from the subject's cortical activity [6,20–22]. As a result, the combination of all these sorts of cognitive brain signals would be sufficient for the operation of any neuroprosthesis, no matter its complexity and number of degrees of freedom. Indeed, as the experimental results reported in this paper illustrate, the burden of learning the (quasi) optimal trajectories is on the controller side rather than on the human.

A full integration and demonstration of this cognitive BMI approach, however, remains to be proven and has a high priority in the research agenda of neuroprosthetics. Besides, further research will uncover additional cognitive brain sigwnals that will enrich this initial basic set, thus enlarging the repertoire of decision-making processes available for natural, intuitive control of neuroprostheses to perform goal-directed movements. An immediate candidate is anticipatory signals [92] that indicate the subject is preparing to execute a new action at the arrival of relevant future events, which can be decoded at the single-trial level [93, 94].

2.4.1 Error-related potentials and reinforcement learning

ErrPs have been previously explored in BMI research. Recent work has shown how detection of ErrPs in response to the outcome of the BMI can improve the BMI performance by correcting the BMI output [56–58]. Additionally, several authors have discussed how ErrPs could be utilize to adapt the BMI decoder [46, 91]. In our preliminary works we explored ErrPs as a teaching signal in simple 1D virtual scenarios [47, 78]. This paper unifies and extends them under an RL framework. Furthermore, here we demonstrate how the BMI teaching paradigm scales to neuroprosthesis with a complexity similar to state-of-the-art BMIs.

Scalability is indeed a crucial property of the BMI teaching approach. The main reason is that, as the experimental results demonstrate, the ErrP is rather independent of the task (e.g., target and/or action) and reflects a common error processing mechanism in the brain [31,42,95]. Importantly, similar ErrPs waveforms can be observed on human EEG, ECoG and intracranial recordings [91, 96]. Such an ErrP can then be combined with a wide range of RL techniques that deal with large or continuous state-action spaces [61, 97] and that have demonstrated to acquire complex robotic skills [67, 98]. It is also worth noticing that in this BMI teaching paradigm the neuroprosthesis is operational as soon as the accuracy of the ErrP decoder is above chance level —which usually takes just minutes as reported in this paper— and keeps adapting indefinitely, as it is the case of human motor control.

Although we have simply combined ErrPs and RL algorithms, there seems to exist a physiological relationship between the two [31, 32]. In particular, the midbrain dopaminergic neurons encodes a reward prediction error that guides adaptive behavior in a way similar to RL [40, 99]. Access to this dopaminergic system could provide richer reward signals than the binary

ErrP signal presented in this paper. Similarly, one could exploit error-related components to measure the quantitative difference between the expected and the actual outcome [32] or to model habituation to the stimulus [50].

RL has also been incorporated in BMI to drive co-adaptation between the BMI and the robot [100]. In this case, the feedback provided to the BMI decoder was derived from external sensors that measured how well the robot performed a reaching task. This feedback, however, could also be the ErrP, the activity of dopaminergic neurons, or signals from other brain reward areas such as the nucleus accumbens. [101] reports a first attempt to exploit the latter in a BMI co-adaptive framework.

3 | Task-dependent signal variations in EEG error-related potentials

3.1 Introduction

As we have seen in the introduction, brain-computer interfaces always rely on the fact that the user performs a mental task, which presents associated brain patterns that are measured on the EEG and decoded to obtain the intention of the user (see Figure 1.1). Subsequently, these intentions are used to operate a device performing given operational tasks, which can be related or not to the decoded mental task. A major difficulty of BCIs is to deal with the non-stationary behaviour and noise of the EEG [102,103]. Most of efforts in this context have been devoted to those BCIs where the mental task and the operational task are not related [69–72, 75, 76]. On the other hand, several BCIs make use of mental tasks that are related to the operational task being performed. One such BCI is the paradigm presented in the previous chapter (see Figure 2.1), where the mental task (the error-related potential) is associated to the operational task (the device performing actions). On these kinds of BCIs, there could exist task-dependent signal variations hindering the transfer learning of classifiers among different operational tasks. As a result, the BCI paradigm presented in the previous chapter needed a calibration phase for each specific task performed (experiments 1 to 3, see Figure 2.2).

A great portion of the existing BCIs rely on mental tasks whose associated brain patterns are independent of operational tasks. For instance, many BCIs have used: (*i*) self-regulation of rhythms' amplitudes in the temporal domain (e.g., slow cortical potentials [104]); (*ii*) changes in frequency power spectrum (such as μ and β-rhythms [15], motor imagery of body limbs [105], or performance of cognitive mental tasks [5]); or (*iii*) attendance to visual stimuli (e.g., visual P300 potentials [84], or steady-state visual evoked potentials [106]). In the aforementioned BCIs, the mental task is decoupled from the operational task and the signal variations are assumed to be independent of the device operation [71]. Consequently, these BCIs generalize among different operational tasks. For instance, slow cortical potentials have been used for spelling devices or controlling 2D cursors, [104, 107]; motor-imagery BCIs have been used to operate 2D cursors, wheelchairs or mobile robots [10, 15, 105]; and P300 BCIs have been used to operate spelling devices, wheelchairs or mobile robots, among others [9, 84, 108]. However, even though the signal variations are independent of the operational task, the time-dependent non-stationarities of the EEG lead to changes in the features distributions and thus on the performance of the BCI [71]. The difficulty is to achieve a robust classification from the calibration phase to the feedback phase, along the feedback phase, and/or between different sessions. The techniques developed to date focus on the feature extraction or the classification process: either relying on finding time-invariant features of the EEG to control the BCI [75, 76]; or adapting the classifier with supervised techniques incorporating labeled examples of subsequent sessions [69, 70] or with unsupervised techniques [71, 72].

However, in other types of BCIs the mental task is coupled with the operational task, such as those based on the error-related potentials (see previous chapter). The principle of these BCIs is to detect in the brain patterns the occurrence of an error during the device operation. Although these BCIs are rather insensitive to time-dependent non-stationarities [47, 109], the operational task-dependent signal variations are natural in this context, as the monitoring of different devices or processes leads to different error-related potentials (e.g., user's own errors [42], interaction errors [46], feedback errors [32, 109], or observation errors committed either by another person [44] or by virtual or real devices [47, 78, 79], namely the observation error-related potentials). In previous studies, the latter potentials have been studied by their difference average, and characterized by three main ERP components (an N2, a P3 and an N4) [46]. These BCIs always require recalibration as, depending on the size of the variation of the error-related potentials, they might not generalize between small variations of the task or among different tasks. This is a large drawback for the practical deployment of these BCIs as calibration is a boring, tedious and tiring process that could last approximately 30-45 minutes before using the BCI [47, 78, 79].

This chapter studies the presence of signal variations in observation error-related potentials, where the mental task is coupled with the operational task. The analysis spanned the three design steps of BCIs: an electrophysiology study to characterize the existence of these variations; a feature analysis that showed how these variations affect the features distributions; and a classification analysis to measure the impact on the final BCI performance.

3.2 Methods

3.2.1 Data recording

The EEG signals were recorded with a gTec system (2 synchronized gUSBamp amplifiers) with 32 electrodes distributed according to an extended 10/20 international system (FP1, FP2, F7, F8, F3, F4, T7, T8, C3, C4, P7, P8, P3, P4, O1, O2, AF3, AF4, FC5, FC6, FC1, FC2, CP5, CP6, CP1, CP2, Fz, FCz, Cz, CPz, Pz and Oz), with the ground on FPz and the reference on the left earlobe. The EEG signals were digitized with a sampling frequency of 256Hz, power-line notch-filtered, common-average-reference (CAR) filtered and band-pass filtered at $[0.5, 10]$ Hz. The data acquisition was developed under the BCI2000 platform [110].

3.2.2 Experimental design

Ten subjects (eight males and two females, mean age 27.20 ± 4.08 years) participated in the experiments. The participants were comfortably seated one meter away of a computer screen displaying all the information related to the experiments. Two experimental protocols were designed. In each protocol the subject monitored the execution of a task where a virtual device had to reach a given goal. The motion of the device could be correct (towards the goal) or erroneous (away from the goal). Each task (denoted operational task, OT) consisted of a set of subtasks where the goal location changed but the device movements were fixed (see Figure 3.1). The subjects were asked to assess the device movements as erroneous or non-erroneous. The participants were allowed to freely move their eyes, while blinking and muscular movements were constrained to the resting periods. Each experiment took place in one session, and the time period between sessions was 3 ± 6 days.

Operational Task 1 (OT1, Figure 3.1a)

The screen displayed the virtual device as a blue circle in the centre of the screen (rest position), and also three rectangles indicating possible destinations, arranged in a triangle-like shape equidistant to the rest position. The goal was marked with a green rectangle. The device could perform three possible actions: move over one of the three rectangles. After the motion, the device returned to the rest position. The users were instructed to assess the actions as non-erroneous when it moved to the goal (green rectangle), or as erroneous otherwise. The three subtasks corresponded to the three possible goal locations (denoted subtasks OT1.Up, OT1.Left and OT1.Right). The

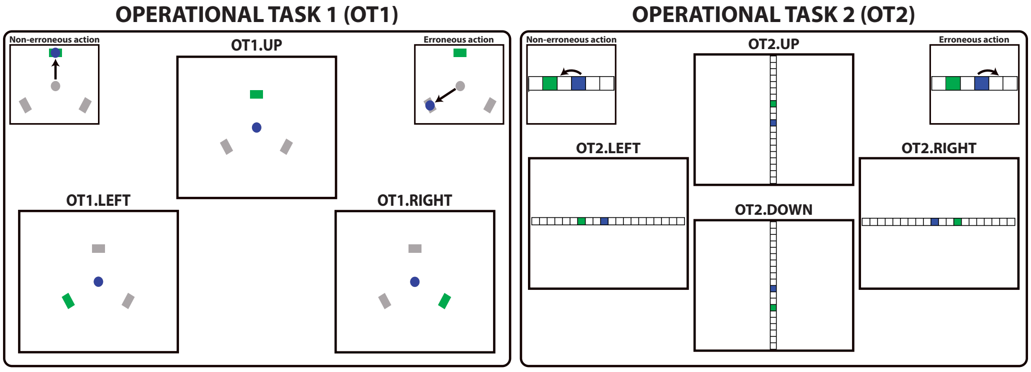


Figure 3.1: Schematic illustration of the operational tasks (OTs). Each operational task consisted of a virtual device (blue circle or blue square) that had to move to the goal location (marked in green). For each operational task, subtasks were defined by changing the goal location. Examples of non-error and error movements are shown for each task.

session was recorded in 36 runs. Each run consisted of 50 actions with the goal fixed at one position, with a 20% probability of not moving to the goal. For each subtask 120 error and 480 non-error potentials were acquired, i.e., a total of 360 errors and 1440 non-error potentials for the operational task. This session lasted 3 hours.

Operational Task 2 (OT2, Figure 3.1b)

The screen displayed the virtual device as a blue square and the goal location as a green square, located on a horizontal or vertical grid composed of 20 positions. The device could perform two possible actions: move one position left or right in the horizontal grid, or up or down in the vertical grid. The users were instructed to assess the actions as non-erroneous when the device moved towards the green square and as erroneous when it moved in the opposite direction. When the device reached the goal, the goal was moved three positions away along the grid (randomly left or right for the horizontal grid, or up or down for the vertical grid). The four subtasks corresponded to the relative position of the goal location with respect to the virtual device: OT2.Left, OT2.Right, OT2.Up and OT2.Down. The session was recorded in 24 runs. Each run consisted of 50 actions with a fixed grid (either the horizontal or the vertical grid), with a 20% probability of not moving towards the goal. For each subtask, 60 error and 240 non-error potentials were acquired, i.e., a total of 240 errors and 960 non-error potentials for the operational task. The session lasted 2 hours. The protocol was based on [46].

3.2.3 Analysis of error-related potentials and their task-dependent variations

The definition of the observation error-related potentials encompasses the appearance of three main and distinct components on the difference (error minus correct) average: an N2, a P3, and an N4 component [46]. Regarding the N2 and P3 components, several studies suggest that they may actually be the error-related negativity (ERN or N_e) and the following positivity (P_e) [42], but there is still an open discussion about it [111]. Regarding the N4 component, previous studies have suggested that its generation could be due to a visual semantic mismatch [48]. Nonetheless, for the observation error-related potentials, the three components are originated in the anterior cingulate cortex (ACC, Brodmann Areas 24 and 32) and the pre-supplementary motor area (pre-SMA, Brodmann area 6) [46], suggesting an activation of an error-processing system on the brain [32].

The electrophysiology of the error-related potentials and their associated variations were studied through the analysis of the raw EEG, and with a filtered EEG where those components not originated in the brain sources involved in the error-related potentials were removed. The filter eliminated several types of artefacts, including electromyographic activity (such as that provoked by scalp and neck muscles), ocular activity (such as eye movements and blinks) and brain activity not originated within the error processing brain sources [112] (such as spatial attention components [113]). The filter is constituted by two main steps: (*i*) application of independent component analysis (ICA) [114]; and (*ii*) isolation of the independent brain components related to error processing, with a posterior re-projection of this information to the sensor space. Note that while ICA techniques have been widely used for the characterization of brain sources [115–117] and the removal of artefacts [118], the filter proposed herein focused on the isolation of the brain process of interest (see [119] for a similar approach for P300 classification).

The ICA spatial filter is a statistical model defined as $\mathbf{x} = A\mathbf{s}$, where $\mathbf{x}$ are the input data, and A and $\mathbf{s}$ are the mixing matrix and independent components estimated by maximizing the temporal independence among the components. Each column vector $\mathbf{a}_i$ of A is the spatial pattern associated with the component $\mathbf{s}_i$. While there are many ways to compute the ICA model [114], its computation has two difficulties in the EEG context: the number of independent components to estimate [117] and the non-reliable nature of the estimation process [120]. The number of components d was estimated using m-fold cross-validation principal component analysis (PCA) in the sensor space [121] (the number of folds was fixed to $m = 5$ in the experiments, leading to $d \simeq 15$ dimensions retained from the original 32 channels). The ICASSO technique was used to address the non-reliability of ICA [120]. ICASSO estimates N ICA filters using the FastICA algorithm [114] under changes of the initial conditions, and then performs clustering on the obtained estimations (N was fixed empirically to 100). Once the ICA model was computed, the DipFit source localization [117] was used to estimate the neural source of each component. Those independent components whose brain source was in the ACC or the pre-SMA were selected, as these areas are believed to be the main generators of error-processing brain activity [32, 42, 116]. For each subject and task/subtask, the number of components selected was between one and four, which were re-projected back to the sensor space to obtain the filtered data.

The analysis of the shape and timing of the potentials (with and without the filter) for each task and subtask was carried out through the computation of time-locked averaged potentials for the error and non-error potentials in channel FCz, through the difference average (error minus non-error averages) [46, 79] and by an r^2 discriminability test [1]. A topographic interpolation of the potentials was obtained at the time of the main peaks of the difference average. A source localization analysis was also performed with sLoreta [122] at the N4 component of the error averages of each operational task [46]. Additionally, the peak amplitudes and latencies of the most prominent negativity were extracted from single-trial signals as the minimum value within the time window $[320, 600]$ ms in channel FCz. The latency-sorted single-trial potentials were plotted as a colour-encoded image with a smoothing window of 50 trials.

Finally, to assess the statistical differences among tasks/subtasks of the error-related potentials, one-way within-subjects ANOVAs (factor: tasks or subtasks) were conducted over the latencies and amplitudes of each component (N2, P3 and N4) of the difference average, averaged from channels Fz, FCz and Cz [50, 123]. When needed, the Geisser-Greenhouse correction was applied to the data to assure sphericity.

3.2.4 Analysis of the impact of task-dependent signal variations

Once the existence of signal variations was studied, its impact on BCI performance was analyzed at two levels: (*i*) changes in the features distributions used for error detection and (*ii*) the corresponding classification accuracy. This analysis was performed using the filtered EEG data to avoid the influence of activity not originated in the error-processing brain areas (i.e., artefacts).

Feature extraction

Previous studies have demonstrated that amplitude values of the error-related potentials from several fronto-central channels are suitable features for their discrimination (error vs non-error) [46, 47, 79]. In this study, features are constructed as linear combinations of amplitudes of channels and time points that best separate these two classes [80]. Given a set of n labelled trials of the two classes, for each trial, eight fronto-central channels (Fz, FC1, FCz, FC2, C1, Cz, C2, and CPz) within a time window of $[200, 800]$ ms were subsampled to 64 Hz and concatenated as a vector of 312 features. The feature vectors of all trials were normalized, and then decorrelated using PCA, retaining 95% of the explained variance. The k-most discriminant features were selected based on a robust variant of the Fisher score [72]:

$$FS(\mathbf{f^i}) = \frac{|med(\mathbf{f^i_1}) - med(\mathbf{f^i_2})|}{med_{ad}(\mathbf{f^i_1}) + med_{ad}(\mathbf{f^i_2})}, \tag{3.1}$$

with $med(\mathbf{f^i_j})$ and $med_{ad}(\mathbf{f^i_j})$ being the median and the median absolute deviation of feature $\mathbf{f}^i$ for class $j \in \{1, 2\}$. The number of features to retain k was determined by a ten-fold cross validation.

The effect of signal variations was measured by the statistical significance between the features distributions for each class, between operational tasks (inter-task) and between subtasks (intra-task). One-way within-subjects ANOVAs (factor: tasks or subtasks) were conducted on the single-trial features of each class to assess the statistical significance. In addition, the inter/intra-task similarity of the features' distributions was quantified by the Kullback-Leibler (KL) divergence. The KL divergence from $P \sim \mathcal{N}(\mu_P, \Sigma_P)$ to $Q \sim \mathcal{N}(\mu_Q, \Sigma_Q)$ is:

$$D_{KL}(P||Q) = \frac{1}{2}\left(tr(\Sigma_Q^{-1}\Sigma_P) + \mathbf{v}^T \Sigma_Q^{-1}\mathbf{v} - \ln\left(\frac{|\Sigma_P|}{|\Sigma_Q|}\right) - k \right) \tag{3.2}$$

with $\mathbf{v} = (\mu_Q - \mu_P)$. High values of $D_{KL}(P||Q)$ entail large differences between distributions. The KL divergences were computed using the $k = 10$ most discriminant features (according to equation 3.1) to compare the intra/inter features distributions.

Single-trial classification

The classifier used in the analysis was a regularized version of the linear discriminant analysis (LDA) [124]. The LDA discriminant function $D(\mathbf{f})$ is the hyperplane that maximally separates the feature distributions corresponding to two classes: $D(\mathbf{f}) = \mathbf{w}^T\mathbf{f} + b$, where $\mathbf{f}$ is the feature vector to be classified, and $\mathbf{w}$ and b are the normal vector to the hyperplane and the corresponding bias computed by: $\hat{\mu} = \frac{1}{2}(\hat{\mu}_1 + \hat{\mu}_2)$, $\mathbf{w} = \tilde{\Sigma}^{-1}(\hat{\mu}_2 - \hat{\mu}_1)$, $b = -\mathbf{w}^T\hat{\mu}$; where $\hat{\mu}_j$ is the sample mean of class j, $\hat{\mu}$ the sample global mean, and $\tilde{\Sigma}$ the regularized sample covariance matrix (shared for both classes).

Regularization aims to minimize the covariance estimation error $E = |\Sigma - \tilde{\Sigma}|$, with Σ being the real covariance matrix, by penalizing very large and very small eigenvalues. The regularized covariance matrix was computed by: $\tilde{\Sigma} = (1 - \gamma(n))\hat{\Sigma} + \gamma(n)\nu I$; where n was the number of trials used for training, $\gamma(n) \in [0, 1]$ the regularization factor (whose value can be computed numerically [124]), $\hat{\Sigma}$ the sample covariance matrix, and $\nu = tr(\hat{\Sigma})/k$ the average eigenvalue of $\hat{\Sigma}$, with k being the number of diagonal elements of $\hat{\Sigma}$.

To tackle the signal variations, the classifier was adapted based on a sequential process in which labelled examples of the new task were used to modify the discriminant function of the LDA classifier [72]. Namely, given a new example of class j at time t, $\mathbf{f}_j(t)$, the mean $\hat{\mu}_j$ was updated using an exponential moving average:

$$\hat{\mu}_j(t) = (1 - \alpha)\hat{\mu}_j(t - 1) + \alpha\mathbf{f}_j(t). \tag{3.3}$$

$\alpha \in [0, 1]$ is the update parameter (fixed to $\alpha = 0.05$ [72]) and the initial values for $\hat{\mu}_j$ were obtained using data of another task or subtask. The corresponding discriminant function was recomputed using the new mean $\hat{\mu}_j$ to update the $\mathbf{w}$ and b parameters accordingly.

The impact of signal variations was analyzed firstly without adaptation, and then with the supervised adaptation. Firstly, the classifier was trained with examples of one task and tested with another one (inter-task) or trained with examples of one subtask and tested with the other subtasks (intra-task). The results were then compared with the performance of the baseline classifier computed using a ten-fold cross-validation scheme for each task (subtask) separately. Secondly, the adaptation was evaluated against the baseline classifier performance as a function of the number of trials used to train/adapt the classifier (i.e., calibration time). The adaptive classification results were obtained using the train-test sets, as follows: the classifier was initially trained using the train dataset and then, the test dataset was split into two subsets (D_1 with 300 examples and D_2 with the remaining). For each trial at time t, the classifier was updated using equation 3.3 and tested on the D_2 dataset. The results were then compared with the performance of baseline classifiers built using trials $[1 \ldots t]$ of D_1 and tested on D_2. This process was repeated 10 times to reduce variability in the results for adaptive and baseline classifiers while shuffling trial positions, and then averaging the obtained accuracies.

3.3 Results

3.3.1 Electrophysiology of potentials their signal variations

The analysis comprises the raw EEG data along with the filtered EEG data (see subsection 3.2.3). The proposed filter eliminated 90% of the ICA components that were not estimated within ACC or pre-SMA. The majority of these components were ocular artefacts such as eye movements (estimated in frontal areas such as Brodmann areas 10, 11 or 38) and brain activity estimated either in parieto-occipital/occipital areas (Brodmann areas 17, 18 and 19) or the posterior cingulate cortex (PCC, Brodmann areas 23 and 31). Although these components contributed to the EEG, they were not originated in the main error-processing areas and thus were eliminated by the filter. Artefact correlation with the subtasks but not with the tasks is an effect worthy of mention, which might affect feature extraction and classification analysis of the generalization study. Figure 3.2 displays an example with the raw and filtered EEG for the subtask OT1.Right. Without filtering, a signed r^2 discriminability test indicated that the most discriminant features were on frontal channels (originated by lateral eye movements); on the other hand, after filtering, the most discriminant features were due to fronto-central activations (originated by error-related potentials). Note that without filtering, the most discriminant features may greatly discriminate the potentials within the subtask OT1.Right (due to the lateral eye movements), but would not generalize for the other subtasks or task as they involved different eye movements.

For both operational tasks and subtasks, the average difference of the raw/filtered potentials for both conditions presented a small negative deflection approximately at 250 ms (an N2 component) and prominent positive and negative peaks (P3 and N4 components) at approximately 300 ms and 500 ms (in agreement with the r^2 test, figure 3.3). The topographical scalp maps at these last two peaks showed fronto-central activations for the two operational tasks.

