Toward a decision support system for the clinical pathways assessment
-draft-

Simona Bernardi, Cristian Mahulea and Jorge Albareda [*]

May 13, 2019

Abstract

This paper presents a decision support system to be used in hospital management tasks which is based on the clinical pathways. We propose a very simple graphical modeling language based on a small number of primitive elements through which the medical doctors could introduce a clinical pathway for a specific disease. Three essential aspects related to a clinical pathway can be specified in this language: (1) patient flow; (2) resource utilization; and (3) information interchange. This high-level language is a domain specific modeling language called *Healthcare System Specification (HSS)*, and it is defined as an Unified Modeling Language (UML) profile. A model to model transformation is also proposed in order to obtain, from the pathways HSS specification, a Stochastic Well-formed Net (SWN) model that enables a formal analysis of the modeled system and, if needed, to apply synthesis methods enforcing specified requirements. The transformation is based on the application of local rules. The clinical pathway of hip fracture from the "Lozano Blesa" University hospital in Zaragoza is taken as an example.

Published as:
S. Bernardi, C. Mahulea and J. Albareda, "Toward a decision support system for the clinical pathways assessment," Discrete Event Dynamic Systems: Theory and Applications, 29(1):91–125, March 2019. DOI: 10.1007/s10626-019-00279-9.

---

[*] Corresponding author is: Cristian Mahulea, María de Luna 1, 50018 Zaragoza, Spain, tel. + (34) 976 76 2517, Fax: + (34) 976761914, e-mail: {cmahulea@unizar.es}.

This work has been partially supported by CICYT - FEDER (Spain-EU) under Grant DPI2014-57252-R and by the Aragonese Government (Spain) under grant T94 DisCo group.

The authors want to thank José Manuel Colom, co-author of the conference paper on which is based this manuscript, for all his help during the initial version.

S. Bernardi and C. Mahulea are with Dept. of Computer Science and Systems Engineering, University of Zaragoza (Spain). J. Albareda is with Orthopedic Surgery Department, University Hospital "Lozano Blesa", Zaragoza (Spain).

1 Introduction

Public healthcare system is managed by national or local governments, who define the purpose and targets of the service together with policies and financial resources. However, because of the complexity of understanding, planning and controlling the system behavior, governments struggle to make good short-term and long-term decisions. This struggle is the same all through the management hierarchy, down to the daily work with patients. This means that the implementation of new legislation is often expensive, can have a long delay and is prone to fail.

This paper proposes a method for the management of the healthcare systems, in particular hospital management, based on *clinical pathways* [24] developed and used by the medical staff in the hospitals. We consider as case study the Clinical Hospital "Lozano Blesa" of Zaragoza, in particular the Orthopedic Department of this hospital. As all the other departments of the hospital, for each disease, treatment or surgery a clinical pathway is developed giving recommendations to the medical staff (medical doctors in general) on the different diagnostics and therapeutic interventions that should follow. These medical pathways are protocolized and approved by the hospital, protecting the medical staff against legal issues. Any deviation from the clinical pathway should be well justified and, in general, approved by the head of the department. Clinical pathways can be used also to promote effective and efficient healthcare by guiding the introduction of new procedures or services, decreasing the waiting lists for surgeries and so on.

When the clinical pathways are deployed, they should have the needed resources to carry out the specified tasks and activities. The correct assignment of resources could have benefits on the implementations of these pathways. Moreover, the interconnections between different pathways can introduce non desired behaviors as for example synchronizations of two pathways that cannot be done as consequence of taking local decisions independently. It is very important to notice that the verification of the correctness of the resources usage (sequences of allocation and release operations) together with the fulfillment of the performance, is impossible to be checked without the use of models allowing to study the pathways before its deployment. High level graphical models are, in general, preferred since the specification of the clinical pathways is carried out by the medical doctors, who have not expertise in modeling using formal languages. On the other hand, the formal models are needed for analysis purposes and, additionally, the analysis cannot be performed without software tools. Observe that numerous techniques and tools have been applied to evaluate the performance and efficiency of healthcare systems, such as simple ratio analysis, least-squares, the frontier regression analysis, etc. [13, 4, 14, 21].

This paper proposes a new modeling methodology for describing healthcare systems, in particular for the management and planning of hospitals. The main idea of the approach is to model different *clinical pathways* (or *care plans*) existing in a hospital for different medical problems and cares of different diseases. These pathways can be defined by the interaction among activities (treatments and cares), resources (medical staff, medical equipments, operating rooms, etc.), and requirements from the stakeholders. Available modeling methods cannot fully handle the complex and often-changing activity interactions found in healthcare systems. In this paper, we propose a domain specific modeling language -called *Healthcare System Specification (HSS)*- that is defined as an UML [42] profile. We start from clinical pathways defined using some basic elements of the UML activity diagram that are easy to understand by the medical doctors. We then extend the UML activity diagram semantics -by applying the HSS profile- in order to specify the resource, workload, timing requirements and information exchange in the health domain. Such UML extensions consist of tagged-values associated to stereotyped model elements of UML, i.e., activity nodes and edges. Transformation patterns from UML basic elements to Stochastic Well-Formed Nets (SWN) are then provided in order to get a formal model of the clinical pathway. The resulting SWN model can be used to provide estimation of different performance indices using available SWN solution techniques, such as state-based or event-driven simulation techniques. However, in many cases, the state-based techniques are impossible to apply since they suffer from the state explosion problem and simulation ones are time-consuming. In order to cope with space and time complexity issues, we propose three types of approximations / reductions that allow obtaining models which can be analyzed with net-level analysis techniques. These new abstractions we call *facets* of the pathways and we provide in this paper three possible facets (see Section 5.2). Additionally, we show that these facets provide models with a well defined structure making possible the use of *structural analysis* to prove properties of the system.

2 Methodology overview

Fig. 1 sketches the modeling and analysis workflow that is carried out by the responsible of the hospital management, according to our proposal. The responsible has to take decisions about the planning of the treatment and care activities considering the available resources.

The decision maker creates specifications of the clinical pathways by using a Domain Specific Modeling Language (DSML), namely *Healthcare System Specifications (HSS)* (Fig. 1-a). Our main objective is to propose an intuitive language that does not require too much effort for the decision makers to learn it while being compliant to standard modeling languages in order to exploit the current available tools.

Among the plethora of standard modeling languages, we use the UML (see the appendix for a short introduction) that is a well known general purpose Object Management Group (OMG) standard for software system specification. UML encompasses several types of diagrams that enable to specify a system from different point of views. Our approach considers two types of UML diagrams: the activity diagrams and the class diagrams. The former can be used to specify the clinical pathways as workflows (one activity diagram for each clinical pathway), the latter can be used to define the overall hospital resources and patients' health information (or medical records).

In order to support the assessment of the clinical pathways through qualitative and quantitative analysis it is necessary to add domain-specific semantics to both diagrams. UML can be customized for a particular application domain, through the *profiling*. Thus, we exploit such extension capability to define the HSS as a profile. The main mechanisms used to define the HSS are the *stereotypes* that allow one to create new model elements, derived from existing ones, having