When not filtering the data, source estimations for OT1 (at 500 ms of the error average) were in the paracentral lobule (Brodmann Area 5), whereas for OT2 (at 450 ms of the error average) were in the ACC (Brodmann area 24). On the contrary, when filtering the data the potentials from both operational tasks were estimated within the ACC, agreeing with previous studies on

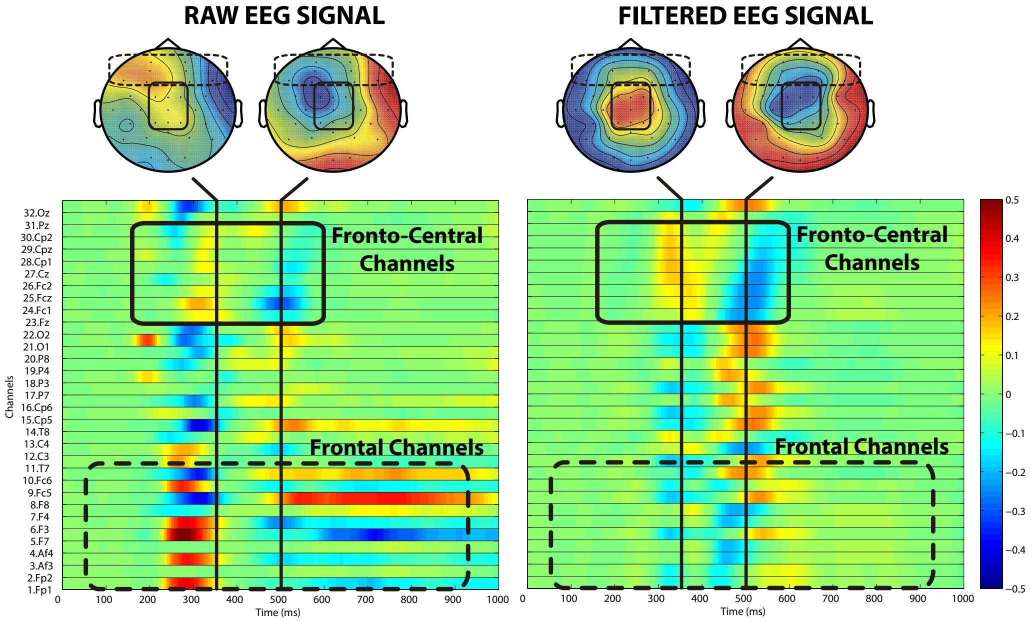


Figure 3.2: Signed r^2 discriminability test of non-error versus error, performed on the subtask OT1.Right, when not filtering the signal (Left) and when filtering the signal (Right). The x-axis represents the time (from 0 to 1000 ms) and the y-axis represents each recorded EEG channel. Topographic interpolation of the r^2 is shown at 350 and 500 ms. The solid boxes mark the position of fronto-central channels, whereas the dashed boxes mark the position of frontal channels. When not filtering the signals, most of the discriminability comes from frontal channels with the sign reversed on the left and right hemispheres. When filtering the signals, most of the discriminability comes from fronto-central channels.

error-related potentials [46, 47, 78, 116]. These results indicated that the use of the ICA filter was advisable for the isolation of the error-processing brain activity.

Regarding the inter-task analysis, visual inspection revealed that the shape of the averaged potentials differed between the two operational tasks (Figure 3.3). In OT1, the averaged error-related potentials presented two positive peaks at 200 and 300 ms, whereas in OT2 the first positive peak was smaller, with a more prominent peak at 280 ms. This difference was also appreciated in the sorted single-trial error-related potentials. Additionally, the ANOVA analyses reported statistical differences on the latency of the three main components both for the raw ($F_{1,9} = 9.574, p = 0.01$; $F_{1,9} = 24.469, p = 0.001$; and $F_{1,9} = 48.442, p = 1 \cdot 10^{-4}$ for the N2, P3 and N4 components) and filtered data ($F_{1,9} = 7.789, p = 0.02$; $F_{1,9} = 24.970, p = 0.001$; and $F_{1,9} = 28.809, p = 0.0005$). For the amplitude of the components, statistical differences were found only for the N4 component of the filtered data ($F_{1,9} = 15.66, p = 0.003$). These results indicated the existence of signal variations in the error-related potentials between operational tasks affecting mainly the latency of their main components.

Regarding the intra-task analysis, visual inspection revealed that the shape of the averaged potentials was very similar among subtasks (Figure 3.3). The ANOVA analyses reported no statistical differences ($p > 0.05$) except for the N2 component of OT1 for the raw data ($F_{2,18} = 12.021, p = 0.0005$). These results indicated that on average, the components did not change among subtasks of the same task.

3.3.2 Features analysis

Regarding the inter-task analysis, visual inspection of the features showed that only the best feature (f^1) reflected similar patterns between OT1 and OT2, whereas the other features presented different spatio-temporal combinations (see Figure 3.4 Top-Middle for representative examples). In the intra-task case, the features were very similar between subtasks. For instance, the best features (f^1 to f^3 in the figure) were almost equal among subtasks while the worst feature f^{10} presented greater variations.

For each class (non-error and error), feature distributions were significantly different for inter- and intra-tasks (ANOVA test, $p < 0.001$ in all the cases). For the inter-task case, KL divergences were 0.64 ± 0.34 and 1.71 ± 0.46 for non-error and error respectively, while for the intra-tasks of OT1 and OT2 the divergences were 0.64 ± 0.51 and 1.06 ± 0.28, and 0.44 ± 0.19 and 1.92 ± 0.60. For all tasks and subtasks, the non-error KL divergences were significantly lower than the error KL divergences (unpaired one-tailed

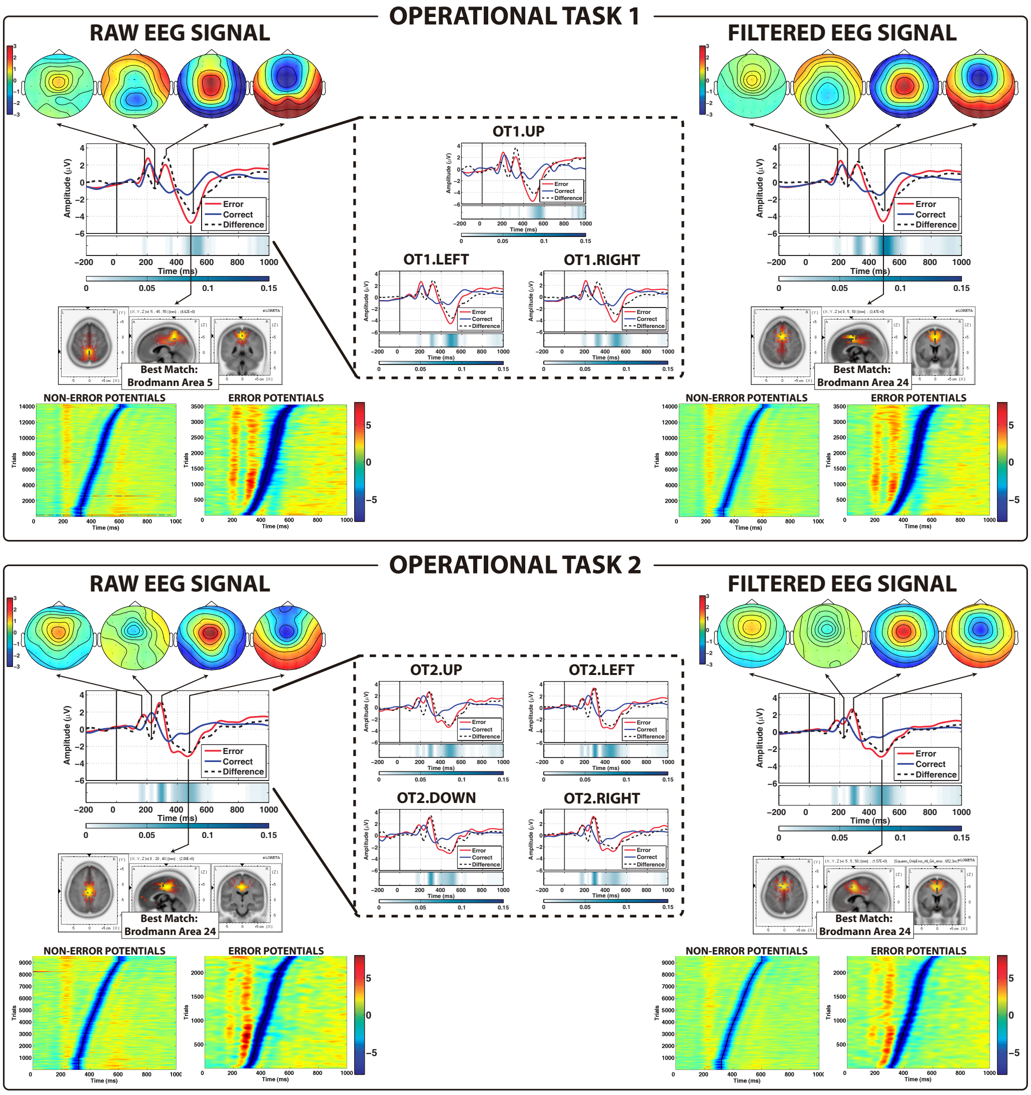


Figure 3.3: Time-locked grand averaged signals for the raw EEG data (left) and after data filtering (right) on channel FCz (averaged for all subjects) for OT1 (top), OT2 (down) and averages of each subtask for the raw data (centre). The time range is $[-200, 1000]$ ms with 0 being the onset of the action. Error and non-error potentials are in red and blue respectively, and the difference averages (error minus non-error averages) are in dashed lines. The r^2 discriminability test [1] between error and non-error potentials is below each plot, where dark colours indicate high values (i.e., large differences) between the potentials in both conditions. The spatial location of each peak of the difference average is displayed as topographical scalp maps, as well as the source location of the error grand average at the most prominent negativity (500 ms and 450 ms for OT1 and OT2). The single-trial potentials sorted by the negative peak latencies are shown below the source localization as a colour encoded image (red and blue indicate amplitudes higher and lower than 0 μV respectively).

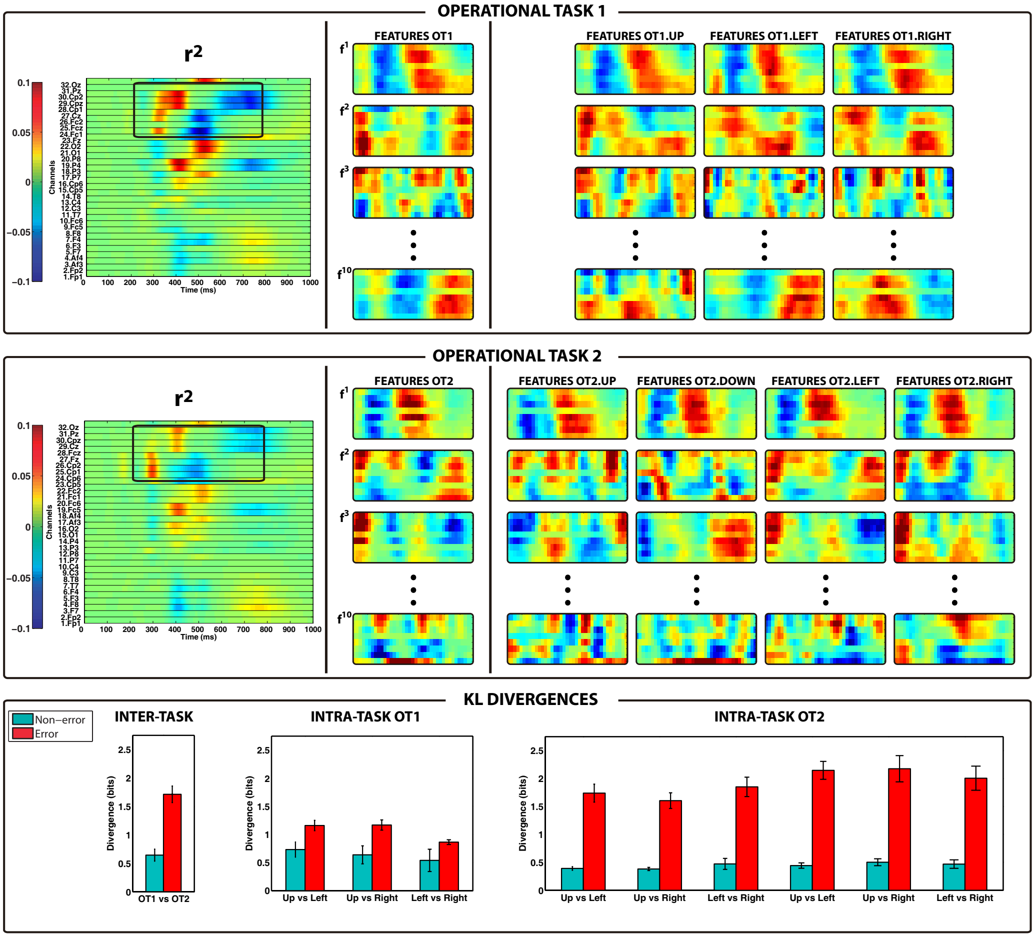


Figure 3.4: (Top-Middle) Representative examples of the feature extraction process for each operational task. The r^2 metric (Left) was used to choose the time window of $[200, 800]$ ms in fronto-central channels and then extract the initial features, as in [79]. These features were the inputs to the spatio-temporal filter, whose outputs were the k-most ($k = 10$ for the features analysis) discriminant features, each of them encoding combinations of time points and channels. The weights of some features for each task and subtask are shown as a colour encoded image (blue and red indicate negative and positive weights, respectively). (Lower part of figure) Bar plots of the KL divergences (mean ± SEM) between the features distributions for the inter-task and intra-task conditions (blue and red for the non-error and error distributions).

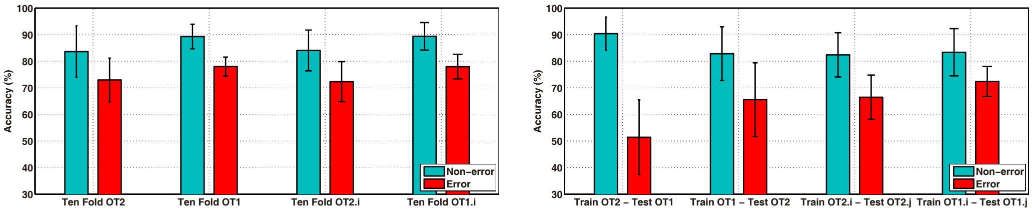


Figure 3.5: Mean ± std classification accuracies averaged for all the subjects, for the (Left) inter-task and their baseline and (Right) intra-task and their baseline. Blue and red bars indicate accuracies for non-error and error potentials respectively.

t-test, $p < 1 \cdot 10^{-4}$). The inter and intra-task KL divergences of the error distributions for OT1 were significantly different (unpaired two-tailed t-test, $t_{38} = 5.36, p = 4 \cdot 10^{-6}$), but the inter/intra-task divergences of the non-error distributions were not ($t_{38} = 0.04, p = 0.97$). For OT2, inter/intra-task KL divergences were significantly different for the non-error distributions ($t_{68} = 2.69, p = 0.01$), but no significant differences were found for the inter/intra-task divergences of the error distributions ($t_{68} = -1.05, p = 0.30$). In summary, feature distributions changed significantly between tasks and among subtasks, and the error distributions changed significantly more than the non-error distributions. Furthermore, the features varied significantly more when changing the task, than when changing the subtask.

3.3.3 Classification

Analysis without adaptation

The ten-fold accuracies of non-error and error potentials were, on average, 89.29% and 78.00% for OT1 and 86.64% and 73.00% for OT2; and 89.38% and 77.97% for subtasks of OT1, and 84.06% and 72.33% for subtasks of OT2 (see Figure3.5). All baseline classifiers were above the chance level.

Regarding the inter-task generalization results (Figure 3.5,Left), when training with OT1 and testing with OT2 there was significant average increase of 6.76% (one-tailed paired t-test, $t_{18} = 1.86, p = 0.04$) in the detection of non-error potentials, but a significant decrease of 21.54% ($t_{18} = -4.19, p = 0.0003$) for error potentials. As can be seen, the standard deviation was also increased compared to the ten-fold accuracies. This indicated that the accuracy drops varied substantially from subject to subject, with subjects having large drops, and others having almost no accuracy decrease. When training with OT2 and testing with OT1, the accuracies presented significant decreases of 6.44% and 12.44% ($t_{18} = -1.83, p = 0.04$ and $t_{18} = -2.75, p = 0.007$), respectively. As with the previous case, the standard deviations increased, and thus the drops varied substantially from subject to subject.

Regarding the intra-task generalization results (Figure 3.5,Right), there was a general decrease in classification accuracy with respect to the ten-fold accuracy. For the subtasks of OT1, significant average decreases of 5.99% and 5.58% ($t_{18} = -1.84, p = 0.04$ and $t_{18} = -2.42, p = 0.01$) were identified for non-error and error potentials. For the subtasks of OT2, average decreases of 1.65% and 5.84% were obtained, but they were not significant ($p > 0.05$) (see the supplementary materials for the accuracies obtained for each subject).

Analysis with adaptation

For the operational tasks, the baseline accuracies reached maximum mean accuracies of 81.26% and 78.30% for OT1 and OT2 after 300 trials. For the subtasks, the baseline reached maximum mean accuracies of 81.51% and 77.63% for OT1 and OT2 after 300 trials. Note that the accuracy convergence was fast, since only few examples were needed to reach high accuracies. For instance, the baseline classifier obtained accuracies of 79.49% and 76.63% (tasks) and 79.09% and 74.56% (subtasks) with 100 trials (approximately 5 minutes of EEG recording, see Figure 3.6, black lines).

The adaptive classifier started with the mean accuracies obtained with the classifiers of the previous subsection (c.f. Fig. 3.5), and as more examples were available, the accuracy of the adaptive classifier increased (see Fig. 3.6). After 300 trials (examples), the mean accuracies were 74.07% and 78.33% for the inter-task cases, and 78.26% and 74.22% for the intra-task cases. This increase in performance was due to a reduction in accuracy differences between the two classes, more relevant in the inter-task case.

In the inter-task analysis (training with OT1 and testing with OT2 and viceversa), the adaptive classifier started with better accuracies than the baseline classifier but: (*i*) after 66 trials and 32 trials, the baseline classifier outperformed the adaptive classifier; and (*ii*) after 300 trials the adaptive classifier presented accuracies 4.23% and 2.93% lower than the baseline (Fig. 3.6, first and second columns). The intra-task analysis showed similar results (Fig. 3.6, third and fourth columns): The adaptive classifier presented worse accuracies than the baseline after 51 and 65 trials for OT1 and OT2 respectively, and a lower accuracy compared to the baseline, with drops of 3.25% and 3.41% after 300 trials.

In summary, the supervised adaptation achieved high accuracies from the beginning of the new task/subtask that were improved as more exam-

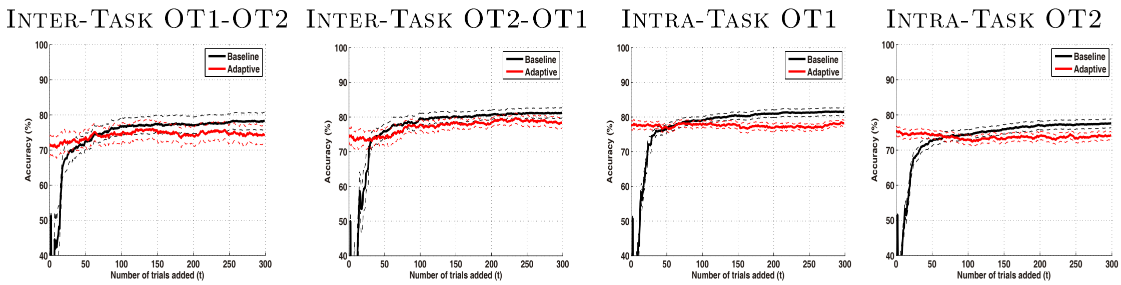

Inter-Task OT1-OT2 Inter-Task OT2-OT1 Intra-Task OT1 Intra-Task OT2

Figure 3.6: Mean ± SEM classification (solid ± dashed lines) accuracies averaged for all subjects, for the baseline and adaptive classifiers across trials. The results are shown for the inter-task (two left columns) and intra-task (two right columns) conditions. For each plot, the x-axis indicates the number of trials used to adapt the previous classifier (adaptive classifier, shown in red lines), or the number of trials used to train the classifier (baseline classifier, shown in black lines), and the y-axis represents the single-trial accuracy.

ples were available. The baseline classifier (calibrating from scratch the new task/subtask) started with lower accuracies than supervised adaptation but rapidly outperformed the latter as the number of examples used to train the baseline classifier increased. In all the situations, less than 100 examples (five minutes of EEG recording) were sufficient to calibrate the BCI to obtain better accuracies than adaptation.

3.4 Discussion

This paper studied and showed the presence of signal variations in error-related BCIs, where the mental task was coupled with the operational task. The electrophysiology analysis presented statistical differences mainly in the latencies of the three error-related potential components. There is previous evidence that the error-related potentials remain very similar between different days [109] or even months and years [47]. Thus, the authors understand that the variations of these potentials were due to the difference of operational tasks and not to the fact that the experimentation was performed in different days.

The study analyzed the raw EEG as well as a filtered EEG signal that retained only EEG information whose origin was estimated in the brain areas involved in the generation of the error-related potentials. This filter played a crucial role in the analysis as, while the grand averages showed that the most prominent activity was due to errors, an r^2 test revealed the presence

Table 3.1: Inter-task accuracies (Mean ± std). Features comparison.

	Ten-Fold OT1		**Ten-Fold OT2**	
	Features as in [46, 47, 79]	**Features used in this work**	**Features as in [46, 47, 79]**	**Features used in this work**
Non-error	90.91 ± 4.05	89.29 ± 4.62	86.21 ± 8.38	83.64 ± 9.67
Error	79.31 ± 4.95	78.00 ± 3.57	73.46 ± 7.42	72.96 ± 8.23
	Train OT2 - Test OT1		**Train OT1 - Test OT2**	
	Features as in [46, 47, 79]	**Features used in this work**	**Features as in [46, 47, 79]**	**Features used in this work**
Non-error	84.43 ± 9.22	82.85 ± 10.11	91.72 ± 5.61	90.40 ± 6.25
Error	64.39 ± 11.89	65.56 ± 13.87	50.87 ± 15.84	51.42 ± 14.04

of artefacts within subtasks of activity not associated to the error-related potentials, but correlated to the erroneous/non-erroneous actions of the device. This affected the intra-task feature extraction and classification process, in such a way that the artefactual information helped to improve the ten-fold intra-task classification (with information not related to the error-processing, i.e., artefacts) but it did not generalize for other subtasks of the same task. For instance, when using raw EEG data from the separated OT1 subtasks, the ten-fold classification accuracy was 92.03%/82.86% for non-error and error potentials respectively, while using the filtered EEG signal the accuracy was 89.38%/77.97%, respectively. However, when generalization of the classifiers was tested on the other subtasks of OT1, the accuracy of the raw EEG dropped a 17.00%/13.83% for error and non-error responses, while for filtered data the decrease was only of 5.99%/5.58%.

The signal variations affected significantly the distributions of the features selected for classification and the final performance of the trained classifiers. The quantitative analysis (KL divergences) confirmed that the features differences were greater for error than for non-error responses. When generalizing among operational tasks and subtasks, there was a decrease in accuracy with respect to the baseline, which was more pronounced in the inter-task than in the intra-task and more pronounced for error potentials than for non-error potentials (the decrease was not always symmetric for error and non-error potentials). Further studies might focus on understanding whether this effect was either due to the error and non-error event-related activity, or dependent on the experimental procedure (as the dataset presented an unbalanced number of examples for each class).

Feature analysis and classification results depend on the type of features, which in this study were computed based on a spatio-temporal filter that decorrelated signals to maximize the difference between classes. A possible question that arises is whether such feature extraction introduces differences

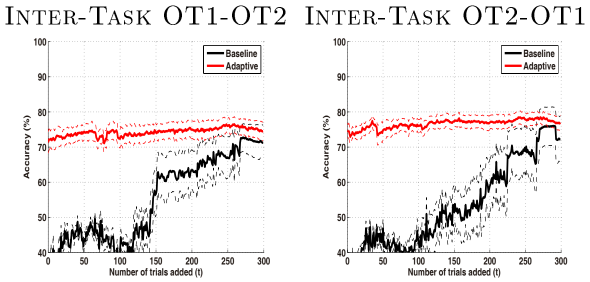

Inter-Task OT1-OT2 Inter-Task OT2-OT1

Figure 3.7: Mean ± SEM classification (solid ± dashed lines) accuracies averaged for all the subjects, for the baseline and adaptive classifiers across trials using the feature extraction method as in [46, 47, 79], for the inter-task case.

with respect to the use of EEG amplitudes in the selected channels and time windows, as it is a widespread procedure in these BCIs [46, 47, 79]. Table 3.1 compares the results obtained using both types of features computed after filtering the EEG for the inter-task generalization (results of the intra-task study were similar). The classification accuracies presented no significant differences neither in the ten-fold baseline case nor in the generalization one, with only slight decreases in the accuracies of around 1% when using the proposed features. Nonetheless, in the performed experiments the proposed feature extraction presented an important advantage over the use of EEG amplitudes in terms of calibration time. The use of a lower number of features (an average of 25 features versus 312) reduces substantially the dimensionality of the classifier's hypothesis space, simplifying the learning process and reducing the calibration time. On the contrary, for higher dimensional spaces (as with EEG amplitudes with no feature selection) the calibration time is much higher and adaptation pays off. Figure 3.7 displays, for the EEG amplitudes, the same comparison between adaptation and calibration shown in Figure 3.6 for the inter-task case. The selection of a set of decorrelated features proposed herein reduces calibration time considerably and achieves a better performance than adaptation in less than five minutes (i.e., less than 100 examples).

In this study, the features were fixed and adaptation was performed only on the classifier. The choice of a different set of features could help reduce the impact of signal variations. For instance, several works have proposed feature extraction methods to find time-invariant features to deal with time-dependent non-stationarities [75], which could be extrapolated to find task-invariant features. However, this would require data from multiple tasks, thus increasing the calibration effort for these BCIs. Other approaches could be used to adapt the classifier to a new task without using labels (i.e., during the device operation), for instance those based on a maximum likelihood estimation of the distribution parameters [125] or on predicted labels [126]. Future work by the authors will consider the possibility of combining both paradigms to jointly adapt the classifier and the features for new tasks, as a way to increase the performance of BCIs for new operational tasks while reducing or removing the calibration effort.

4 | Latency correction of event-related potentials between different experimental protocols

4.1 Introduction

The previous chapter has shown that error-related potentials are affected by the task being performed, changing up to the point of preventing a classifier to generalize or adapt among different tasks. This result was expected, as many neurophysiological studies have shown that event-related potentials (to whom error-related potentials belong) are affected by aspects related to the task performed [50]. In this chapter, we further study the reasons behind this drop in the classification performance. The chapter focuses on two widely studied ERPs: the error-related potentials and the P300 [84]. This requires characterizing the ERPs by the acquisition of enough trials to build a reliable model represented by their grand averages [50]. This is due to the poor signal-to-noise ratio of the EEG as well as several sources of variability that may affect the amplitude or the latency of the ERP components. For instance, the early ERP components (appearing within 200 ms from the stimulus presentation) are affected by application-specific factors such as the spatial attention [113] or the stimuli contrast [50]; as well as user-specific factors such as arousal or valence [49]. In turn, late ERP components (occurring later than 200 ms) are affected by application-specific factors such as the probability of occurrence of the expected stimulus [50] or the inter-stimulus interval [127]; user-specific factors such as the age and the cognitive capabilities [128, 129]; or application- and user-specific variability such as the stimulus evaluation time (i.e., the amount of time required to perceive and categorize a stimulus) [50, 51].

Typically, experiments are designed in a well-controlled manner to reduce the ERP variability. In consequence, it is not clear whether the obtained

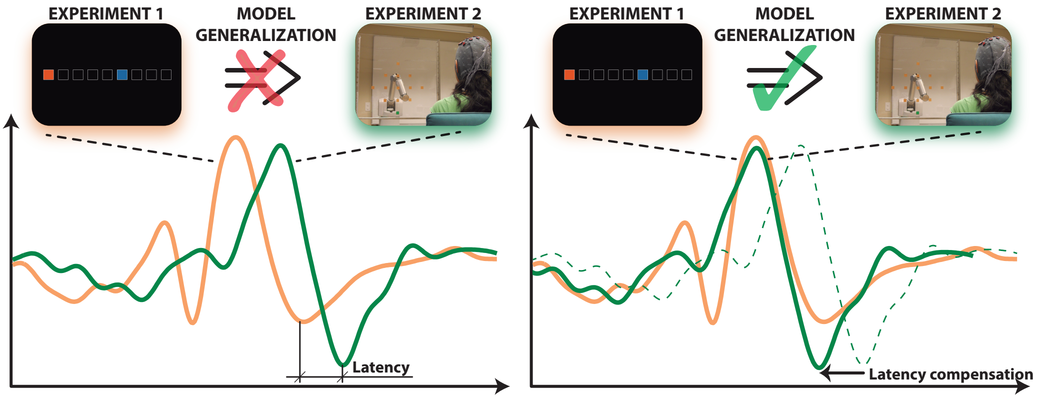


Figure 4.1: (Left) Example of the latency between two grand averaged event-related potentials elicited from different experimental protocols. Such difference prevents from having classifiers that generalize among protocols. (Right) By estimating and removing the latency of the two ERPs, the classifier would be able to work under different experimental protocols.

model also reflects the same neural phenomena under different performed tasks. This is of particular importance for practical BCI applications where decoding algorithms are expected to keep their performance level irrespective of external factors. Moreover, BCIs often exploit the same brain processes in different applications where changes are introduced in the used stimuli, the feedback modality or the controlled device (e.g., see [45, 47, 83, 130] for different applications based on error-related processing). In the ideal case, these systems should be able to generalize across different operating BCIs independent of the controlled device. In practice, however, there is a need for training a model for each new experimental protocol or session, which is a time-consuming operation and a major issue when deploying BCIs out of the lab. To address this issue, previous researches have tried to reduce this calibration time either by using adaptive classifiers [71, 83], or by initializing the model with data from a pool of subjects [73, 74].

Although previous studies have described the effect of variations in the ERP amplitudes [47] and latencies [131] within the same BCI experimental protocol, the effect of these variations among different protocols remains unclear. We hypothesize that it could be possible to build or re-adjust models that compensate for these variations by using information from previous experimental protocols, thus enabling generalization of existing BCI decoders to different protocols or applications. In fact, the previous chapter has shown that this is the case for a specific ERP, the error-related potentials (see section 3.3.1). The main idea is depicted in Figure 4.1 Left, where two experimental protocols elicit the same ERP with similar waveforms and amplitudes but different latencies. If we could estimate the latency variations between the two experimental protocols, previous models could be used in the new protocol after compensating for the latency shift (see Figure 4.1 Right).

In this chapter, we analyze the effect of ERP amplitude and latency variations among different experimental protocols based on the same cognitive process. We also present a method to analyze and compensate for the latency variations in BCI applications. Two widely used signals were analyzed: the P300 evoked potentials [9, 50, 84] and the observation error-related potentials (ErrP) [42, 45, 47]. For each kind of ERP, three different experimental protocols with different levels of difficulty were designed. The latencies between protocols were studied from two points of view: the characteristics of the ERPs and the single-trial classification. The results illustrate (*i*) how the experimental protocols significantly affect the latency of the recorded potentials but not the amplitudes, and (*ii*) how the use of latency-corrected data allows for the generalization of BCI decoders, reducing this way the calibration time when facing a new experimental protocol. This work extends our previous work [85] with a more robust technique to compensate the latencies and shows its application to ERPs of different nature.

4.2 Experimental methods

We focus on two types of ERPs: the P300 evoked potentials and the observation error-related potentials (ErrP). For each of these signals, three types of experimental protocols were designed (i.e., three different ways of evoking the P300 and the ErrPs).

4.2.1 Data recording and experimental setup

EEG was recorded with a gUSBAmp amplifier (*gTec medical engineering*, Schiedelberg, Austria) with 16 active electrodes, with the ground and reference placed on the forehead and the left earlobe. Different montages were made for the P300 and ErrP protocols (see details below). The EEG was digitized at 256 Hz, power-line notch filtered at 50 Hz, and zero-phase band-pass filtered at $[1, 10]$ Hz. Participants were seated on a comfortable chair facing the visual displays of the protocols approximately one meter away. During all experiments participants were asked to restrict eye movements and blinks to specific resting periods.

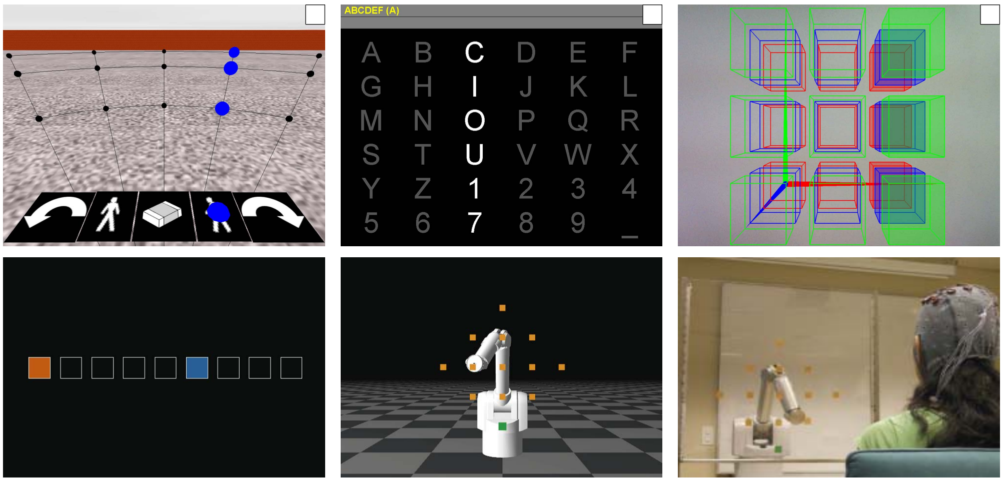


Figure 4.2: Experiments performed for the (Top) P300 potentials and (Bottom) observation error potentials (from left to right: experiments 1 to 3).

P300 experimental protocols

For these protocols we recorded EEG signals with the BCI2000 framework [110] from 16 active electrodes located at Fp1, Fp2, Fz, FC1, FCz, FC2, Cz, CP1, CPz, CP2, P3, Pz, P4, O1, Oz and O2 according to the 10-10 system. Five participants (one female, mean age 27.80 ± 2.49 years) took part in the study. We synchronized the onset of visual stimuli with the EEG by means of an optical trigger placed on the monitor [132]. This removed latencies introduced by the protocol implementation and thus the latency variations across experiments were restricted to the user side [51].

Three experimental protocols were used to evoke the P300 potentials (Figure 4.2, Top), with different types of stimuli. The stimulation process followed the oddball paradigm [84], where subsets of potential targets (e.g. an entire row or column) are sequentially highlighted in random order. The stimulus (row or column) remained highlighted for 125 ms on the screen, and the inter-stimulus interval was random within the range $[1.7, 3.0]$ s. The participants were instructed to observe the stimulation process fixing their attention to a given target, and to count the times the target was highlighted while ignoring the other targets. All participants executed the experiments in the same order, each experiment lasting $\approx$ 1.5 hours and with a time between experiments of 1.10 ± 0.81 days.

Experiment 1, 2D Simulated Wheelchair (Figure 4.2 Left, Top) [9] The visual display showed a virtual environment with 20 possible targets to drive a wheelchair, located in 2D in a 4x5 matrix. For the stimulation process, the rows and columns were highlighted showing a blue dot over each possible target position. The probability of target appearance was 22%. For each subject, all possible target positions were recorded, obtaining 144 target (P300) and 720 non-target responses respectively.

Experiment 2, 2D Speller (Figure 4.2 Middle, Top) [84] The visual display showed a matrix of 36 possible letters to spell represented in 2D as a 6x6 matrix. The stimulation was made by illuminating the corresponding row or column. The probability of target appearance was 17%. For each subject, all possible target positions were recorded, obtaining 200 and 700 target and non-target responses respectively.

Experiment 3, 3D Augmented Reality Protocol (Figure 4.2 Right, Top) The display showed a gray background and 27 possible targets located in 3D in a 3x3x3 matrix. The stimulation was made by illuminating rows, columns, and depths. To facilitate the user's distinction among the three depths, each depth was illuminated with a different colour (green, blue or red). The probability of target appearance was 33%. For each subject, all possible target positions were recorded, obtaining 273 and 610 target and non-target responses respectively.

Error potentials experimental protocols

We recorded ErrPs with a custom C++ framework using 16 active electrodes located at Fz, FC3, FC1, FCz, FC2, FC4, C3, C1, Cz, C2, C4, CP3, CP1, CPz, CP2, and CP4 according to the 10-10 system. Six participants (one female, mean age 27.33 ± 2.73 years) took part in the study. In these experiments, the use of an optical trigger was not possible since one experiment involved a real robotic device instead of visual stimuli on the screen (see Experiment 3). Thus, latency variations could be originated by both the subject and the implementation (i.e. the amount of time of receiving and executing the delivered command).

The experimental protocols designed to elicit error potentials (Figure 4.2, Bottom) had different setups (and devices), where in all cases the goal of the device was to reach a target from different starting points. The device executed random movements with approximately 30% probability of performing an erroneous movement. The time between two movements was random within the range $[1.7, 4.0]$ s. The target position was randomly changed after 100 actions. The participants were instructed to observe the device movements and evaluate them as correct when there was progress towards the target position, and as incorrect otherwise. Each participant executed the experiments in the same order, each experiment lasting ≈ 2.5 hours and with a time between experiments of 17.58 ± 10.09 days.

Experiment 1, Virtual Moving Square (Figure 4.2 Left, Bottom) [47] The visual display showed a one-dimensional space with 9 possible positions (marked by a horizontal grid), a blue square (device) and a red square (target). The device could execute two discrete actions: move one position to the left or to the right. For each subject, the left- and right- most positions were tested as targets, and around 250 and 600 error and non-error potentials were recorded.

Experiment 2, Simulated Robotic Arm (Figure 4.2 Middle, Bottom) The display showed a simulated robotic arm (Barrett WAM) with 7 degrees of freedom (device) [133] moving within a two-dimensional space with 13 possible positions (marked in orange), and a target location (green square). The robot was situated behind the squares pointing at one position, and could perform four possible actions: moving one position to the left, right, up, or down. The robot actions were continuous, with each displacement lasting ≈ 500 ms. For each subject, the left-, right-, up- and down-most positions were tested as targets, and around 300 and 700 error and non-error potentials were recorded.

Experiment 3, Real Robotic Arm (Figure 4.2 Right, Bottom) This experiment followed the same configuration of Experiment 2 but using a real Barret WAM robotic arm (*Barret Technology Inc.*). The user was seated two meters away from the robot, and between them there was a transparent panel to mark the positions (the distance between two neighbor positions was 15 cm). For each subject, the left-, right-, up- and down-most positions were tested as targets, and around 300 and 700 error and non-error potentials were recorded.

4.2.2 Analysis of Event-Related Potentials

We assessed protocol-dependent variations in the latency and amplitude of the ERPs for each of the experimental protocols. First, the grand averaged signals were computed for each condition (target and non-target trials for the P300; error and correct trials for the ErrP), for the time window $[-200, 1000]$ ms, being 0 ms the stimulus/action onset. Following previous studies, we analyzed the activity over parietal areas from the target average [50] for the

P300, and over fronto-central areas from the difference average (error minus correct averages) for the ErrPs [47]. A one-way within-subjects ANOVA was performed separately for each type of signal (P300 or ErrP), where the factor was the experiment (three levels corresponding to each experiment), and two dependent variables were tested: the peak amplitudes and the peak latencies. For the P300 experiments, the peak amplitudes and latencies were measured from the P3 component (most prominent positive peak) of the target average from the parieto-occipital channels. For the ErrP experiments, the amplitudes and latencies were measured from the P3 and N4 components (most prominent positive and negative peaks) of the difference average from the fronto-central channels. When needed, the Geisser-Greenhouse correction was applied to data to assure sphericity [50]. Pairwise post-hoc tests (t-tests with the Bonferroni correction) were computed to determine the differences between pairs of experiments.

4.2.3 Estimation and evaluation of latencies among different protocols

The first goal is to estimate the temporal variations between two experimental protocols, which can be achieved using the cross-correlation. The cross-correlation has been used in the past for the detection and analysis of brain signals with successful results [131,134,135]. In order to assure the best estimations, the input to the cross correlation (for each channel) were the grand averages of the condition of interest, with the time window narrowing to the event-related potential elicitation. For the P300 experimental protocols, the average ERP for target stimuli within the time window $[50, 400]$ ms was used; for the ErrP experimental protocols, the error average within the time window $[0, 500]$ ms was used. The outputs of the cross-correlation were the maximum correlation value of the two grand averages and the latency variation between them (i.e. the shift that yields the maximum cross-correlation).

We then assessed whether the main ERP change was due to the latency variation and whether this variation could be compensated for. To do so, the latency variation across two protocols was estimated as described above using all the available data. Let E_i and E_j be the datasets from two experiments i and j, we compensate for the variation by shifting the trials in E_j by the estimated latency shift between them, $d_{E_iE_j}$. Then, we computed the same ANOVA test for the peak latencies performed in subsection 4.2.2.

We performed further analysis on how sensitive the latency estimation was with respect to (i) the number of trials used to compute the grand average for experiment j, and (ii) the channel used to perform the estimation. Assuming that data from a previous experiment i is available, we computed the latency variation using training data, E_j^{Tr}, from the new experiment ($E_j^{Tr} \subseteq E_j$). We assessed the estimation using different sizes of the training dataset (ranging from 10 to 200 trials with increments of 10). For each size, we perform 10 repetitions and report the average of the maximum cross-correlation value, $\max(C_{E_i}^{E_j^{Tr}})$, and the average latency variation, $d_{E_i E_j^{Tr}}$. In each repetition the training subset E_j^{Tr} was randomly drawn from E_j, keeping the proportion of target/non-target and error/correct trial. The analysis was performed independently for each recorded channel.

The latency variations were computed in a pair-wise manner among the three experiments for each of the signals of interest. The combinations of experiments tested were E_1E_2, E_1E_3, and E_2E_3 for both the P300 and the ErrPs. For each pair of experiments a within-subjects two-way ANOVA (factors: number of trials and repetitions) was performed on the latency estimations. The ANOVA results served to study the latency variations by the number-of-trials main effect, to determine whether the amount of trials used from E_j led to different latency estimations; and by the number of trials x repetitions interaction, to determine whether different data from a fixed number of trials affected the latency estimations.

As a sanity check, we also evaluated the method by computing the latency variation among datasets from the same experiment ($d_{E_i^1 E_i^2}$), with the two datasets E_i^1 and E_i^2 mutually exclusive. Therefore, this baseline latency computation should give correlations close to one for latencies near to 0 ms.

4.2.4 Single-trial classification of latency-corrected ERPs

The objective of the single-trial classification study was to determine whether it is possible to reduce the calibration time of a new experiment by re-using latency-corrected data from a previous experiment. The same combination of experiments detailed before served as the basis of the study. The process of the latency estimations used the number of trials and repetitions tested in the previous subsection. All the classification parameters (including the latency estimations) were learned using only the training sets.

Feature extraction and classification

Feature extraction was based on a spatio-temporal filter [80]. The filter input was a dataset with labeled trials, and worked as follows: Firstly, the EEG data were common-average-reference (CAR) filtered and downsampled to 64 Hz. For each trial, the features were extracted using a combination of channels and time points. For the P300, eight centro-parietal and occipital channels (Cz, CPz, P3, Pz, P4, O1, Oz, and O2) were used within a time window of $[100, 700]$ ms. For the ErrP, eight fronto-central channels (Fz, FC1, FCz, FC2, C1, Cz, C2, and CPz) were used within a time window of $[200, 800]$ ms. For both cases, this resulted in a feature vector of 312 features per trial. Then, the features were normalized, and decorrelated using PCA retaining 95% of the explained variance, leading to an average of 45 ± 10 features. Single-trial classification was carried out using a linear discriminant analysis (LDA) [124].

Analysis of the single-trial classification

We compared the accuracies of three different classifiers (see Figure 4.3). The first one, denoted baseline classifier, followed the standard calibration approach of current BCIs, where the classifier for E_j was trained using only the data from E_j^{Tr}. The size of the training data was increased as in the previous subsection to assess the accuracy of the classifier for different calibration times. The second and third classifiers were both trained using all the data from E_i and the data from the new experiment E_j^{Tr}. The difference between them is that in the latter one the latency of dataset E_i was corrected for each channel used for classification, c.f. subsection 4.2.4. These three classifiers were tested on a left-apart fixed subset of E_j (denoted E_j^{Te}), composed of 400 trials (see Figure 4.3). Additionally, the performance of the three classifiers was also compared to the result of performing ten-fold cross-validation (CV) using all the data from E_j.

For each pair of experiments (E_1E_2, E_1E_3, and E_2E_3), two metrics are reported: (i) the mean accuracy ($\frac{target\_acc+non\_target\_acc}{2}$ and $\frac{error\_acc+correct\_acc}{2}$ for the P300 and ErrP experiments respectively); and (ii) the accuracy bias ($|target\_acc - non\_target\_acc|$ and $|error\_acc - correct\_acc|$ for the P300 and ErrP experiments). To assess statistical differences among the classification results, two-tailed paired t-tests were computed, where the p-values were adjusted with the false discovery rate (FDR) procedure [136].

4.2.5 Reducing calibration time during online sessions of experiments

Finally, the latency estimation method was used for real online experiments. For this analysis, 6 new subjects (two females, mean age 27.17 ± 4.07 years) performed the error-related potentials experiments where the calibration time was reduced using latency-corrected data from the previous experiment. For E_1, the standard calibration was followed. On the other hand, during E_2 the

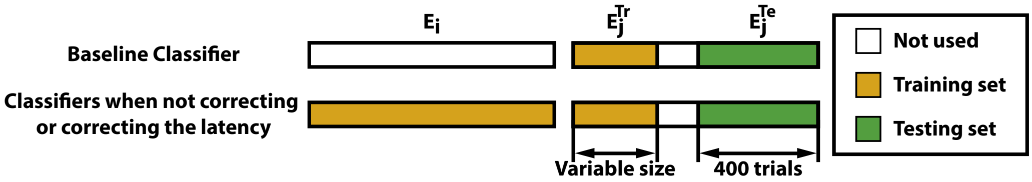


Figure 4.3: Training and testing datasets used for each classifier. For the baseline classifier, the training dataset from the new experiment E_j^{Tr} was used to train the classifier. When information from a previous experiment E_i was available, it was used as part of the training set, either not correcting or correcting the latency $d_{E_iE_j}$. To correct the latency, the training datasets E_i and E_j^{Tr} were fed to the latency correction algorithm, and then the latency from E_i was corrected. To assess the accuracies for different calibration times, the number of trials of E_j^{Tr} was variable (from 10 to 200 trials with increments of 10 trials), while the testing set E_j^{Te} remained fixed to 400 trials.

data from E_1 was latency-corrected using a few trials from E_2. Similarly, during E_3 the data from E_2 was used. The latency between experiments was estimated using the difference average of channel FCz within the window $[0, 500]$ ms. The groups of subjects were denoted control group (the subjects present in this work, where they followed a standard calibration approach) and experimental group (latency-corrected calibration). The calibration phase finished whenever a ten-fold mean accuracy of 75% was obtained in the training data. Then, both groups of subjects performed the online experiment presented in chapter 2, which was use to extract the mean online accuracy obtained.

Mixed two-way ANOVAs (within factor: experiments; between factor: group of subjects) were performed to test whether (i) the number of calibration trials in the experimental group decreased from E_1 to E_2 and from E_2 to E_3; (ii) the number of calibration trials was significantly different between groups; and (iii) the mean online accuracies of both groups were not different. In order to find the underlying significances, post-hoc one-tailed Bonferroni-corrected t-tests were performed.

4.3 Results

4.3.1 Analysis of Event-Related Potentials

Figure 4.4 shows the ERP grand averages of all experiments. In the P300 experiments, as in previous studies [9,50,84], a clear sharp positive peak (P3)

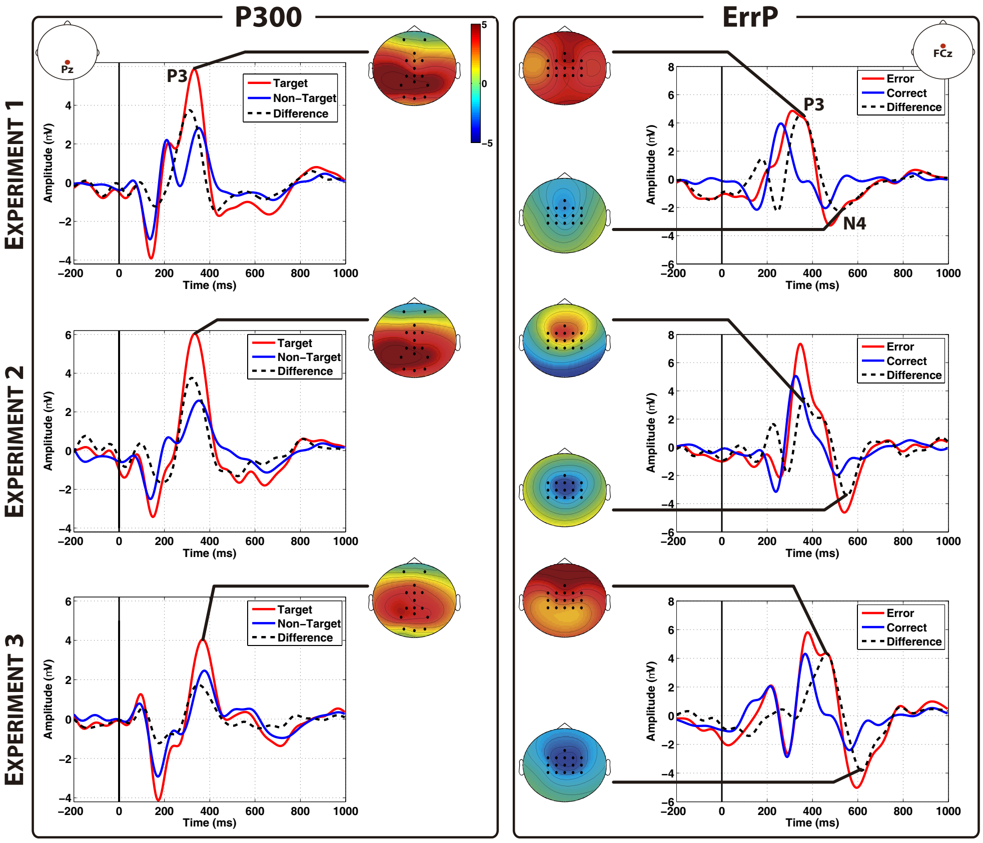


Figure 4.4: Grand averages of each experiment for the (Left) P300 potentials at channel Pz, and (Right) error potentials at channel FCz. Time 0 ms indicates when the stimulus was presented on the screen (P300), or when the device started the action (ErrPs). For the P300, the topographic interpolation of the most prominent positive peak of the target average is shown. For the ErrPs, the topographic interpolation of the most prominent positive and negative peaks of the difference average are shown.

appears on parietal channels after presentation of the target stimuli. For the ErrP experiments, the difference grand averages (error minus correct) are also consistent with the literature [47], with two early positive and negative peaks in fronto-central sites, followed by two larger positive and negative peaks (P3 and N4).

Regarding the P300 experimental protocols, the amplitude of the P3 component showed no statistical differences among the three experiments ($p = 0.123$). In contrast, its latency does exhibit statistical differences ($F_{2,8} = 22.924, p = 0.0005$). Post-hoc tests revealed significant differences between experiments 2 and 3 ($p = 0.032$), and between experiments 1 and 3 ($p = 0.01$), but not between experiments 1 and 2 ($p = 1.0$).

Similarly, no differences were found for the P3 and N4 amplitudes of the ErrPs ($p = 0.510$ and $p = 0.391$ respectively). Interestingly, significant differences were found on the latencies of both the P3 ($F_{2,10} = 29.422, p = 0.00006$) and the N4 component ($F_{2,10} = 6.979, p = 0.013$). For the former, post-hoc tests showed significant differences between experiments 1 and 2 ($p = 0.018$), and between experiments 1 and 3 ($p = 0.003$), and nearly significant differences between experiments 2 and 3 ($p = 0.053$). For the N4 component, there were significant differences between experiments 1 and 3 ($p = 0.006$), but not between experiments 1 and 2 ($p = 0.472$) nor between experiments 2 and 3 ($p = 0.492$). Thus, the main differences on the elicited ERPs across the experiments were due to latency variations of the components, while the amplitudes remained similar.

4.3.2 Analysis of latency estimations

The ANOVA analysis yielded no significant differences in latency after performing the correction for the P300 ($p = 0.12$ for the P3 component), nor for the ErrP experiments ($p = 0.67$ and $p = 0.17$ for the P3 and N4 components, respectively). Thus, the latency correction algorithm successfully removed the latency variations among experiments.

Figures 4.5 and 4.6 (Top) show the maximum correlation (see section 4.2.3) for all electrodes when different numbers of trials from E_j are used. Unsurprisingly, correlation values increase until they converge to an upper value as more trials are used to compute the grand average. ERPs elicited in the P300 experiments (Figure 4.5, Top) show high correlation (≥ 0.8) in parieto-occipital channels when more than 50 trials are used. In turn, the ErrPs (Figure 4.6, Top) required at least 100 trials to yield correlation values higher than 0.8, always over fronto-central channels. These locations, as for the P300, agree with the locations reported as more discriminant for these phenomena.

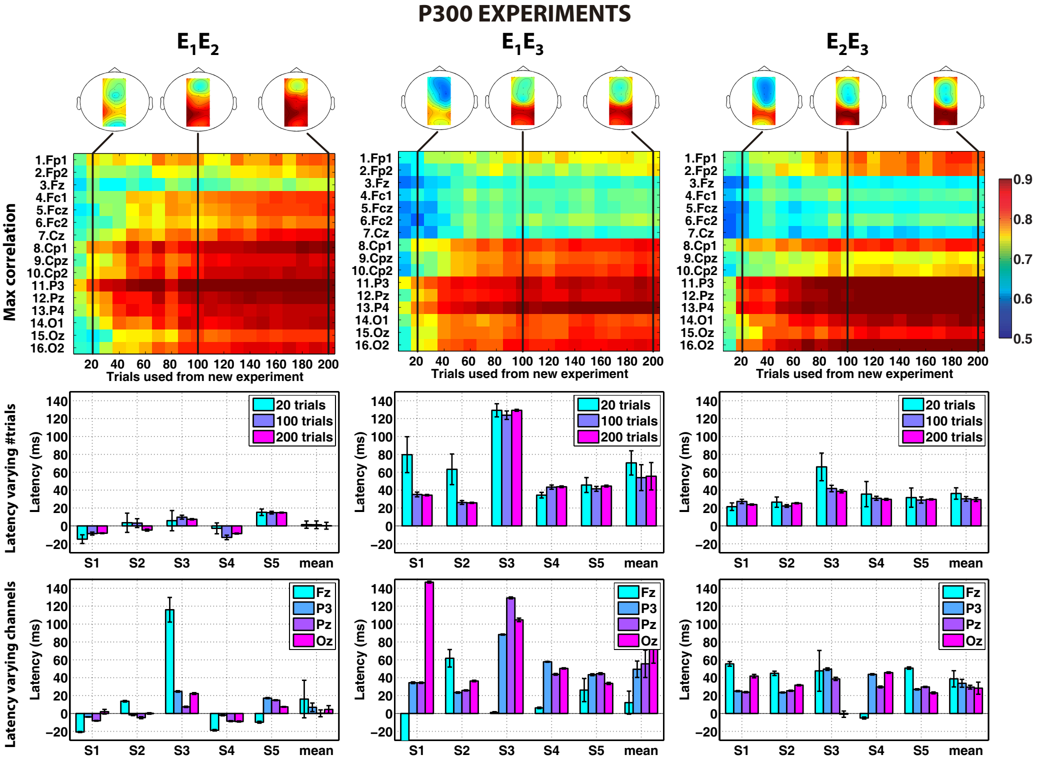


Figure 4.5: Latency results computed for each pair of experiments E_iE_j (from left to right, E_1E_2, E_1E_3, and E_2E_3) for the P300 experiments. For each pair of experiments, the results represent: (Top) Colour encoded image of the maximum correlation values (averaged for all subjects), when varying the number of trials used from E_j^{Tr} (x-axis) and the channel used for the latency computation (y-axis). The topographic interpolation of the correlation values is shown when using 20, 100, and 200 trials from E_j^{Tr} (for the sake of simplicity, the topographic plot is shown only within the field of the recorded channels). (Middle) Mean ± SEM latency estimations (in ms) of each subject, and subject-wise average latency for channel Pz while varying the number of trials used (20, 100 and 200 trials), and (Bottom) Mean ± SEM latency estimations (in ms) of each subject, and subject-wise average latency for 200 trials while varying the channels (Fz, P3, Pz and Oz). Figure is best viewed in colour.

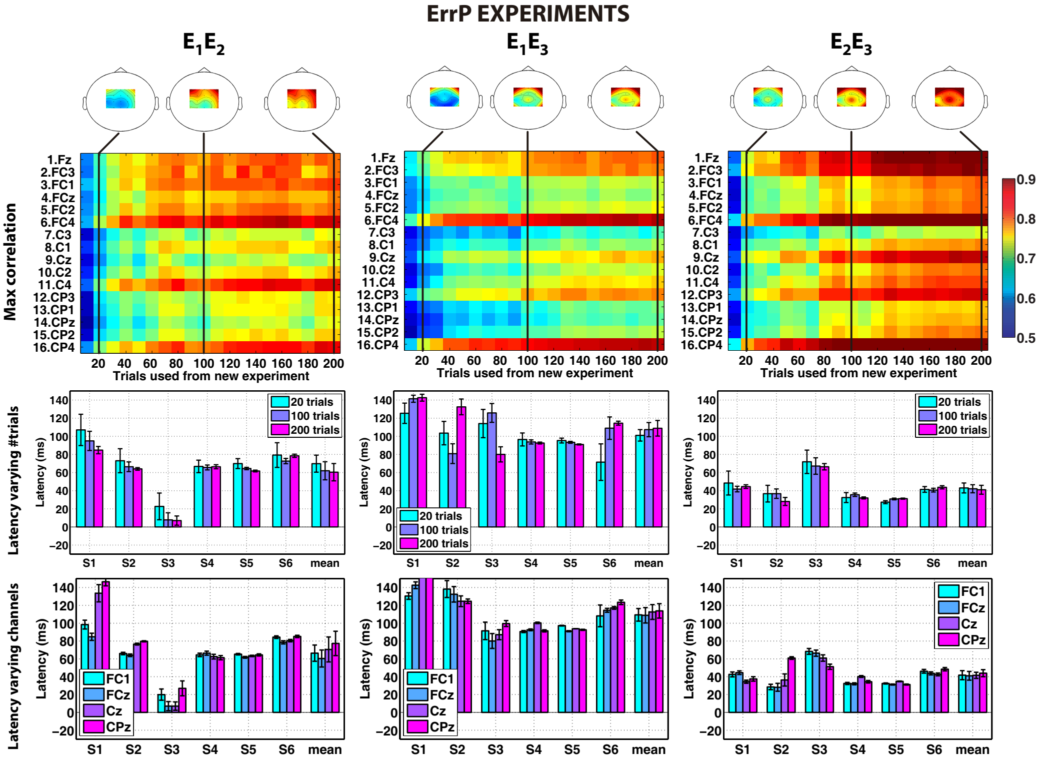


Figure 4.6: Latency results computed for each pair of experiments E_iE_j for the ErrP experiments. (Top) Maximum correlation values. (Middle) Latency estimations for channel FCz while varying the number of trials (20, 100 and 200 trials) and (Bottom) Latency estimations for 200 trials while varying the channels (FC1, FCz, Cz, and CPz).

When we computed the correlation using data from the same experiment, in the case of the P300 signal we obtained correlations above 0.8 when more than 40 trials were used. In contrast, correlations computed in the ErrP protocols required more than 70 trials to exceed this value. Thus both cases needed comparable number of trials to reach similar correlations values as when generalizing among experiments.

Figures 4.5 and 4.6 (Middle) show the latency values of each subject computed for different number of trials in E_j^{Tr} (20, 100 and 200 trials). We show the latency calculation for channels Pz and FCz for the P300 and ErrP experiments respectively, since they had high correlation values and are commonly used for studying this signals [47,50]. For the P300 experiments (Figure 4.5, Middle), the baseline latencies (i.e. computed on the same experiment) after 200 trials from E_i^{Tr} were $-1.17{\pm}14.01$ ms, $-3.90{\pm}7.69$ ms, and $-3.13{\pm}13.87$ ms for experiments 1 to 3 respectively. The latency between E_1 and E_2 using 200 trials was of 0.16 ± 9.36 ms. This agrees with the previous results, where no statistical differences in the latencies were found between these experiments (c.f. Section 4.3.1). For the E_1E_3 and E_2E_3 cases, larger latencies were estimated (on average 55.54 ± 37.51 and 29.45 ± 5.16 ms, respectively). No statistical differences were found in the computed latencies as the number of trials varied ($p > 0.05$ for the three combinations of experiments). Similarly, no significant interactions between the number of trials and repetitions was found ($p > 0.05$). These results suggest that the latency estimation is rather robust to the number of trials used for their computation, and that the specific trials used (i.e. repetition) did not affect the latencies obtained.

For ErrPs (Figure 4.6, Middle), the baseline latencies after 200 trials were 5.40 ± 5.62, 12.96 ± 21.77, and 2.02 ± 4.80 ms for experiments 1 to 3. On the other hand, the latency variations across experiments were larger than those obtained for the P300: 60.42 ± 25.24, 108.85 ± 22.86 and 41.02 ± 12.95 ms for the E_1E_2, E_1E_3, and E_2E_3 pair of experiments. Larger inter-subject variability was also observed. There were statistical differences in the latency computation as the number of trials increased for the E_1E_2 and E_2E_3 cases ($F_{19,95} = 3.329, p = 0.0001$, and $F_{19,95} = 2.249, p = 0.005$, respectively), but not for the E_1E_3 ($p > 0.4$). On the other hand, no significant interactions between number of trials and repetitions were found for any case ($p > 0.05$). This indicates that the latency estimation was robust when estimated using different trials. However, the latency estimation was affected by the number of trials used from E_j.

Figures 4.5 and 4.6 (Bottom) show the latency values of each subject computed for different channels. The number of trials remained fixed to 200. For the P300 experiments (Figure 4.5, Bottom), using frontal channels (e.g. Fz in the plot) for the latency calculation led to different results and higher standard deviations than using parietal channels (e.g. P3 and Pz). Regarding the ErrP experiments (Figure 4.6, Bottom), the latency estimations were more uniform across channels. Nonetheless, higher standard deviations and lower correlation values were obtained when using parietal channels, except for the E_2E_3 case, where similar results were obtained.

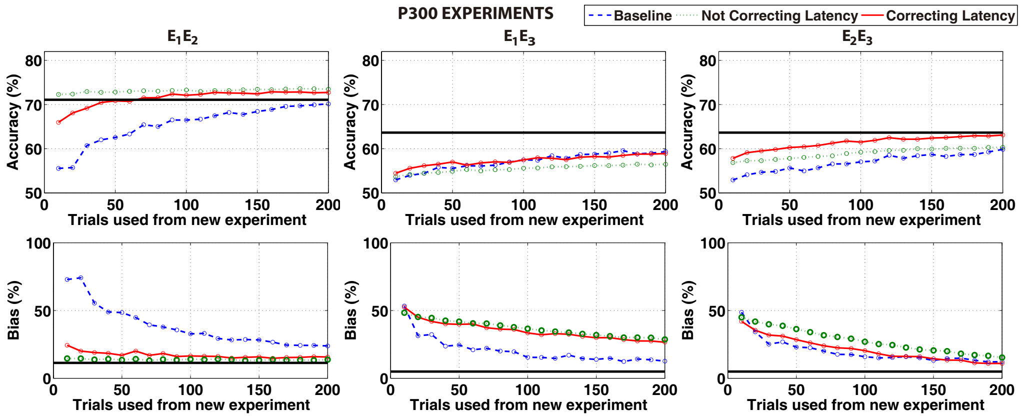


Figure 4.7: (Top) Mean accuracy and (Bottom) classifier bias when correcting the latency from E_1E_2, E_1E_3 and E_2E_3 for the P300 experiments. The x-axis represents the number of trials of the training dataset E_j^{Tr}. Blue-dashed, green-dotted and red-solid lines represent, respectively, the results for the baseline classifier, the classifier trained when not correcting the latency, and the classifier trained when correcting the latency. Horizontal black lines mark the ten-fold cross-validation accuracies of the E_j experiment.

4.3.3 Single-Trial classification of latency-corrected ERPs

P300 potentials

Figure 4.7 shows the mean classifier accuracy and bias for all experiments and tested conditions (see Figure 4.3). In the E_1E_2 case the mean accuracy of the baseline classifier (i.e. trained only with data from the new experiment) increased as more examples were added, reaching 70.15% after 200 trials. In contrast, using data from the previous experiment (E_1) significantly improved (two-tailed paired t-test, $p < 0.0001$) the accuracy, both when correcting the latency (reaching 72.74% after 200 trials) and when not correcting the latency (73.46% after 200 trials). In these cases, only 10 trials from E_2 were enough to improve the accuracies with respect to the baseline classifier. Additionally, these two classifiers had better accuracies than the ten-fold CV with more than 50 trials from E_2. Thus, re-using data from a previous experiment

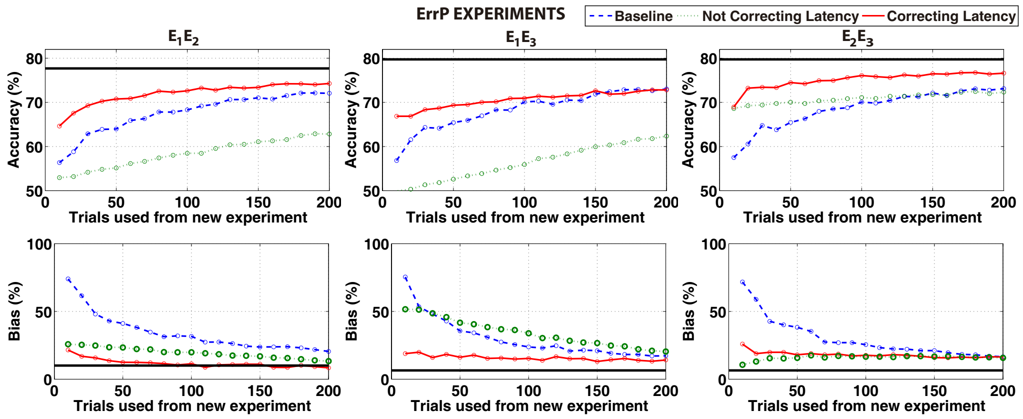


Figure 4.8: (Top) Mean accuracy and (Bottom) classifier bias when correcting the latency from E_1E_2, E_1E_3 and E_2E_3 for the ErrP experiments.

allowed for an improvement both in the classifier accuracy and calibration time. However, at least 90 trials were required for the latency correction method to perform as well as the no correction approach, seemingly due to errors in the latency estimation. The baseline classifier showed large bias when few trials are available. This bias decreased as more trials were added reaching 23.85% after 200 trials. In contrast, re-using data from the previous experiment yielded a low bias since the beginning, both when correcting and not correcting the latency (15.70% and 13.70% after 200 trials, respectively). These values were similar to the ten-fold CV. As with the accuracy, the baseline classifier bias was always significantly worse than the results obtained with the other two classifiers ($p < 0.0001$).

Compared to the previous case, going from E_1 to E_3 resulted in lower accuracies for all types of classifiers (c.f. Figure 4.7 central column), always lower than the CV accuracies. After 200 trials the accuracies were of 59.33%, 58.89% and 56.48% for the baseline, latency-corrected and latency non-corrected classifiers, respectively. When the latency was not corrected, significantly lower accuracies were obtained with more than 100 trials ($p < 0.05$). No significant improvement was found when correcting the latency with respect to the baseline ($p > 0.05$). This suggests a smaller effect of re-using data from the previous experiment. Furthermore, the baseline classifier exhibited significantly lower bias ($p < 0.0001$) than the other classifiers, reaching 12.67% after 200 trials, versus 26.60% and 28.59% when correcting and not correcting the latency, respectively.

In the last case (E_2E_3, c.f. Figure 4.7 right), the latency correction mechanism yielded significantly higher accuracies ($p < 0.05$) than the baseline or no correction approaches (63.09%, 59.85%, and 60.31% after 200 trials, respectively), converging to the accuracy of the 10-fold CV. These differences appeared even when a small number of trials were available. Thus, the use of latency-corrected data allowed for a significant improvement in the accuracies. Moreover, although after 200 trials the bias level was similar for all approaches (around 12%), not correcting the latency yielded significantly higher bias than the other two classifiers in almost all the cases ($p < 0.05$). No significant differences were found between the latency-corrected and the baseline bias.

Error potentials

Results for the error potential protocols are shown in Figure 4.8. In the first case, E_1E_2, the accuracy of the baseline classifier reached 72.05% after using 200 trials for training. The latency-corrected classifier showed better performance of 74.27%, similar to the ten-fold CV accuracy. Notably, the latter always performed significantly better than the baseline (two-tailed paired t-test, $p < 0.0001$). In contrast, the use of previous data without correcting the latency led to significantly lower accuracies than the other classifiers for less than 120 trials ($p < 0.05$), reaching 62.81% after 200 trials. Furthermore, despite its high accuracy, the baseline classifier had significantly higher bias than the other classifiers ($p < 0.0001$), whereas the latency correction yielded to a significantly lower bias than the classifier when not correcting the latency ($p < 0.0001$).

In the second case, E_1E_3, the latency-corrected classifier significantly outperformed the baseline when a small number of trials were used ($p < 0.05$, with less than 100 trials). These classifiers performed similarly after 200 trials (with accuracies of around 73%), without reaching the accuracy of the ten-fold CV. Again, the classifier using not corrected data always performed significantly worse than the others ($p < 0.0001$), with an accuracy of 62.37% after 200 trials. Similarly, the bias of the latency-corrected classifier was significantly lower than the bias of the other two classifiers when less than 160 trials are used ($p < 0.05$).

In the last case (E_2E_3, c.f. Figure 4.8 right), the baseline classifier was always significantly worse than the latency-corrected one ($p < 0.0001$). After 200 trials, the baseline classifier reached a 73.10% of mean accuracy versus a 76.60% when correcting the latency, comparable to the ten-fold CV classifier. The latter classifier was also significantly better than the non-corrected classifier with 20 trials or more ($p < 0.0001$). Finally, the latency-corrected classifier obtained significantly lower biases than those obtained with the baseline classifier for less than 130 trials ($p < 0.05$).

To summarize, apart from one case –generalization from the P300 experiments E_1 to E_3– the latency-correction mechanism improved the classification performance in all the analyzed cases. It allows significantly higher accuracies and/or lower bias than the baseline condition when a small number of trials from the new experiment are available.

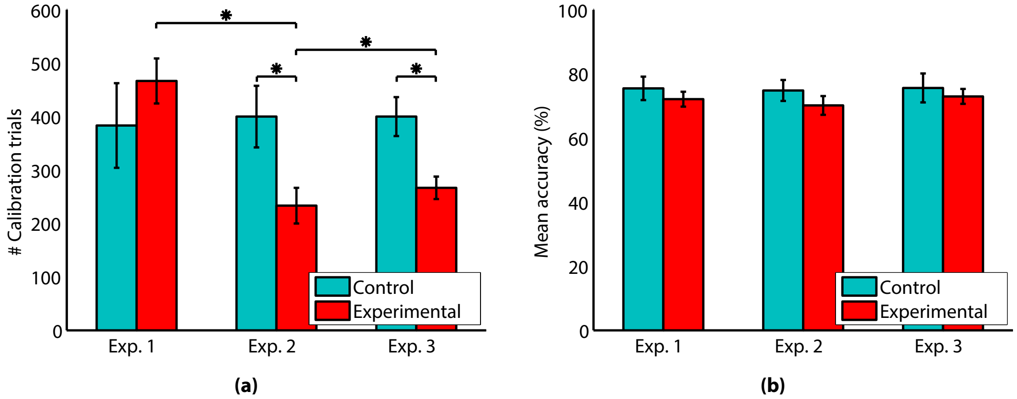


Figure 4.9: (a) Mean ± SEM calibration trials needed for the control group (red) and experimental group (blue), for each experiment. Significant differences ($p < 0.05$) between pairs are marked with a star. (b) Mean classifier accuracy during the online control phase for each group and experiment.

4.3.4 Online results

Figure 4.9a shows the number of calibration trials needed for each group of subjects and experiment. The ANOVA test revealed a significant interaction between the experiment and group ($F_{2,20} = 8.65, p = 0.002$). Post-hoc tests showed that significant differences were found between groups in experiments 2 and 3 (one-tailed unpaired t-tests, $p < 0.05$), and also significant differences within the experimental group between experiments 1 and 2 (one-tailed paired t-test, $p = 0.004$), and between experiments 1 and 3 ($p = 0.004$), see Figure 4.9a. On the other hand, the accuracies was significantly similar for all the experiments and subjects ($p > 0.85$) (see Figure 4.9b). These results indicate that, given data from previous experiments, the latency correction algorithm allowed for a reduction in the calibration time of error-related potentials.

4.4 Discussion

A practical issue in the study of event-related brain activity and its use for BCI applications is the time required to acquire sufficient data to have a reliable model or a usable classifier of the EEG signals. In general, each protocol is treated as a completely new experiment even if they are tapping into similar cognitive processes. Besides the increase in the required time and resources, this also provides little information about how similar responses are across experimental conditions. Using several protocols on two well-studied signals, we showed that the experimental design mainly affected the ERP latencies. Moreover, we proposed a simple, yet powerful mechanism to compensate for these changes allowing to generalize BCI classifiers across experiments using a reduced amount of new data.

Although previous studies have reported modulations on the P300 and ErrP amplitude depending on the target or error probability [47, 50], variations in our protocols did not result in statistically significant amplitude differences across experiments. In contrast, ERP latencies were found to be different across several experiments (c.f. Section 4.3.1). In the P300 experiments, significant variations appeared when changing from experiments 1 or 2 to the third experiment. Interestingly, this experiment had the most complex visual stimuli (a three-dimensional grid), seemingly requiring the subject longer time to evaluate the stimulus. This is supported by neurophysiological studies suggesting that the P300 is related to the stimulus evaluation time [51, 137].

Regarding the error potentials experiments, the latency changes were larger for both peaks (P3 and N4) than for P300 when changing the experimental conditions. The selected protocols were designed so as to have an increased level of complexity both in the number and type of possible actions: changing from two to four possible actions at each state (from E_1 to E_2 and E_3); changing from 1D to 2D (from E_1 to E_2 and E_3); and changing from a simulated to a real device with a wider field of view needed (from E_2 to E_3). Accordingly, increasingly longer latencies were found from protocols E_1 to E_3. However, it should be noticed that part of the measured latencies may be also due to implementation issues (e.g. the time it takes to the robot to actually start moving once the control command has been issued). Nonetheless, the use of a simple technique such as the cross-correlation allowed to significantly remove these latency jitters among experiments, as presented by the ANOVA results after correcting the latencies.

As expected, the latency estimation improves when more data is available. Nonetheless, 100 trials proved to be sufficient to obtain a reliable estimation with correlations higher than 0.8 as seen on Figures 4.5 and 4.6. In this work, the latency was estimated and corrected separately for each channel. However, in principle more reliable estimations –and higher correlation values– should be obtained in those areas where the ERP is generated. This information could be combined to study protocol-dependent spatio-temporal variations of a given brain process.

Focusing on applications of brain-computer interfacing, we propose a simple latency correction mechanism to re-use data from previous experiments when building classifiers for new experiments on a related phenomenon. This yields a reduction in the calibration time as a smaller number of trials is required to achieve similar classification performance (in terms of accuracy and bias) than if a new classifier is built from scratch (c.f. Figures 4.7 and 4.8). Moreover, in those cases where the experimental change has little effect on the ERP latencies, the effect of the latency correction mechanism does not have a negative effect on the classification performance (e.g. moving from P300 experiment E_1 to E_2). In a similar way, Thompson et al. [131] also found that the latency variations among trials but within the same experimental protocol were one of the main problems for the classification performance, and proposed the use of within-experiment latency variations as a predictor of online BCI accuracies. The authors also argued that a brute-force method (i.e. testing a classifier for each possible latency and taking the classifier with maximum accuracy) could be used to estimate these latencies.

The latency variations have been assessed on two different ERPs (P300 and observation error potentials), showing their effect on the single-trial classification. In the future, the generalization of BCI decoders across protocols can be assessed for other ERPs, such as those generated during rapid visual processing [138], or the N2 evoked component [139]. Moreover, additional studies of event-related potentials in controlled and non-controlled applications may yield new findings. The proposed method could be used to elucidate common patterns across conditions, not only in BCI applications but also in neurophysiological studies, e.g. comparing latency variations between error-related activity in choice reaction tasks [42], and in feedback tasks [32].

Furthermore, more sophisticated techniques could be tested to cope with the latency variations such as dynamic time warping [140, 141]. Finally, one disadvantage of the proposed approach is that it relies on the assumption that there are only temporal changes in the ERPs, whereas the spatial contributions remain fixed among experiments. However, this assumption may be wrong. Thus, a more complete approach could be designed by performing a spatio-temporal compensation of the ERP variations.

5 | Shared-control BCI for a two dimensional reaching task using error-related potentials

5.1 Introduction

Chapter 2 presented an alternative BMI paradigm where the user was able to teach a device how to solve a task. Besides the calibration phase needed prior to the control phase (addressed in chapters 3 and 4, and further studied in chapter 6), the bottleneck of the proposed paradigm is the scalability of the system. Due to the reinforcement learning framework, the device needed to explore the entire state space to learn the optimal behavior for each position. Furthermore, although ErrPs provide feedback about the device actions, the amount of information conveyed by them is limited. In particular, as shown in the previous chapters, the decoders do not contain any information about direction or magnitude, and have a non-negligible number of misdetections. As a result, the system presented in chapter 2 required more time to converge as the task space increased its state-action complexity (e.g. the system proposed needed more time to converge in experiments 2 and 3 than in experiment 1, see section 2.3). One way of solving these issues is to further increase the controller intelligence. In this sense, recent approaches started to explore shared-control strategies where the device does not only execute the decoded commands, but is also involved in executing the task [2] (e.g. taking into account the environment while reaching a target or avoiding an obstacle [9, 10]).

In this chapter, we propose the use of a shared-control scheme to greatly reduce the time needed to solve a task in the BMI paradigm proposed in chapter 2, by exploiting the intrinsic constraints of the task being performed. In this work, we present a 2D reaching task of a cursor over a discrete grid of possible targets (similar to the experiments performed in chapter 2 but with greater complexity), using error-related potentials as supervision signals. Under this control scheme, the user has to evaluate whether the cursor is correctly moving towards the goal or is making wrong movements. Based on this signal, the device estimates which is the desired target while moving towards it based on the optimal motion policies for each of the possible goals. The results show that all users were able to reach predefined and self-selected goal locations in around 23 actions (about 19 seconds of EEG signal). The use of this scheme has several interesting advantages with respect to the paradigm presented previously. First, it is possible to learn the entire motor behavior space using much less actions, and at the same time reach the desired target; secondly, it offers a natural way of not only learning motor behaviors, but also choosing among a set of previously stored ones; third, within a shared-control strategy it can be combined with other brain signals to correct decoding failures or recover from wrong or ambiguous decisions of the device.

5.2 Methods

5.2.1 Data Recording

Electroencephalographic (EEG) and electrooculographic (EOG) activity were recorded using a gTec system (3 synchronized gUSBamp amplifiers). For the EEG, 32 electrodes were recorded, distributed according to an extended 10/20 international system (FP1, FP2, F7, F8, F3, F4, T7, T8, C3, C4, P7, P8, P3, P4, O1, O2, AF3, AF4, FC5, FC6, FC1, FC2, CP5, CP6, CP1, CP2, Fz, FCz, Cz, CPz, Pz and Oz), with the ground on FPz and the reference on the left earlobe; for the EOG, 6 monopolar electrodes were recorded (placed above and below each eye, and from the outer canthi of the left and right eyes [142]), with the ground on FPz and the reference on the left mastoid. The EEG and EOG signals were digitized with a sampling frequency of 256 Hz, power-line notch filtered, and band-pass filtered at $[0.5, 10]$ Hz. The EEG was also common-average-reference (CAR) filtered. Additionally, the horizontal, vertical, and radial EOG were computed as in [142] to remove the EOG from the EEG using a regression algorithm [143]. The data acquisition and online processing was developed under a self-made BCI platform.

5.2.2 Experimental Protocol

Four subjects (mean age 26 ± 2 years) participated in the experiments. The participants were seated on a comfortable chair one meter away of a computer

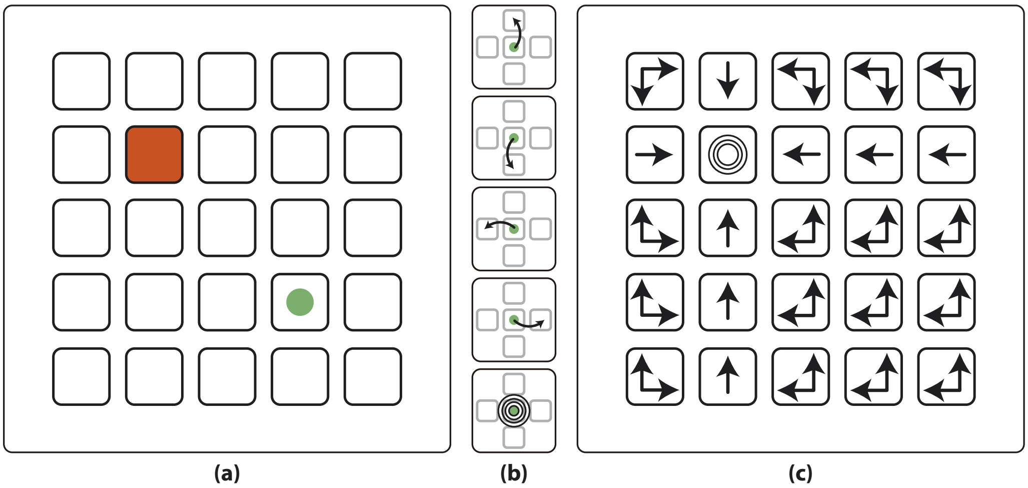


Figure 5.1: (a) Experimental protocol designed. The protocol showed a 5x5 grid with a virtual cursor (green circle) and a goal location (shadowed in red). The up-left-most position is the [1 1] (row and column). (b) The cursor could perform five different actions (from top to bottom, move one position up, down, left or right, or performing a goal-reached action). (c) Correct actions (i.e. optimal policy) from each state for the goal exemplified on (a).

screen that displayed all the information related to the experiment. The experimental protocol is shown in Fig. 5.1. The protocol consisted of a 5x5 squares grid, a virtual cursor (green circle), and a goal location shadowed in red. The cursor performed five different instantaneous actions: move one position left, right, up or down; and a goal-reached action, represented as concentric blue circumferences (see Fig. 5.1b). The time between two actions (inter-action interval) was random within the range $[3, 3.5]$ s. The role of the subjects was to assess the cursor actions as correct when the cursor performed (*i*) a movement towards the goal position, or (*ii*) a goal-reached action over the goal position; and as incorrect otherwise (see Fig. 5.1c). The users were instructed not to move their eyes during the cursor actions, and to restrain blinks only to the resting periods.

The experiment consisted of two phases: the training phase used to calibrate a classifier able to distinguish between correct and error user's assessments; and the control phase, where the user controlled the cursor to a goal position. During the control phase, two different groups of goal locations were tested: (*i*) the first group was fixed for all the subjects, and consisted of five predefined goals and initial cursor positions (see table 5.1); (*ii*) for the second group, each user was asked to freely choose five different initial cursor

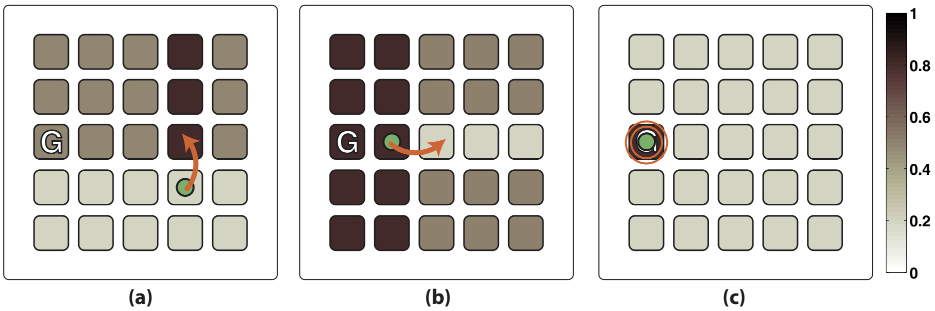


Figure 5.2: Likelihoods of each policy π_i after performing different actions: (a) correct movement with $p(c_t = 1|x_t) = 0.2$ (b) incorrect movement $p(c_t = 1|x_t) = 0.8$ (c) or a goal-reached action $p(c_t = 1|x_t) = 0.2$. The goal position is marked with a capital G.

positions and goals to reach. During this group of goals, the goal position was not shadowed in red, since it was the user who chose it.

Table 5.1: Initial and goal positions for the fixed group of goals

	Run 1	**Run 2**	**Run 3**	**Run 4**	**Run 5**
Initial position	[1 1]	[2 3]	[1 2]	[1 4]	[3 3]
Goal position	[5 5]	[3 1]	[3 3]	[4 1]	[3 3]

5.2.3 Calibration of error potentials

For the calibration of the error potential detection, a training phase was first executed to acquire sufficient examples of potentials while the user assessed the device actions. During this phase, the virtual cursor performed random actions, with a 20% of probability of performing an erroneous action. This phase lasted for 30 minutes, acquiring around 80 correct and 320 erroneous trials.

Once the training data was acquired, features were extracted from eight fronto-central channels (Fz, FC1, FCz, FC2, C1, Cz, C2, and CPz) within a time window of $[200, 800]$ ms downsampled to 64 Hz, forming a vector of 312 features. The feature vectors of all trials were normalized and decorrelated using PCA, retaining those that explained 95% of the variance. Finally, a regularized linear discriminant (LDA) [124] classifier was trained using the retained features. The classifier output has the form $y(\mathbf{x}) = \mathbf{w}'\mathbf{x} + b$, where $y(\mathbf{x}) \geq 0$ is classified as a correct assessment (class 1), and $y(\mathbf{x}) < 0$ as

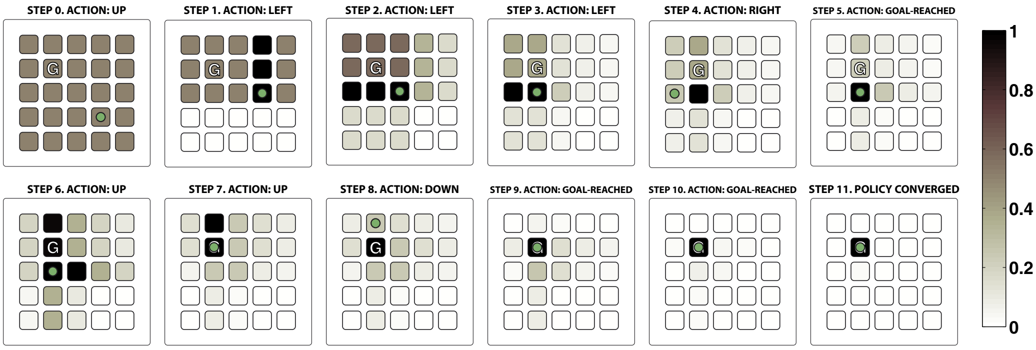


Figure 5.3: Device control simulation for the goal shown in Fig. 5.1a ([2 2], marked with letter G). At each step t, the device was on a state s_t and executed an action a_t (shown over each individual plot). Each state i is colour-encoded as the probability of the goal being at that state i, $p(\pi_i^*|a_{t...1}, s_{t...1})$. The classifier was ideal, with its output $p(c = 1|x) = 0.8$ for correct actions, and $p(c = 1|x) = 0.2$ for incorrect actions. Initially, all the probabilities were equiprobable. While more steps were executed several policies were discarded, whereas others increased their probability. After 11 steps, the policy convergence criterion was reached, and the device learned the goal position.

an error assessment (class 0). This output $y(\mathbf{x})$ was transformed into the probability that a trial $\mathbf{x}$ belonged to the correct class, $p(c = 1|\mathbf{x}) = \frac{1}{1+e^{-y(\mathbf{x})}}$ [144].

5.2.4 Shared-control for a reaching task

This section describes the proposed shared-control strategy with error potentials. Although these potentials provide feedback about the device actions, the amount of information conveyed by them is limited. In particular, the decoder of Section 5.2.3 does not contain information about direction or magnitude. Furthermore, it has a non-negligible number of misdetections. Therefore, the proposed shared control uses memory to accumulate evidence about possible goals while executing a trajectory. The proposed approach consists of two phases. The first one computes offline optimal trajectories (i.e. policies) for each potential target, while the second one ranks them during execution using error potentials elicited for wrong actions.

Offline policies computation

The offline phase was used for the computation of the optimal motion policies (see Fig. 5.1c) for all possible goal locations. Let $\pi_i^*(s)$ be the optimal

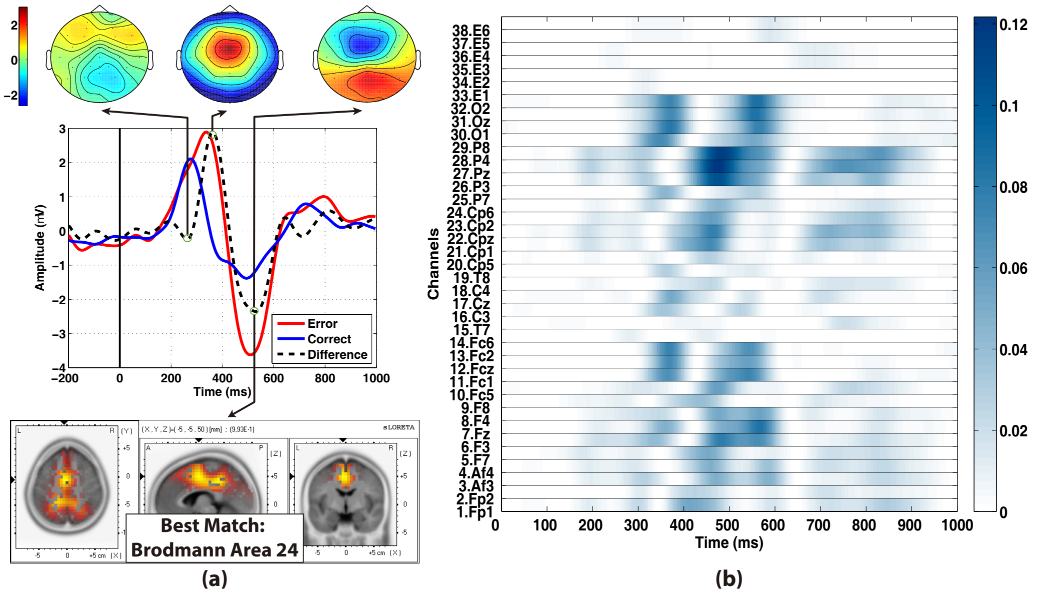


Figure 5.4: (a) Action-locked averages from channel FCz (error, correct and difference in red, blue and black respectively), together with the color-encoded topographic interpolation of the three most prominent peaks of the difference average. Time 0 ms indicates when the cursor executed an action. The source localization for the most prominent negative peak of the difference average is also shown. (b) Color encoded r^2 discriminability test for each time point (x-axis) and channel (y-axis), including the 6 monopolar EOG channels recorded (denoted as E1 to E6).

policy for reaching the goal position i from the state s, whose output is the best action to compute in s (see Fig. 5.1c for a representation of the optimal policy when the goal location was [2 2]). The optimal policies can be represented with their associated Q-values $Q_i^*(s,a)$, which represents the value of executing the action a in state s when the goal location is i. These Q-values were converted into probabilities, following a soft-max normalization [61]:

$$\hat{Q}_i^*(s,a) = \frac{e^{Q_i^*(s,a)/\tau}}{\sum_b e^{Q_i^*(s,b)/\tau}}, \tag{5.1}$$

where τ is denoted the temperature (fixed to $\tau = 0.3$ for the current experiment). This parameter served as a degree of reliability of the observed information (classifier output). For the designed protocol, the Q-values for all the possible goal locations can be computed offline prior to the control phase using the Q-learning reinforcement learning algorithm [61].

Online control

The online cursor control was performed by updating the probabilities of having the goal location on a state i $p(\pi_i^*)$, which where initially equiprobable (see Figure 5.3, step 0). At each time step t, the device performed an action a_t from state s_t. This action was assessed by the user, and the EEG classified following the procedure described in subsection 5.2.3, obtaining $p(c = 1|x, a_t, s_t)$. This probability was used to update $p(\pi_i^*)$ for each goal location i (see Figure 5.3, steps 1 to 10). Finally, the execution finished when a probability $p(\pi_i^*)$ reached a convergence criterion, $p_c = 0.9$ (see Figure 5.3, step 11). At time t, the probability of each policy i being the correct one given an action a_t executed in a state s_t is:

$$p(\pi_i^*|(a, s, \mathbf{x})_{1\ldots t}) \propto p(a_t|\pi_i^*, (s, \mathbf{x})_t) \cdot p(\pi_i^*|(a, s, \mathbf{x})_{1\ldots t-1}), \tag{5.2}$$

where the likelihood is computed as:

$$p(a_t|\pi_i^*, (s, \mathbf{x})_t) = p(c = 1|\mathbf{x}) \cdot \hat{Q}_i^*(s_t, a_t) + p(c = 0|\mathbf{x}) \cdot (1 - \hat{Q}_i^*(s_t, a_t)), \tag{5.3}$$

Notice that the first term of the likelihood represents how we should increase the policy π_i^* if the user's assessment was correct, while the second term penalized the policy π_i^* weighted by the probability of having and incorrect user's assessment (see Figure 5.2). Fig. 5.3 shows an example of the device control when reaching the goal shown in Fig. 5.1a.

Table 5.2: Results of the reaching task for the fixed goals

	s1	**s2**	**s3**	**s4**	**mean ± std**
♯ Targets reached (out of 5)	5	5	5	5	5 ± 0
♯ Actions	16 ± 2	43 ± 9	23 ± 12	17 ± 5	25 ± 13
Net time (s)	12.96 ± 1.91	34.56 ± 7.10	18.40 ± 9.73	13.60 ± 4.38	19.88 ± 10.75
Total time (s)	52.65 ± 7.76	140.40 ± 28.83	74.75 ± 39.54	55.25 ± 17.80	80.76 ± 73.68
Accuracy correct / error (%)	82.46 / 79.17	60.98 / 78.49	82.43 / 63.41	71.19 / 76.92	74.27 ± 10.32 / 74.50 ± 7.45

Table 5.3: Results of the reaching task for the freely chosen goals

	s1	**s2**	**s3**	**s4**	**mean ± std**
♯ Targets reached (out of 5)	5	5	5	5	5 ± 0
♯ Actions	23 ± 9	21 ± 7	16 ± 6	23 ± 11	21 ± 8
Net time (s)	18.08 ± 7.37	16.48 ± 5.84	12.96 ± 5.10	18.08 ± 8.44	16.40 ± 6.61
Total time (s)	73.45 ± 29.93	66.95 ± 23.73	52.65 ± 20.73	73.45 ± 34.29	66.63 ± 26.85
Accuracy correct / error (%)	75.00 / 72.97	75.00 / 83.87	83.93 / 84.00	70.37 / 76.19	76.08 ± 5.67 / 79.26 ± 5.56

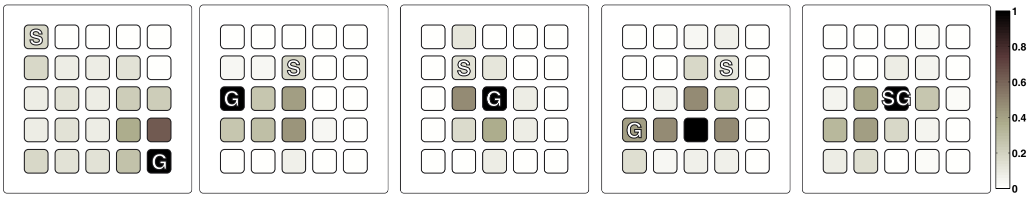


Figure 5.5: States visited by all the subjects, for each of the five runs executed with the fixed goals (from left to right, runs 1 to 5). Darker colors indicate more visited states. The range was normalized from 0 to 1 according to the most visited state for each run. The initial and goal positions are marked with an S and a G respectively.

5.3 Results

5.3.1 Electrophysiology analysis

Fig. 5.4a shows the error, correct and difference grand averaged potentials (error minus correct averages) in channel FCz, averaged for all the subjects. The difference grand average was characterized by three components: a negative deflection at around 250 ms, a positive deflection at around 400 ms, and a another negative component at around 500 ms. The topographic interpolations of the two broader peaks of the difference average showed that the signals were generated mostly in fronto-central channels. These results were in agreement with previous studies using error potentials [46]. The sLoreta source localization [122] was applied to the most prominent negativity of the difference average, confirming that the signals were generated in the anterior cingulate cortex (ACC, Brodmann area 24), which is the area thought to be the main generator of error-processing brain activity [46]. Fig. 5.4b shows the r^2 discriminability test of the signals obtained. The test revealed two main zones of discriminability between correct and error signals, at around 350 and 500 ms, agreeing with the two broader peaks appearing in the difference average. Furthermore, no discriminable information was found on the EOG channels (E1 to E6 in Fig. 5.4b), thus indicating that no EOG activity was contaminating the recorded potentials, and that the EOG was not involved in the device control.

5.3.2 Control phase analysis

For each group of goals (fixed and freely-chosen), five metrics were evaluated: (*i*) Number of goals reached; (*ii*) number of actions needed to reach the goal, (*iii*) EEG seconds needed to reach the goal (net time); (*iv*) total time needed to reach the goal; and (*v*) classifier accuracy, measured as the percentage of detection of correct and erroneous signals.

Table 5.2 shows the results for each subject (averaged for goals). The first result was that the device was able to reach all the target location, independently of their relative location. The users needed 25 ± 13 actions to reach the goal. With the proposed inter-action interval (around 3.25 s), the total time needed to reach the goals was of 80.76 ± 73.68 s. Nonetheless, the net time (i.e., the seconds of EEG signal used for decoding) was of 19.88 ± 10.75 s. Note that the difference between the net and total times was the seconds belonging to inter-action intervals, which could be easily removed. The mean classifier accuracy across subjects was of 74.27 ± 8.94 and 74.50 ± 6.45 for correct and erroneous actions respectively. As expected, there was a significant negative correlation between the classifier mean accuracy and the time needed to reach the task ($r = -0.47, p = 0.038$). Notice that a random walk strategy that selected random actions at each step would require around 150 actions for a 5x5 grid size. Fig. 5.5 shows the states visited for the five runs performed. As can be seen, not all the states needed to be visited to reach the goal. For instance, during run 3, mostly all the central states were visited, whereas the peripheral states were not. This is an interesting property of the proposed control scheme, since it allows for a better scalability of the system, due to the fact that each step of the trajectory provides information for multiple goals simultaneously. As a result, the percentage of visited states would decrease for larger state spaces.

Table 5.3 shows the results for each subject. As with the fixed goals, all the targets were correctly reached by the device. The number of actions needed to reach the goal was 21 ± 8, with a total time of 66.63 ± 26.85 s, and a net time of 16.40 ± 6.61 s, and thus slightly better than the results with fixed goals. However, these differences were not significant (paired t-test, $t_{19} = 1.05, p = 0.31$). Regarding the classifier accuracy, the detection rate was better in this case: 76.08 ± 5.67 and 79.26 ± 5.56 for correct and erroneous actions. The error accuracy was significantly better than the one obtained for the fixed goals, (paired t-test, $t_{19} = -2.27, p = 0.04$), but the correct accuracy was not (paired t-test, $t_{19} = -1.12, p = 0.28$). The fact of this slight increase of accuracy could be due to a protocol habituation, since the users always executed the free goals after the fixed ones. Again, there was a significant negative correlation between the mean accuracy and the time needed to reach the task ($r = -0.79, p = 3 \cdot 10^{-5}$).

5.4 Conclusions and future work

In this work, we have presented a shared-control BCI for a 2D virtual cursor based on error potentials. Most of the intelligence of the system was moved to the device side by comparing optimal trajectories with the executed one based on the detected error potentials (actions perceived as wrong by the user). The proposed approach compensates the low information transfer rate of error potentials by exploiting the structure of trajectories. The experimental results showed that, for a 5x5 grid, the system reached any goal after only 23 actions on average (less than a minute in our protocol). Indeed, this was a large improvement compared to the results obtained in chapter 2. For experiments 2 and 3 (see Figure 2.2), the total number of possible states and actions was 13 and 22 respectively; whereas for the proposed approach was of 25 states and 105 actions. Furthermore, in the previous approach, the subjects needed 100 actions to learn around an 80% of the optimal motor behavior, whereas in this chapter the subjects needed a 75% less actions to learn the entire optimal motor behavior and at the same time reach (stop at) the target.

The proposed shared-control BCI might scale to more complex scenarios because: *i*) it is not necessary to explore every single trajectory or potential goal; and *ii*) the user only has to monitor the device actions and evaluate if they are right or wrong. The authors are currently applying this BCI approach to real devices (e.g. mobile robots or manipulators) with larger and continuous state-action spaces.

6 Zero-calibration BMIs for sequential tasks using error-related potentials

6.1 Introduction

Previous chapters have shown how it is possible to teach a device how to solve a task following the proposed teaching paradigm. As for all non-invasive BMIs, this paradigm needed a decoder trained after a calibration phase that translated the error-related potentials into feedback for a device. Despite we have seen how this calibration phase can be reduced thanks to re-using data from previous experiments, several examples from the new task were always needed to train the decoder. Furthermore, these examples needed to be acquired in open loop, that is, where the user is not performing the task. This is a key problem for the paradigm developed throughout this thesis, since the time needed to learn the decoder adds to the time needed to learn the task as such, greatly delaying the final device operation and hindering the deployment of the BMI out of the lab.

This chapter addresses the problem of removing the calibration phase in the BMI paradigm proposed in chapter 2 and extended in chapter 5. Here, we show how it is possible to fuse both the calibration and the task learning phases in a single closed-loop phase that learns the classifier and the task at the same time and in an unsupervised manner. The main idea is the same as the one presented in the previous chapter: to exploit the task constraints. The previous chapter showed how these constraints allowed to estimate all the possible optimal policies (one for each target) while a recursive filter estimated which target was the correct one. Similarly, it is possible to compute all the possible signal labels (i.e. whether the current EEG is generated from a wrong or correct assessment) associated to each optimal policy, and use a recursive filter that learns which labels best fit the

EEG signals.

Related work on this domain is very limited. This idea was addressed in a similar way by Kindermans et al. for P300 signals by exploiting the fact that these ERPs usually require of multiple visual stimulations [74, 77]. Thus, a recursive filter was able to unsupervisedly estimate the optimal labeling for the signals. Furthermore, it was possible to further increase the convergence speed by using prior information from a pool of subjects. In invasive BCIs, Orsborn et al. [145] learned from scratch in closed loop a decoder for known targets using pre-defined policies to each target. However, the approach needed for a warm-up period of around 15 minutes.

The main contribution of this chapter is a method that simultaneously calibrates the feedback decoder and executes in closed loop a task only known by the user. Our method exploit the described task constraints, namely optimal policies, to hallucinate virtual labels for the feedback signals. Using these labels, it is possible to compute the expected classification rate of the decoder of the task. Since the right task will assign the right labels to the EEG signals, the expected classification rate is also a good measure to identify the user's intended task. Furthermore, we show that it is possible to use model-based planning based on the uncertainty about the task and the feedback signals to explore the space efficiently while learning. The method has been evaluated performing online experiments where four users directly controlled an agent on a 2D grid world to reach a target. The results show that the proposed method is able to learn good feedback models and solve the task efficiently without any calibration. Offline experiments show that our unsupervised trained decoder has the same accuracy as a standard one and illustrate the benefits of our active strategy.

6.2 BCI-feedback Based Control Without Explicit Calibration

6.2.1 Experimental protocol

The experimental protocol followed the same reaching task as the one described in the previous chapter (see subsection 5.2.2), where a device was placed in a 5x5 grid world and whose objective was to reach a target position. The user goal was to assess the device actions (thus generating error-related potentials) and teach it to reach the desired position (see Figure 5.1).

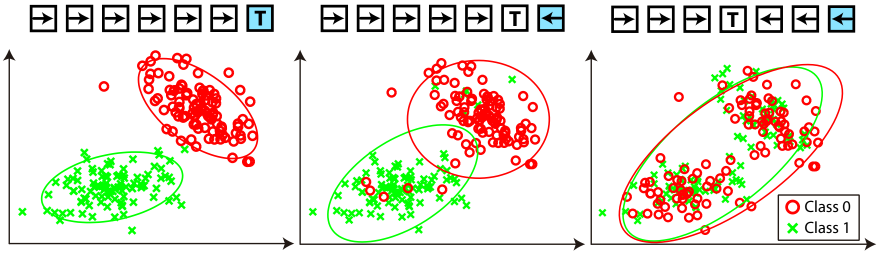


Figure 6.1: Task labels for a 1-D grid world. For the represented example, the arrows indicate for each state what action should elicit a positive feedback to reach the target position shadowed in blue (i.e., the optimal policies). 2D Gaussian distributions of binary feedback signals for three possible targets are shown below. While for the correct target the distributions shows a large separability (Left), the overlaps increases as the believed target moves away from the real one (Middle, Right).

6.2.2 Simultaneous Estimation of Task and Signal Model

In a common BCI scenario, the EEG signals are usually trained in open loop, where the user has no control whatsoever over the device (see for instance subsection 5.2.3). Whenever the calibration is done, the user performs the closed loop experiment, where it controls (or teaches) the device. In this chapter, we faced the question of whether it is possible to estimate at the same time the task being performed (reach a target position) and the signal model (binary classifier trained with ErrP signals).

The main idea of this work is depicted in Fig. 6.1 for a toy 1D example with 7 possibles targets to reach (i.e. 7 possible tasks). The user wants the device to reach the right-most state. However, neither the target nor the signal model are known. The feedback signals (ErrPs) are generated as a response to the execution of an action a in state s according to the true unknown task the user wants to solve. The key point is that these signals are generated from an underlying model that for binary signals has two different classes. Given sufficient feedback signals, it is possible to build the underlying distributions for each possible target. Only the right task will provide the right meanings (or labels) to each of the feedback signals (Fig. 6.1 Left), while the other tasks will gradually mix both classes as the task gradually differs more to the original task (Fig. 6.1 Middle-Right), up to the point of almost mirroring the labels when the target is mirrored. If we extrapolate this to the proposed protocol, it is clear to see that there are 25 possible tasks (one for each possible target). In the remainder of this section we show how this property can be exploited to estimate the task and the model generating the feedback signals.

Let $e_i \in \mathbf{R}^n$ be the EEG measurements e obtained after the execution of action a_i in state s_i. The meaning (or label) $z_i \in \{c, w\}$ of each feedback signal belongs to one of two classes (correct or wrong). Following the literature [124] and the previous chapters, the EEG signals are modeled using independent multivariate normal distributions for each class, $\mathcal{N}(\mu_c, \Sigma_c), \mathcal{N}(\mu_w, \Sigma_w)$. Let θ be the set of parameters $\{\mu_c, \Sigma_c, \mu_w, \Sigma_w\}$.

Regarding the tasks, the system has access to a set of task hypotheses $\xi_1, \dots, \xi_T$ which includes the task the user wants to solve[1]. We do not make any particular assumption on how the task is represented given that for each particular task ξ we are able to compute a policy π which represents the probability of choosing a given action a in state s, $\pi_\xi(s, a) = p(a|s, \xi)$. As mentioned above, these are the policies that, conditioned on the task, provide meanings to the feedback signals of an action-state pair. (e.g. in the proposed reaching task, progressing towards the goal will generate correct answers while moving apart from it will generate wrong ones).

Our goal is to learn which task $\hat{\xi}$ the user wants to solve based on the assessment of the user extracted from EEG measurements collected while executing actions. Thus collected data are in the form $\{(e_i, s_i, a_i),\ i = 1, \dots, N\}$, this is, a sequence of states, actions and teaching signals triplets. Following the discussion of Fig. 6.1, a straightforward option to estimate the task ξ is to measure the coherence of the signal model for each possible task using the virtual meanings provided by the target policy. In other words, the best (ξ, θ) pair should provide the lowest predictive error (perr) on the observed signals $p(e|s, a, \xi, \theta)$. One possible way of solving this problem is to maximize the expected predictive classification rate:

$$\hat{\xi}, \hat{\theta} \quad = \quad argmax_{\xi,\theta} E_e \left(\delta(z(s, a, \xi), z(e, \theta)) \right) \tag{6.1}$$

where $\delta()$ is an indicator function, $z(s, a, \xi)$ is the label (wrong or correct) corresponding to the execution of action a in state s under task ξ and $z(e, \theta_\xi)$ is the label provided by the Gaussian classifier with parameters θ_ξ. The expected predictive error can be explicitly written dependent on the task and decoder model:

$$E_e \left(\delta(z(s, a, \xi), z(e, \theta)) \right) \quad = \quad \sum_{l=c,w} p(z = l|s, a, \xi) p(z = l|e, \theta) \tag{6.2}$$

where $p(z = l|s, a, \xi)$ represents the probability of the user assigning label l when assessing task ξ. We add a noise term to cope with those situations

[1] If this is not the case, the system will find the most suitable task.

were the user assessment may be wrong. The model is then

$$p(z = l|s, a, \xi) = \begin{cases} 1 - \alpha, & a = \operatorname{argmax}_a \pi_\xi(s, a) \\ \alpha, & \text{otherwise} \end{cases} \tag{6.3}$$

with α modeling error rate of the user. The term $p(z = l|e, \theta_\xi)$ is just the probability of the meaning under the Gaussian model provided by θ_ξ

$$p(z = l|e, \theta) = \frac{p(e|z = l, \theta)p(z = l)}{\sum_{k=c,w} p(e|z = k, \theta)p(z = k)} = \frac{\mathcal{N}(e|\mu_l, \Sigma_l)p(z = l)}{\sum_{k=c,w} \mathcal{N}(e|\mu_k, \Sigma_k)p(z = k)} \tag{6.4}$$

Note that the optimization process has be factored using the fact that given the task, the estimation of θ under the Gaussian model is trivial. It basically requires to compute the maximum-likelihood estimate θ_ξ^{ML} for each target ξ. Using the labels of target ξ, the estimation of θ under the Gaussian model described above simply accounts for computing the posterior mean μ_z and covariance Σ_z. In order to avoid numerical problems when estimating the covariance for a low number of examples, a regularization term was applied to penalize very large and very small eigenvalues [146]:

$$\Sigma_z = (1 - \lambda)\Sigma_z + \lambda \frac{trace(\Sigma_z)}{n} \mathbf{I}_n, \tag{6.5}$$

with n the feature dimension, $\mathbf{I}_n$ the identity matrix of size n, and $\lambda = 0.5$ the regularization term.

The second part of the optimization requires to estimate the expected classification rate to select the best target. Possible solutions are use cross-validation or bootstrapping methods to estimate the classification rate using the available data up to time i. However, for small amounts of data, these methods result in estimates with high variance (see also the discussion in [147] about the variance of cross-validation estimators) and have a high computational cost.

Alternatively, we propose another approximation of the expected estimation error: the Bhattacharyya coefficient. This coefficient has been related to the classification error of Gaussian models [148]. Although there is no analytical relation between the coefficient and the classification rate, it is possible to derive bounds and good empirical approximations [149]. Under the Gaussian model, the Bhattacharyya coefficient ρ between the distributions of the signals, associated to each meaning, $\mathcal{N}(\mu_c, \Sigma_c), \mathcal{N}(\mu_w, \Sigma_w)$, is simply $\rho = e^{-D_B}$ where D_B is the Bhattacharyya distance:

$$D_B(\theta) = \frac{1}{8}(\mu_c - \mu_w)^t(\Sigma_c + \Sigma_w)^{-1}(\mu_c - \mu_w) + \frac{1}{2} ln \left(\frac{det(\Sigma_c + \Sigma_w)}{\sqrt{det \Sigma_c \Sigma_w}} \right). \tag{6.6}$$

6.2.3 Estimation of Task and Online Re-Estimation of Signal Model

The use of the Bhattacharyya coefficient provides a simple and efficient way to estimate the target when the number of examples is small and avoids the problems of cross-validation. However, when changing to a new target, the signal model does not change and does not have to be re-learned from scratch. Indeed, once the system has correctly identified a task, it has access to correctly labeled data and so it is possible to train a classifier as in usual BCI calibration approaches. To achieve this, we factorize the joint distribution

$$\begin{aligned} p(\xi, \theta \mid D_i) &\propto p(\xi \mid D_i) p(\theta \mid \xi, D_i) \\ &= p(e_i \mid \xi, (s,a)_i, D_{i-1}) p(\xi \mid D_{i-1}) p(\theta \mid \xi, D_i), \end{aligned} \tag{6.7}$$

where D_i contains the triplets (e_i, s_i, a_i) up to time i associated to its labels. The factorization makes explicit that given the target the distribution $p(\theta \mid \xi, D_i)$ can be easily evaluated using the labels of each target. We approximate this posterior using the maximum likelihood point estimate $\hat{\theta}^{ML}_{\xi}$ per target. For the term $p(\xi \mid D_i)$, we use a recursive Bayes filter

$$\begin{aligned} p(\xi = t, \mid D_i) &\propto p(e_i \mid \xi = t, (s,a)_i, D_{i-1}) p(t \mid D_{i-1}) \\ &\approx p(e_k \mid \hat{\theta}^{ML}_{\xi = t}) p(\xi = t \mid D_{k-1}). \end{aligned} \tag{6.8}$$

Notice that we are keeping a different symbol model $\hat{\theta}^{ML}_{D^{\xi_t}_{i-1}}$ for each possible target ξ_t, the maximum likelihood estimation needs to be done in relation to a dataset $D^{\xi_t}_{i-1}$ corresponding to the expected labels of target ξ up to time $i - 1$. Whenever a task is identified, its labels are transferred to all the triplets of the other tasks to correct the prior for the next step with the right labels. This scheme performs a long term adaptation of θ to accommodate slight variations of EEG such as non-stationarities or variations induced by the task.

6.2.4 Action planning

The previous algorithm is able to acquire the needed knowledge but it is not explicit on the process it uses to acquire the data. The goal of the device is to fulfill the task desired by the user but, as it has to simultaneously estimate which is the task, it has also to explore regions that allow to disambiguate among different tasks. There are several efficient model-based reinforcement learning exploration methods that besides using the task reward function to plan for actions, add an additional exploration bonus for states that might provide more learning gains. Several theoretical results show that these approaches allow to learn tasks efficiently [150, 151]. We define an uncertainty measure and use model-based planning to select sequences of actions that guide the agent to states that better identify the desired task.

The proposed planning method is based on reducing the expected prediction error of the label corresponding to each brain signal. If we choose an action in a given state, given each possible task, we can predict to receive a correct or wrong label. Such label is linked to a signal generation model which differ from task to task, and to the optimal action at that particular state. A state-action where the optimal actions and the signal model is the same for all the hypothesis will be less informative that any other state-action where either actions or models differ. Thus, our measure of global uncertainty $U(s,a)$ will be higher when, for a given state-action there is an high incongruity between either actions or signal models. For this we compute a similarity matrix S where each element $S_{ij}(s,a)$ corresponds to the similarity of the distributions of the signals of the expected label according to tasks i and j if action a is performed in state s. In order to improve computation efficiency we do not rely on a precise metric between distribution and only consider the similarity between the mean of the distribution (empirical tests did no show a significant impact on the results). The final uncertainty value $U(s,a)$ can be computed as the sum of the upper diagonal of the similarity matrix.

This measure is then used as a classical exploration bonus method by summing the task reward and the uncertainty measure. As the world dynamics is known we can plan actions that maximize the surrogate reward function $R(s,a) = (1-\beta)R_{task}(s,a) + \beta U(s,a)$. We can, for instance, use value iteration and then follow the optimal policy. The reward function R includes a parameter β that provides a gain schedule of exploration versus exploitation. We can either tune this parameter or switch to a pure exploitation approach after reaching the desired confidence level on the task model. Interestingly the same approach can be used either in the case of known or unknown signal models. As the uncertainty function combines task and symbol uncertainty, when former is known, the latter becomes the sole source of ambiguity.

6.2.5 Methods for the online experiments

The objective of the online evaluation was to determine whether a user was able to teach the virtual device how to reach a target while learning at the same time the classifier. For this purpose, we followed the algorithm presented in the previous subsections. For the executed experiments, EEG and EOG signals were recorded following the same configuration as in the previous chapter (see subsection 5.2.1). Four subjects (aged between 25 and 28) performed the online experiments. Each subject completed 5 runs of learning from scratch how to reach a randomly chosen target. A run was stopped whenever a task model had more than 99% of the probability of being the correct one, according to Eq. 6.8.

During the online execution of the experiments, the device performed actions while the user assessed them. After each action, the device updated its knowledge about the environment and the task following the proposed algorithm (section 6.2.2), and using several features from the user's EEG. These features were extracted from two fronto-central channels (FCz and Cz) within a time window of $[200, 700]$ ms (being 0 ms the action onset of the agent) and downsampled to 32 Hz, leading to a vector of length 34.

6.2.6 Methods for the offline experiments

Additionally to the online experiments, further offline experiments were executed. For this analysis, a larger dataset was used to ensure that the results obtained were statistically significant. Specifically, we used the datasets acquired from ten subjects during the experiments described in chapter 3, namely OT1 and OT2. For each subject and each dataset, we simulated twenty runs of 400 steps following the control task by sampling a signal from the data.

The objectives of the offline analysis were: (*i*) study the impact of the exploration approach proposed in section 6.2.4; (*ii*) evaluate whether the classifier learned during the first phase of the algorithm (section 6.2.2)) has similar decoding capabilities to classifiers trained with standard calibration approaches; (*iii*) the advantages of using a classifier rather than the Bhattacharyya coefficient during the second phase of the algorithm; and (*iv*) the number of tasks (targets) that can be learned depending on the dataset tenfold accuracy.

6.3 Results

6.3.1 Online experiments

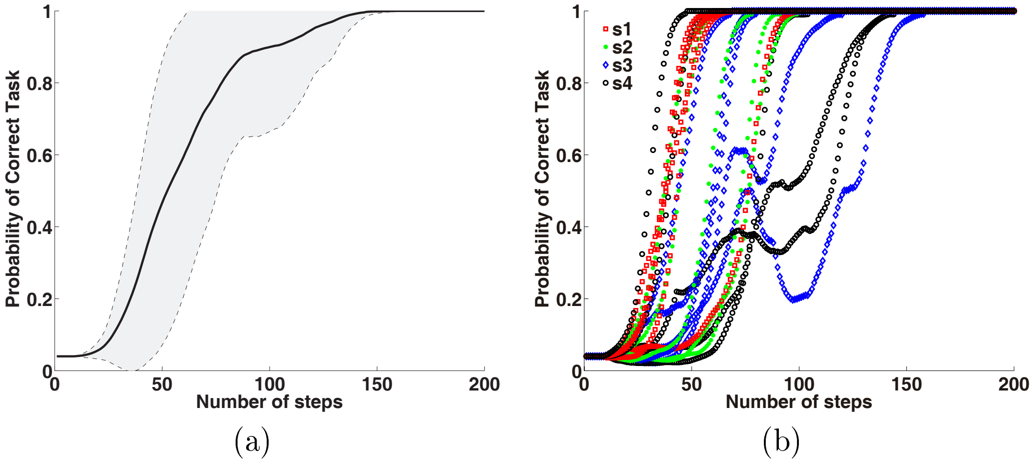
The first result obtained was that, for all the subjects, the device was able to reach the correct target while learning the signal model at the same time. Table 6.1 shows for each subject and run the number of steps needed to

Table 6.1: Number of steps needed to reach the correct target.

	Run 1	**Run 2**	**Run 3**	**Run 4**	**Run 5**	**mean ± std**
Subject 1	95	62	56	60	64	67 ± 16
Subject 2	89	77	98	60	62	77 ± 17
Subject 3	68	80	118	76	157	100 ± 37
Subject 4	98	142	57	142	47	97 ± 45

learn the correct task. Despite there was no sufficient information to perform statistical tests, no substantial differences were found among subjects or runs. On average, the number of steps needed to reach the task was of 85 ± 32. Notice that, in the previous chapter, the time needed to reach a target was 23 steps on average. However, the previous protocol required of a calibration phase of 400 actions on average, whereas in this approach the signal model was also learned during operation.

Figure 6.2 shows the evolution of the probability of the correct task, averaged for all subjects and runs (Fig. 6.2a), and separated for each subject and run (Fig. 6.2b). Despite on average the probability function is increasing continuous, there were some runs were the probability had a more erratic behavior, especially on subjects 3 and 4. This indicated that, initially, there were other tasks which were increasing their probability. Thus, the algorithm needed more time to converge to the correct task.



(a) (b)

Figure 6.2: Evolution of the probability of the correct task (a) averaged across subjects and runs (the shadowed area indicates the standard deviation); and (b) for each separate subject and run.

6.3.2 Offline experiments

Planning Methods: We compared the average number of steps (with maximum values of 400 steps) needed to identify the first task (section 6.2.2) when learning from scratch with different planning methods. Figure 6.3 shows the results averaged across subjects, runs and the two datasets. Notice that the large standard deviations are mainly due to the large variations in classification accuracy across subjects and datasets. Several planning methods were able to correctly estimate the tasks. On the other hand, a greedy exploration (i.e. always trying to follow the most probable task) does not allow the system to explore sufficiently, and at least some random exploration is necessary to allow a correct identification of the task (e.g. ε-greedy). The proposed planning method based on expected signal uncertainty (section 6.2.4), leads the system to regions that improve disambiguation among hypotheses in a faster way. Compared to assessing uncertainty only on the meaning space, the proposed algorithm performs better as such methods do not take into account the signal to meaning uncertainty inherent to the problem to solve.

Online re-estimation of classifier: After identifying the first task, the second phase of the algorithm was training and adapting a classifier (see section 6.2.3). The quality of the classifier learned with the proposed method can be measured according to the percentage of labels correctly learned (according to the ground truth label), see Figure 6.4. Notice how this percentage varied with the ten-fold accuracy of a given subject and dataset. In general, having accuracies higher than 75% guaranteed that more than 90% of the labels were correctly learned. This result shows that our algorithm could also be used to collect training data for calibrating any other state-of-the-art error potentials classifier, but has the important advantage of controlling the device at the same time.

Figure 6.5a demonstrate the advantage of training a classifier after the first reached task instead of keeping the estimation given by the Bhattacharyya coefficient. Indeed, the Bhattacharrya coefficient works very well for small amounts of data because it directly compares model parameters. On the other hand, when there are sufficient data, training a classifier allows for a faster adaptation since the classifier makes a much harder decision when evaluating a new EEG signal. Finally, figure 6.5b shows the number of tasks identified with respect to the accuracy of the dataset, and the number of tasks wrongly identified. Notice how the number of identified task is correlated to the quality of the dataset. Importantly, the algorithm was able to identify on average 20 tasks in 400 steps on average without the need for a calibration procedure.

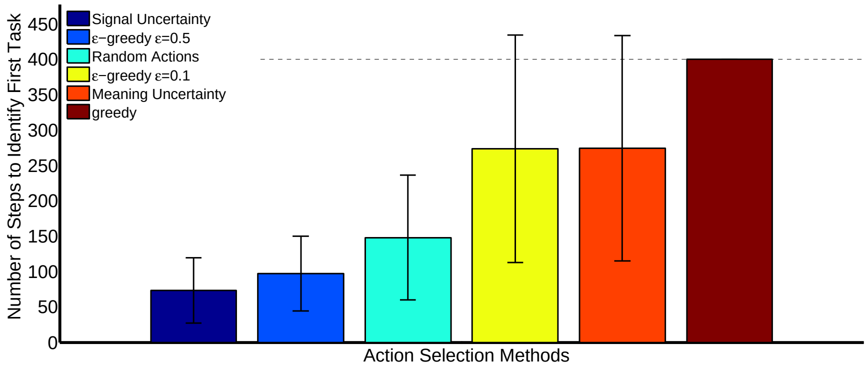


Figure 6.3: Number of steps needed to reach the first task following different exploration methods. Note that, if no task was learned after 400 steps the simulated run was considered a failure.

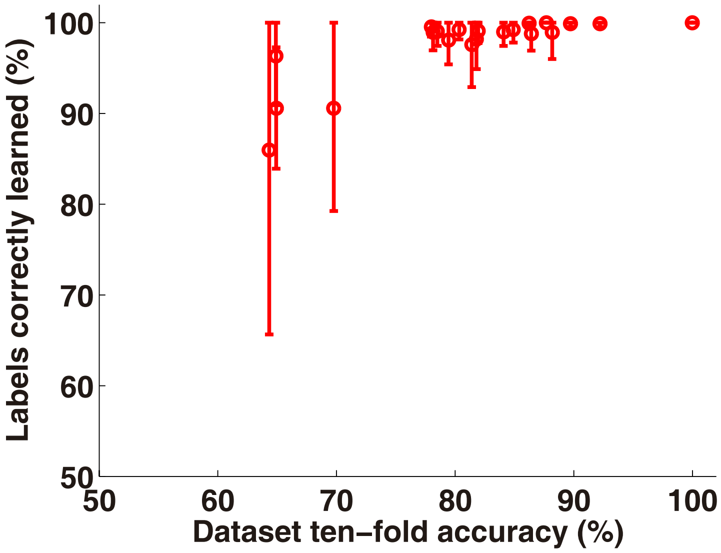


Figure 6.4: Percentage of estimated labels that matched with the ground truth labels, depending on the dataset ten-fold accuracy.

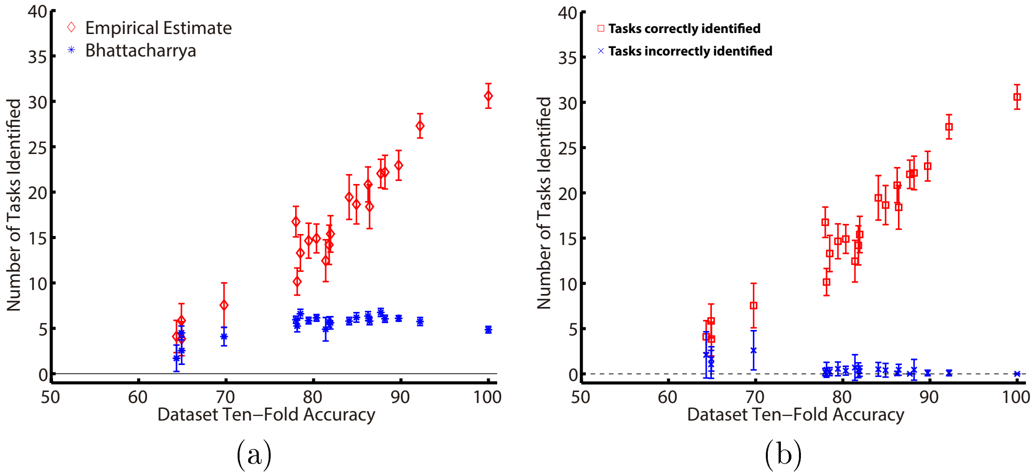


Figure 6.5: (a) Bhattacharrya versus empirical estimate of the predictive error during reuse as a function of the dataset ten-fold accuracy. (b) Number of tasks identified in 400 steps as a function of the dataset ten-fold accuracy (correctly and incorrectly identified in red and blue respectively).

6.4 Conclusions

In this work, we introduced a novel method for BCI based control of sequential tasks with feedback signals that do not require any calibration process. As a by-product, the method provides an unsupervised way to train a decoder with the same performance as state-of-the-art supervised classifiers, while keeping the system operational and solving, with a lower performance during the first steps, the unknown task. The algorithm has been tested with online experiments, showing that the users were able to guide from scratch an agent to a desired position. Furthermore, the system quickly learns the feedback signal model and in 400 steps (a reasonable number of trials needed for a typical supervised calibration [47, 79]), solved 20 tasks on average (where the maximum, in the presence of a perfect decoder, would be 30 times).

To improve the efficiency of the algorithm, we introduced a new planning method that uses the uncertainty in the decoder-target estimation. This planner is inspired by exploration methods with exploration bonuses that allow guiding to reduce the uncertainty in an efficient way. We showed that trying to follow the best hypothesis does not explore the space significantly to reduce uncertainty and thus identify the correct task. Only through an approach that plans how to reduce the uncertainty multiple steps ahead are we ensure that the agent will reach states that can only be explained by the correct hypothesis. A critical case is when there are many hypothesis that can be explained by almost *symmetric* hypotheses, by planning we are sure to go to the parts of the state space where they differ. To solve this issue, it is mandatory at least one action that disambiguate among hypothesis (e.g. the goal-reached action for the proposed protocol).

7 | Conclusions

In this thesis, we have presented an alternative brain-machine interface paradigm to neuroprosthetics that makes use of high-level cognitive signals associated to the task being performed by the device. The core idea is to delegate part of the intelligence to the device controller, emulating the role of subcortical levels during motor control. Specifically, this thesis has presented the first steps towards the development of this paradigm by making use of a non-invasive cognitive signal related to the user's error processing: the error-related potentials. Throughout the thesis, we have shown the feasibility of the proposed paradigm by illustrating how different devices can learn how to perform a task driven by the brain feedback given by the users, and given a set of these behaviors choose one among them. As the results showed, this BMI paradigm has several advantages: the paradigm can scale to different tasks and protocols, as long as the signal used (the error potentials) can be elicited; it allows to perform complex tasks thanks to the intelligent controller; it has a potential to adapt not only to what the user wants to do, but also how to do it (i.e. learn the user preference for a given task).

As most BMIs, the proposed paradigm requires a calibration phase to build a online detector of the ErrPs. For the proposed paradigm, this issue is of key importance as this calibration time adds to the task learning time greatly delaying the device operation. Thus, it would be convenient to have classifiers able to work irrespectively of the learning task being performed. This thesis has deeply studied this issue by designing five different experimental protocols (plus other two that were not presented here [78,79]), characterizing the cognitive signals used for the paradigm and their stability across different tasks. In fact, the thesis has shown how the latency of the error-related potentials is affected by the task being performed by the device, a problem well-known to be present in other ERPs such as the P300. To cope with this problem, the thesis has studied two different ways of re-using information among different tasks to generalize classifiers among different tasks. The first one uses state-of-the-art adaptive classifiers with proven utility in other BMI approaches, and the second one exploits the knowledge acquired during the neurophysiology analyses (i.e., that the latency of the ERPs is affected by the task) to correct the ErrP and P300 variations among tasks. Despite the adaptive classifiers were slower than calibrating each a new task from scratch, correcting the ERP latency among tasks proved to be of great utility as a way of generalizing classifiers among tasks.

As in the case of human motor control, the proposed paradigm must not only learn new policies, but it also has to store them so that they can be executed later on demand. The thesis demonstrates this second step using the same type of cognitive signals as feedback to select among a set of stored policies. Interestingly, the policy selection does not require that the stored policies have been learned using brain signals. In fact, in Chapter 5, the stored policies corresponded to optimal reaching trajectories to some predefined goals and the user simply selected among them. Pre-computing optimal policies is an interesting option, especially when the task to be learned is too complex for self-exploration or requires changes based on the scenario. Indeed, it is easy to incorporate context into the paradigm. For instance, the system could analyze the scene using computer vision techniques to create a set of active policies based on the environment and let the user choose among them (e.g. having detected a cup, a door and a switch, the set of policies will contain drinking from a cup, opening a door or turning a light on). The policy repertoire could also be a mix of learned policies and pre-computed ones, or even adapt previously learned ones to the current situation.

7.1 Future work

This thesis has shown the potential use of an alternative paradigm for current BMI systems. The results obtained here open the door to future developments, and pose several questions from both the BMIs and the neuroscience point of view. We will first discuss these questions in the context of cognitive neuroscience, and then dissertate about the paradigm in relation to BMIs.

One of the major difficulties of error-related potentials (shared with all ERPs), is the fact that these signals are a response to an event that elicits them. Indeed, there is no study in the literature of error potentials in which the event marking the onset of the potential is not clearly defined and measured. Consequently, this work has presented results using only discrete setups where the actions can be instantaneously evaluated as wrong or correct. However, real world applications (such as performing a complex trajectory with a robotic arm) imply the use of continuous actions where the error could appear at any moment of the trajectory being executed and not only at the beginning. Furthermore, being the error potential a cognitive process, its elicitation will depend on the subjective evaluation of each user. Thus, an important question is to evaluate whether there is any measurable assessment information in the user's brain during continuous movements performed by a device; and the feasibility of detecting these signals online. Our preliminary works with two subjects have shown that these potentials do exist during continuous motions of a virtual device, and we can detect them on single trial using frequency information [152]. Furthermore, these findings have also been studied in a realistic task with a mobile robot, where preliminary results suggested that it is feasible to teach devices under continuous spaces following the proposed BMI paradigm [88].

Despite this thesis presented a deep analysis of the error potentials under constrained conditions, it is still unclear what are the mechanisms underlying the generation of these signals. In fact, there is still a debate on the field of psychophysiology about whether the error-related negativity encodes a quantitative reward predictive error signal (RPE) or valence [41] (the difference between expected and received outcome), or rather the absolute value of this RPE [35]. According to some authors, the latter would imply that the ERN actually encodes the surprisingness of an event. In the specific case of ErrPs, several works have shown that this signal is also present under situations where the percentage of errors is very high ($40\% - 50\%$) but with a decrease in amplitude [47, 153]. In this thesis we showed that, during the beginning of any run of the teaching paradigm, there is a high percentage of errors (up to 70%) due to the device exploring the task space. Even so, the ErrP was present and we were able to detect it online. After a certain number of actions, this percentage of errors decreased drastically to 10% when the device learns the policy. Nonetheless, what would happen under situations where there is always a large number of errors for a large amount of time? It could happen that the user finds the errors the expected outcome, being the correct task execution an unexpected outcome. Some of these situations are present when performing very demanding tasks, e.g. hitting a ball with a baseball bat. Thus, designing and performing experiments would be necessary to address this issue.

Even after understanding the relationship between surprise, expectance and error potentials, there is still another open question linking error potentials and cognitive neuroscience: To what extent are the error-related potentials the same phenomenon that the error-related negativity? As stated in the introduction, the error-related potentials are characterized by three main peaks: N2, P3 and N4, whereas the ERN (and the FRN) is a single component appearing at around the same timing that the N2. There has been works comparing the N2, the FRN and ERN under similar experimental conditions [111], but what about the other ErrP components? In order to answer this question, a reasonable option would be to perform experiments with simultaneous recordings of EEG and fMRI [154] to determine the underlying generation of the ErrPs. In fact, a recent work showed the effectiveness of this experiment to understand the brain mechanisms generating the feedback-related negativity [35].

The presented paradigm strongly relies on the fact that a feedback signal can be extracted from the brain. In the instances presented throughout the thesis, this feedback encoded binary information about the actions wellness. However, no study was performed to understand up to what point the feedback given by the users encode the wellness about long term goals. Recently, Knox et al. suggested that the feedback given by humans could actually be encoded as the expected return in the long term, rather than a binary reward of the current action performed by a device [155–157]. In principle, this finding should be the case also for the error-related potentials. However, more studies would be needed to address this question.

From the point of view of brain-machine interfaces, the future work is associated to further study the proposed paradigm. This framework has shown how the users are able to teach a device by giving feedback. Thus, the device can adapt to the user's preferences. But, at the same time, the user is implicitly adapting to the device since this device provides a visual feedback of the task to the user. This term is usually known as co-adaptation between user and machine, and it is well known in the non-invasive and invasive BMI community [72, 100, 145], demonstrating that co-adaptation improves the final decoding performance that can be obtained with the BMI decoder. Despite a co-adaptation seemed to be present in the proposed paradigm, this characteristic has not be explicitly addressed. In which sense and how the user adapts to the machine (e.g. how the ErrPs vary in shape throughout time) is still an open question that needs to be studied.

In the introduction, a broad definition of the proposed paradigm was given (i.e. use cognitive signals and plug them into intelligent device controllers). On the contrary, throughout this thesis only a specific instance of this paradigm (error potentials + reinforcement learning) has been presented. Thus, the proposed paradigm has room for many improvements as there are many other instances of this paradigm that could be either work independently or together with the proposed instance. In non-invasive BMIs, there are several cognitive signals that could help in improving the paradigm, ranging from anticipatory signals encoding goal direction or movement intention [22]; to user's (c)overt attention to increase or decrease the device learning rate [113, 158]. For instance, thanks to the combination of goal-oriented cognitive signals (such as the target or hand direction), it would be possible to use the proposed system as a complementary paradigm fitting the approach into a two-step system: first decide where to go, then how to go. On the other hand, in principle this paradigm could also work using invasive BMIs, where richer information can be extracted from the brain. Several works have already reported that it is possible to measure single-neuron firing rates associated to error processing, and encoding it as rewards [101]. The use of invasive approaches would allow to obtain more informative rewards, that could be use to finely adapt the device to the user's preferences.

Finally, this paradigm was only tested in laboratory conditions. Thus, the natural next step of this thesis would be to apply the paradigm in realistic experiments so as to demonstrate its feasibility to work in out-of-the-lab scenarios. Indeed, the last two chapters solved two key problems of the paradigm: the calibration phase and the large amount of learning time. Thanks to these improvements, several experiments could be designed involving daily-life activities, such as drinking from a glass using a neuroprosthetic device. In the given example, this robotic device would learn from a set of optimal policies and adapt to the user's preferences without the need of any calibration phase. Once these experiments were designed, the proposed approach would be one step nearer to restore or replace the motor execution of limbs as a way of improving the daily life with real patients. Thus, performing experiments with patients would be the last and more demanding improvement of the current thesis.

Bibliography

[1] Wolpaw JR, Birbaumer N, McFarland DJ, Pfurtscheller G, and Vaughan TM. Brain-computer interfaces for communication and control. *Clin Neurophy*, 113(6):767–91, June 2002.

[2] Millán JdR, Rupp R, Müller-Putz GR, Murray-Smith R, Giugliemma C, Tangermann M, Vidaurre C, Cincotti F, Kübler A, Leeb R, Neuper C, Müller KR, and Mattia D. Combining brain–computer interfaces and assistive technologies: State-of-the-art and challenges. *Front Neurosci*, 4, 2010.

[3] Lebedev MA and Nicolelis MAL. Brain-machine interfaces: past, present and future. *Trends Neurosci*, 29(9):536–546, 2006.

[4] Carmena JM, Lebedev MA, Crist RE, O'Doherty JE, Santucci DM, Dimitrov DF, Patil PG, Henriquez CS, and Nicolelis MAL. Learning to control a brain-machine interface for reaching and grasping by primates. *PLoS Biol*, 1:193–208, 2003.

[5] Millán JdR, Renkens F, Mouri no J, and Gerstner W. Noninvasive brain-actuated control of a mobile robot by human EEG. *IEEE Trans Biomed Eng*, 51:1026–1033, 2004.

[6] Musallam S, Corneil BD, Greger B, Scherberger H, and Andersen RA. Cognitive control signals for neural prosthetics. *Science*, 305:162—163, 2004.

[7] Müller-Putz GR, Scherer R, Pfurtscheller G, and Rupp R. EEG-based neuroprosthesis control: A step towards clinical practice. *Neurosci Lett*, 382:169–174, 2005.

[8] Velliste M, Perel S, Spalding MC, Whitford AS, and Schwartz AB. Cortical control of a prosthetic arm for self-feeding. *Nature*, 453:1098–1101, 2008.

[9] Iturrate I, Antelis J, Kübler A, and Minguez J. Non-invasive brain-actuated wheelchair based on a P300 neurophysiological protocol and automated navigation. *IEEE Trans Robot*, 25:614–627, 2009.

[10] Tonin L, Carlson T, Leeb R, and JdR Millán. Brain-controlled telepresence robot by motor-disabled people. In *Conf Proc IEEE Eng Med Biol Soc*, 2011.

[11] Ethier C, Oby ER, Bauman MJ, and Miller LE. Restoration of grasp following paralysis through brain-controlled stimulation of muscles. *Nature*, 485:368–371, 2012.

[12] Hochberg LR, Bacher D, Jarosiewicz B, Masse NY, Simeral JD, Vogel J, Haddadin S, Liu J, Cash S, van der Smagt P, and Donoghue JP. Reach and grasp by people with tetraplegia using a neurally controlled robotic arm. *Nature*, 485:372–375, 2012.

[13] Carlson T and Millán JdR. Brain–controlled wheelchairs: A robotic architecture. *IEEE Robot Autom Mag*, 20:65–73, 2013.

[14] Collinger JL, Wodlinger B, Downey JE, Wang W, Tyler-Kabara EC, Weber DJ, McMorland AJC, Velliste M, Boninger ML, and Schwartz AB. High-performance neuroprosthetic control by an individual with tetraplegia. *The Lancet*, 381:557–564, 2013.

[15] Wolpaw JR and McFarland DJ. Control of a two-dimensional movement signal by a noninvasive brain-computer interface in humans. In *Proc Natl Acad Sci USA*, volume 101, pages 17849–17854, 2004.

[16] Santhanam G, Ryu SI, Yu BM, Afshar A, and Shenoy KV. A high-performance brain-computer interface. *Nature*, 442:195–198, 2006.

[17] Bradberry TJ, Gentili RJ, and Contreras-Vidal JL. Reconstructing three-dimensional hand movements from noninvasive electroencephalographic signals. *J Neurosci*, 30:3432–3437, 2010.

[18] O'Doherty JE, Lebedev MA, Ifft PJ, Zhuang KZ, Shokur S, Bleuler H, and Nicolelis MAL. Active tactile exploration using a brain-machine-brain interface. *Nature*, 479:228–231, 2011.

[19] Hauschild M, Mulliken GH, Fineman I, Loeb GE, and Andersen RA. Cognitive signals for brain-machine interfaces in posterior parietal cortex include continuous 3D trajectory commands. *Proc Natl Acad Sci USA*, 109:17075–17080, 2012.

[20] Ball T, Schulze-Bonhage A, Aertsen A, and Mehring C. Differential representation of arm movement direction in relation to cortical anatomy and function. *J Neural Eng*, 6:016006, 2009.

[21] Fried I, Mukamel R, and Kreiman G. Internally generated preactivation of single neurons in human medial frontal cortex predicts volition. *Neuron*, 69:548–562, 2011.

[22] Lew E, Chavarriaga R, Silvoni S, and Millán JdR. Detection of self-paced reaching movement intention from EEG signals. *Front Neuroeng*, 5:13:doi: 10.3389/fneng.2012.00013, 2012.

[23] Kim HK, Biggs J, Schloerb W, Carmena J, Lebedev MA, Nicolelis MAL, and Srinivasan MA. Continuous shared control for stabilizing reaching and grasping with brain-machine interfaces. *IEEE Trans Biomed Eng*, 53(6):1164–1173, 2006.

[24] Scott SH. Optimal feedback control and the neural basis of volitional motor control. *Nat Rev Neurosci*, 5:534–546, 2004.

[25] Dimitrijevic MR, Gerasimenko Y, and Pinter MM. Evidence for a spinal central pattern generator in humans. *Annals of the New York Academy of Sciences*, 860:360–76, November 1998.

[26] Courtine G, Gerasimenko Y, van den Brand R, Yew A, Musienko P, Zhong H, Song B, Ao Y, Ichiyama RM, Lavrov I, Roy RR, Sofroniew MV, and Edgerton VR. Transformation of nonfunctional spinal circuits into functional states after the loss of brain input. *Nat Neurosci*, 12:1333–1342, 2009.

[27] Takakusaki K, Saitoh K, Harada H, and Kashiwayanagi M. Role of basal ganglia-brainstem pathways in the control of motor behaviors. *Neuroscience research*, 50(2):137–51, October 2004.

[28] Kalaska JF, Scott SH, Cisek P, and Sergio LE. Cortical control of reaching movements. *Current opinion in neurobiology*, 7(6):849–59, December 1997.

[29] Rizzolatti G and Luppino G. The cortical motor system. *Neuron*, 31:889–901, 2001.

[30] Waldert S, Preissl H, Demandt E, Braun C, Birbaumer N, Aertsen A, and Mehring C. Hand movement direction decoded from MEG and EEG. *J Neurosci*, 28:1000–108, 2008.

[31] Holroyd CB and Coles MGH. The neural basis of human error processing: Reinforcement learning, dopamine, and the error-related negativity. *Psychol Rev*, 109:679–709, 2002.

[32] Nieuwenhuis S, Holroyd CB, Mola N, and Coles MGH. Reinforcement-related brain potentials from medial frontal cortex: origins and functional significance. *Neuroscience and Biobehavioral Reviews*, 28:441–448, 2004.

[33] Frank MJ, Woroch BS, and Curran T. Error-related negativity predicts reinforcement learning and conflict biases. *Neuron*, 47(4):495–501, August 2005.

[34] Holroyd CB, Krigolson OE, and Lee S. The impact of deliberative strategy dissociates ERP components related to conflict processing vs. reinforcement learning. *Frontiers in neuroscience*, 6(April):43, January 2012.

[35] Hauser TU, Iannaccone R, Stämpfli P, Drechsler R, Brandeis D, WalitzaS, and Brem S. The feedback-related negativity (FRN) revisited: New insights into the localization, meaning and network organization. *NeuroImage (to be published)*, August 2013.

[36] Allman JM, Hakeem A, Erwin JM, Nimchinsky E, and Hof P. The anterior cingulate cortex. *Annals of the New York Academy of Sciences*, 935(1):107–117, 2001.

[37] Schultz W. Predictive reward signal of dopamine neurons. *Journal of Neurophysiology*, 80(1):1–27, July 1998.

[38] Bayer HM and Glimcher PW. Midbrain dopamine neurons encode a quantitative reward prediction error signal. *Neuron*, 47(1):129–41, July 2005.

[39] Niv Y, Duff MO, and Dayan P. Dopamine, uncertainty and TD learning. *Behavioral and Brain Functions*, 1(6):1–9, 2005.

[40] Glimcher PW. Understanding dopamine and reinforcement learning: The dopamine reward prediction error hypothesis. *Proc Natl Acad Sci USA*, 108(Supplement 3):15647–15654, 2011.

[41] Luo Q and Qu C. Comparison enhances size sensitivity: neural correlates of outcome magnitude processing. *PloS one*, 8(8):e71186, January 2013.

[42] Falkenstein M, Hoormann J, Christ S, and Hohnsbein J. ERP components on reaction errors and their functional significance: A tutorial. *Biol Psychol*, 51:87–107, 2000.

[43] Miltner WHR, Braun CH, and Coles MGH. Event-related brain potentials following incorrect feedback in a time-estimation task: Evidence for a generic neural system for error detection. *Journal of Cognitive Neuroscience*, 9(6):788–798, 1997.

[44] van Schie HT, Mars RB, Coles MGH, and Bekkering H. Modulation of activity in medial frontal and motor cortices during error observation. *Neural Netw*, 7:549–554, 2004.

[45] Schalk G, Wolpaw JR, McFarland DJ, and Pfurtscheller G. EEG-based communication: Presence of an error potential. *Clin Neurophysiol*, 111(12):2138–2144, 2000.

[46] Ferrez PW and Millán JdR. Error-related EEG potentials generated during simulated brain-computer interaction. *IEEE Trans Biomed Eng*, 55:923–929, 2008.

[47] Chavarriaga R and Millán JdR. Learning from EEG error-related potentials in noninvasive brain-computer interfaces. *IEEE Trans Neural Syst Rehabil Eng*, 18(4):381–388, 2010.

[48] Duncan CC, Barry RJ, Connolly JF, Fischer C, Michie PT, Näätänen R, Polich J, Reinvang I, and van Petten C. Event-related potentials in clinical research: Guidelines for eliciting, recording, and quantifying mismatch negativity, P300, and N400. *Clin Neurophysiol*, 120:1883–1908, Sep 2009.

[49] Olofsson JK, Nordin S, Sequeira H, and Polich J. Affective picture processing: An integrative review of ERP findings. *Biol Psychol*, 77(3):247–65, March 2008.

[50] Luck SJ. *An introduction to the event-related potential technique*. The MIT Press, 2005.

[51] Kutas M, McCarthy G, and Donchin E. Augmenting mental chronometry: The P300 as a measure of stimulus evaluation time. *Science*, 197(4305):792, 1977.

[52] Buttfield A, Ferrez PW, and Millan JdR. Towards a robust BCI: Error potentials and online learning. *Neural Systems and Rehabilitation Engineering, IEEE Transactions on*, 14(2):164–168, 2006.

[53] Blumberg J, Rickert J, Waldert S, Schulze-Bonhage A, Aertsen A, and Mehring C. Adaptive classification for brain computer interfaces. In *Conf Proc IEEE Eng Med Biol Soc*, pages 2536–9, January 2007.

[54] Zeyl TJ and Chau T. A case study of linear classifiers adapted using imperfect labels derived from human event-related potentials. *Pattern Recognition Letters*, May 2013.

[55] Blankertz B, Dornhege G, Schafer C, Krepki R, Kohlmorgen J, Muller KR, Kunzmann V, Losch F, and Curio G. Boosting bit rates and error detection for the classification of fast-paced motor commands based on single-trial EEG analysis. *IEEE Trans Neural Syst Rehabil Eng*, 11(2):127–131, 2003.

[56] Ferrez PW and Millán JdR. Simultaneous real-time detection of motor imagery and error-related potentials for improved BCI accuracy. In *Proc 4th Int BCI Workshop and Train Course*, 2008.

[57] Dal Seno B, Matteucci M, and Mainardi L. Online detection of P300 and error potentials in a BCI speller. *Comput Intell Neurosci*, 2010:307254, 2010.

[58] Schmidt NM, Blankertz B, and Treder MS. Online detection of error-related potentials boosts the performance of mental typewriters. *BMC Neurosci*, 13(1):19, 2012.

[59] Llera A, Gómez V, and Kappen HJ. Adaptive classification on brain-computer interfaces using reinforcement signals. *Neural Computation*, 24(11):2900–2923, 2012.

[60] Spüler M, Bensch M, Kleih S, Rosenstiel W, Bogdan M, and Kübler A. Online use of error-related potentials in healthy users and people with severe motor impairment increases performance of a P300-BCI. *Clin Neurophysiol*, 123(7):1328–37, July 2012.

[61] Sutton RS and Barto AG. *Reinforcement learning: An introduction.* MIT Press, 1998.

[62] Sutton RS. Introduction: The challenge of reinforcement learning. In *Reinforcement learning: An introduction*, pages 1–3. Springer, 1992.

[63] Sutton RS, McAllester D, Singh S, and Mansour Y. Policy gradient methods for reinforcement learning with function approximation. In *Advances in Neural Information Processing Systems (NIPS)*, volume 12, pages 1057–1063, 1999.

[64] Lauer M and Riedmiller M. An algorithm for distributed reinforcement learning in cooperative multi-agent systems. In *In Proceedings of the Seventeenth International Conference on Machine Learning*. Citeseer, 2000.

[65] Geramifard A, Bowling M, Zinkevich M, and Sutton RS. iLSTD: Eligibility traces and convergence analysis. *Advances in Neural Information Processing Systems (NIPS)*, 19, 2007.

[66] Whiteson S, Taylor ME, and Stone P. Adaptive tile coding for value function approximation. *Technical report*, 2007.

[67] Peters J and Schaal S. Reinforcement learning of motor skills with policy gradients. *Neural Networks*, 21:682–697, 2008.

[68] Kober J and Peters J. Reinforcement learning in robotics: A survey. *Reinforcement Learning*, 2012.

[69] Perdikis S, Tavella M, Leeb R, Chavarriaga R, and Millán JdR. A supervised recalibration protocol for unbiased BCI. In *5th Int Brain-Computer Interface Conf*, 2011.

[70] Shenoy P, Krauledat M, Blankertz B, Rao RPN, and Müller KR. Towards adaptive classification for BCI. *J Neural Eng*, 3:13–23, 2006.

[71] Vidaurre C, Kawanabe M, von Bünau P, Blankertz B, and Müller KR. Toward unsupervised adaptation of LDA for brain-computer interfaces. *IEEE Trans Biomed Eng*, 58(3):587 –597, 2011.

[72] Vidaurre C, Sannelli C, Müller KR, and Blankertz B. Co-adaptive calibration to improve BCI efficiency. *J Neural Eng*, 8(2):025009, April 2011.

[73] Lotte F and Guan C. Learning from other subjects helps reducing brain-computer interface calibration time. *Int Conf on Audio Speech and Signal Processing (ICASSP)*, 1:614–617, 2010.

[74] Kindermans PJ and Verschore H. A P300 BCI for the masses: Prior information enables instant unsupervised spelling. In *NIPS*, pages 1–9, 2012.

[75] von Bünau P, Meinecke F, Király F, and Müller KR. Finding stationary subspaces in multivariate time series. *Phys Rev Lett*, 103(21):1–4, November 2009.

[76] Kawanabe M and Vidaurre C. Robust common spatial filters with a maxmin approach. In *Conf Proc IEEE Eng Med Biol Soc*, pages 2470–3, 2009.

[77] Kindermans PJ, Verstraeten D, and Schrauwen B. A bayesian model for exploiting application constraints to enable unsupervised training of a P300-based BCI. *PloS one*, 7(4):e33758, January 2012.

[78] Iturrate I, Montesano L, and Minguez J. Robot reinforcement learning using EEG-based reward signals. In *Proc of the IEEE Int Conf on Robotics and Automation (ICRA), Anchorage, USA*, pages 4822–4829, 2010.

[79] Iturrate I, Montesano L, and Minguez J. Single trial recognition of error-related potentials during observation of robot operation. In *Conf Proc IEEE Eng Med Biol Soc*, pages 4181–4184, 2010.

[80] Iturrate I, Montesano L, Chavarriaga R, Millán JdR, and Minguez J. Spatio-temporal filtering for EEG error related potentials. In *5th Int Brain-Computer Interface Conf*, pages 12–15, 2011.

[81] Iturrate I, Chavarriaga R, Montesano L, Minguez J, and Millán JdR. Cognitive signals for brain-machine interfaces: an alternative paradigm to neuroprosthetics control. *Submitted*, 2013.

[82] Iturrate I, Montesano L, Chavarriaga R, R Millán JdR, and Minguez J. Minimizing calibration time using inter-subject information of single-trial recognition of error potentials in brain-computer interfaces. In *Conf Proc IEEE Eng Med Biol Soc*, pages 6369–6372, 2011.

[83] Iturrate I, Montesano L, and Minguez J. Task-dependent signal variations in EEG error-related potentials for brain-computer interfaces. *Journal of Neural Engineering*, 10(2):026024, 2013.

[84] Farwell LA and Donchin E. Talking off the top of your head: Toward a mental prosthesis utilizing event-related brain potentials. *Electroencephalography and clinical neurophysiology*, 70(6):510–523, 1988.

[85] Iturrate I, Chavarriaga R, Montesano L, Minguez J, and Millán JdR. Latency correction of error potentials between different experiments reduces calibration time for single-trial classification. In *Conf Proc IEEE Eng Med Biol Soc*, pages 3288–3291, 2012.

[86] Iturrate I, Chavarriaga R, Montesano L, Minguez J, and Millán JdR. Latency correction of event-related potentials between different experimental protocols. *Journal of Neural Engineering (accepted with changes)*, 2013.

[87] Iturrate I, Montesano L, and Minguez J. Shared-control brain-computer interface for a two dimensional reaching task using eeg error-related potentials. In *Conf Proc IEEE Eng Med Biol Soc*, pages 5258–5262, 2013.

[88] Iturrate I, Omedes J, and Montesano L. Shared control of a robot using EEG-based feedback signals. In *Proceedings of the 2nd Workshop on Machine Learning for Interactive Systems (MLIS13) at IJCAI13*, pages 45–50, 2013.

[89] Iturrate I, Montesano L, and Minguez J. Shared-control of a virtual 2D cursor using error-related potentials. *Submitted*, 2013.

[90] Grizou J, Iturrate I, Montesano L, Lopes M, and Oudeyer PY. Zero-calibration BMIs for sequential tasks using error-related potentials. *Submitted*, 2013.

[91] Milekovic T, Ball T, Schulze-Bonhage A, Aertsen A, and Mehring C. Error-related electrocorticographic activity in humans during continuous movements. *J Neural Eng*, 9:026007, 2012.

[92] Walter WG, Cooper R, Aldridge VJ, McCallum WC, and Winter AL. Contingent negative variation: An electric sign of sensorimotor association and expectancy in the human brain. *Nature*, 203:380–384, 1964.

[93] Khaliliardali Z, Chavarriaga R, Gheorghe LA, and Millán JdR. Detection of anticipatory brain potentials during car driving. In *Proc 34th Annual Int Conf IEEE Eng Med Biol Soc*, 2012.

[94] Garipelli G, Chavarriaga R, and Millán JdR. Single trial analysis of slow cortical potentials: A study on anticipation related potentials. *J Neural Eng*, to appear, 2013.

[95] Hoffmann S and Falkenstein M. Predictive information processing in the brain: Principles, neural mechanisms and models. *Int J Psychophysiol*, 83:208Ð–212, 2012.

[96] Brázdil M, Roman R, Falkenstein M, Daniel P, Jurák P, and Rektor I. Error processing–evidence from intracerebral ERP recordings. *Exp Brain Res*, 146:460–466, 2002.

[97] Millán JdR, Posenato D, and Dedieu E. Continuous-action Q-learning. *Mach Learn*, 49:247–265, 2002.

[98] Martín P and JdR Millán. Robot arm reaching through neural inversions and reinforcement learning. *Robot Auton Syst*, 31:227–246, 2000.

[99] Schultz W, Dayan P, and Montague PR. A neural substrate of prediction and reward. *Science*, 275:1593–1599, 1997.

[100] DiGiovanna J, Mahmoudi B, Fortes J, Principe JC, and Sanchez JC. Coadaptive brain–machine interface via reinforcement learning. *IEEE Trans Biomed Eng*, 56:54–64, 2009.

[101] Mahmoudi B and Sanchez JC. A symbiotic brain-machine interface through value-based decision making. *PloS one*, 6:e14760, 2011.

[102] Millán JdR. On the need for on-line learning in brain-computer interfaces. In *Proc of the Int Joint Conf on Neural Networks (IJCNN), Budapest, Hungary*, volume 4, pages 2877–2882, 2004.

[103] Vidaurre C and Blankertz B. Towards a cure for BCI illiteracy. *Brain Topogr*, 23(2):194–198, 2010.

[104] Birbaumer N, Kubler A, Ghanayim N, Hinterberger T, Perelmouter J, Kaiser J, Iversen I, Kotchoubey B, Neumann N, and Flor H. The thought translation device (TTD) for completely paralyzed patients. *IEEE Trans Rehabil Eng*, 8(2):190–193, June 2000.

[105] Pfurtscheller G, Flotzinger D, and Kalcher J. Brain-computer interface: A new communication device for handicapped persons. *J Microcomput Appl*, 16:293–299, July 1993.

[106] Middendorf M, Mcmillan G, Calhoun G, and Jones KS. Brain-computer interfaces based on the steady-state visual-evoked response. *IEEE Trans Rehabil Eng*, 8(2):211–214, 2000.

[107] Elbert T, Rockstroh B, Lutzenberger W, and Birbaumer N. Biofeedback of slow cortical potentials. i. *Electroencephalogr Clin Neurophysiol*, 48(3):293–301, 1980.

[108] Escolano C, Antelis J, and Minguez J. A telepresence mobile robot controlled with a noninvasive brain-computer interface. *IEEE Trans Syst Man Cybern B Cybern*, 42(3):793–804, 2012.

[109] López-Larraz E, Iturrate I, Montesano L, and Minguez J. Real-time recognition of feedback error-related potentials during a time-estimation task. In *Conf Proc IEEE Eng Med Biol Soc*, 2010.

[110] Schalk G, McFarland DJ, Hinterberger T, Birbaumer N, and Wolpaw JR. BCI2000: A general-purpose brain-computer interface (BCI) system. *IEEE Trans Biomed Eng*, 51(6), 2004.

[111] Gruendler TOJ, Ullsperger M, and Huster RJ. Event-related potential correlates of performance-monitoring in a lateralized time-estimation task. *PloS one*, 6(10):e25591, January 2011.

[112] Delorme A, Palmer J, Onton J, Oostenveld R, and Makeig S. Independent EEG sources are dipolar. *PloS one*, 7(2):e30135, January 2012.

[113] Li L, Yao D, and Yin G. Spatio-temporal dynamics of visual selective attention identified by a common spatial pattern decomposition method. *Brain Res*, 1282:84–94, 2009.

[114] Hyvärinen A, Karhunen J, and Oja E. *Independent Component Analysis*. Wiley Interscience, 2001.

[115] Makeig S, Westerfield M, Jung TP, Enghoff S, Townsend J, Courchesne E, and Sejnowski TJ. Dynamic brain sources of visual evoked responses. *Science*, 295(5555):690–694, 2002.

[116] Debener S, Ullsperger M, Siegel M, Fiehler K, Von Cramon DY, and Engel AK. Trial-by-trial coupling of concurrent electroencephalogram and functional magnetic resonance imaging identifies the dynamics of performance monitoring. *J Neurosci*, 25(50):11730, 2005.

[117] Onton J and Makeig S. Information-based modeling of event-related brain dynamics. *Prog Brain Res*, 159:99–120, 2006.

[118] Fatourechi M, Bashashati A, Ward RK, and Birch GE. EMG and EOG artifacts in brain computer interface systems: A survey. *Clin Neurophysiol*, 118(3):480–494, 2007.

[119] Xu N, Gao X, Hong B, Miao X, Gao S, and Yang F. BCI competition 2003-data set IIb: Enhancing P300 wave detection using ICA-based subspace projections for BCI applications. *IEEE Trans Biomed Eng*, 51(6):1067–1072, 2004.

[120] Himberg J, Hyvärinen A, and Esposito F. Validating the independent components of neuroimaging time series via clustering and visualization. *Neuroimage*, 22(3):1214–1222, 2004.

[121] Diana G and Tommasi C. Cross-validation methods in principal component analysis: A comparison. *Stat Methods Appt*, 11(1):71–82, 2002.

[122] Pascual-Marqui RD. Standardized low resolution brain electromagnetic tomography (sLORETA): Technical details. *Methods Find Exp Clin Pharmacol*, pages 5–12, 2002.

[123] Handy TC. *Event-related potentials: A methods handbook.* The MIT Press, 2005.

[124] Blankertz B, Lemm S, Treder M, Haufe S, and Müller KR. Single-trial analysis and classification of ERP components: A tutorial. *Neuroimage*, 2010.

[125] Duda RO, Hart PE, and Stork DG. *Pattern Classification*, chapter Unsupervised learning and clustering. Wiley, 2000.

[126] Lu S, Guan C, and Zhang H. Unsupervised brain computer interface based on intersubject information and online adaptation. *IEEE Trans Neural Syst Rehabil Eng*, 17(2):135–145, 2009.

[127] Sellers E, Krusienski D, McFarland D, Vaughan T, and Wolpaw J. A P300 event-related potential brain-computer interface (BCI): The effects of matrix size and inter stimulus interval on performance. *Biol Psychol*, 73(3):242–52, October 2006.

[128] Polich J. On the relationship between EEG and P300: Individual differences, aging, and ultradian rhythms. *Int J Psychophysiol*, 26(1-3):299–317, 1997.

[129] Davies PL, Segalowitz SJ, and Gavin WJ. Development of error-monitoring event-related potentials in adolescents. *Ann N Y Acad Sci*, 1021:324–328, Jun 2004.

[130] Chavarriaga R, Perrin X, Siegwart R, and Millán JdR. Anticipation- and error-related EEG signals during realistic human-machine interaction: A study on visual and tactile feedback. In *Conf Proc IEEE Eng Med Biol Soc*, 2012.

[131] Thompson DE, Warschausky S, and Huggins JE. Classifier-based latency estimation: A novel way to estimate and predict BCI accuracy. *Journal of neural engineering*, 10(1):016006, December 2012.

[132] Wilson JA, Mellinger J, Schalk G, and Williams J. A procedure for measuring latencies in brain-computer interfaces. *IEEE Transactions on Biomedical Engineering*, 57(7):1785–97, July 2010.

[133] Sauser E. Robottoolkit. Available online: lasa.epfl.ch/RobotToolKit.

[134] Woody CD and Nahvi MJ. Application of optimum linear filter theory to the detection of cortical signals preceding facial movement in cat. *Exp Brain Res*, 16:455–465, 1973.

[135] Levine SP, Huggins JE, BeMent SL, Kushwaha RK, Schuh LA, Rohde MM, Passaro EA, Ross DA, Elisevich KV, and Smith BJ. A direct brain interface based on event-related potentials. *IEEE Trans Rehabil Eng*, 8(2):180, 2000.

[136] Benjamini Y and Yekutieli D. The control of the false discovery rate in multiple testing under dependency. *Ann Stat*, pages 1165–1188, 2001.

[137] Jung TP, Makeig S, Westerfield M, Townsend J, Courchesne E, and Sejnowski TJ. Analysis and visualization of single-trial event-related potentials. *Hum Brain Mapp*, 14(3):166–185, 2001.

[138] Gerson AD, Parra LC, and Sajda P. Cortically-coupled computer vision for rapid image search. *IEEE Trans Neural Syst Rehabil Eng*, 14(2):174–179, Jun 2006.

[139] Hong B, Guo F, Liu T, Gao X, and Gao S. N200-speller using motion-onset visual response. *Clin Neurophysiol*, 120(9):1658–1666, 2009.

[140] Berndt D and Clifford J. Using dynamic time warping to find patterns in time series. In *AAAI Workshop on Knowledge Discovery in Databases*, volume 398, pages 229–248, 1994.

[141] Casarotto S, Bianchi AM, Cerutti S, and Chiarenza GA. Dynamic time warping in the analysis of event-related potentials. *IEEE Eng Med Biol Mag*, 24(1):68–77, 2005.

[142] Croft RJ and Barry RJ. EOG correction of blinks with saccade coefficients: a test and revision of the aligned-artefact average solution. *Clinical neurophysiology*, 111(3):444–51, March 2000.

[143] Schlögl A, Keinrath C, Zimmermann D, Scherer R, Leeb R, and Pfurtscheller G. A fully automated correction method of EOG artifacts in EEG recordings. *Clinical neurophysiology*, 118(1):98–104, January 2007.

[144] Bishop CM. *Pattern recognition and machine learning*, chapter Linear models for classification. Springer, 2006.

[145] Orsborn AL, Dangi S, Moorman HG, and Carmena J. Closed-loop decoder adaptation on intermediate time-scales facilitates rapid BMI performance improvements independent of decoder initialization conditions. *IEEE Transactions on neural systems and rehabilitation engineering*, 20(4), 2012.

[146] Friedman JH. Regularized discriminant analysis. *Journal of the American statistical association*, 84(405):165–175, 1989.

[147] Bengio Y and Grandvalet Y. No unbiased estimator of the variance of k-fold cross-validation. *The Journal of Machine Learning Research*, 5:1089–1105, 2004.

[148] Kailath T. The divergence and bhattacharyya distance measures in signal selection. *IEEE Trans. Commun. Technol.*, 15(3):52–60, 1967.

[149] Chulhee L and Euisun C. Bayes error evaluation of the gaussian ml classifier. *Geoscience and Remote Sensing, IEEE Transactions on*, 38(3):1471–1475, 2000.

[150] Brafman RI and Tennenholtz M. R-max-a general polynomial time algorithm for near-optimal reinforcement learning. *Journal of Machine Learning Research*, 3, 2003.

[151] Kolter JZ and Ng AY. Near-bayesian exploration in polynomial time. In *International Conference on Machine Learning*. ACM, 2009.

[152] Iturrate I, Omedes J, and Montesano L. Detection of event-less error related potentials. In *IROS 2013 Workshop on Neuroscience and Robotics*, 2013.

[153] Ferrez PW. *Error-Related EEG Potentials in Brain-Computer Interfaces.* PhD thesis, École Polytechnique Fédérale de Laussane, 2007.

[154] Rosa MJ, Daunizeau J, and Friston KJ. EEG-fMRI integration: a critical review of biophysical modeling and data analysis approaches. *Journal of Integrative Neuroscience*, 09(04):453–476, December 2010.

[155] Knox WB and Stone P. Interactively shaping agents via human reinforcement: The tamer framework. In *Proceedings of the fifth international conference on Knowledge capture*, pages 9–16. ACM, 2009.

[156] Knox WB and Stone P. Combining manual feedback with subsequent mdp reward signals for reinforcement learning. In *Proceedings of the 9th International Conference on Autonomous Agents and Multiagent Systems (AAMAS'10)*, pages 5–12, 2010.

[157] Knox WB, Glass B, Love B, Maddox W, and Stone P. How humans teach agents: A new experimental perspective. *International Journal of Social Robotics, Special Issue on Robot Learning from Demonstration*, 2012.

[158] Tonin L, Leeb R, and Millán JdR. Time-dependent approach for single trial classification of covert visuospatial attentions. *Journal of Neural Engineering*, 9(4):e045011, 2012.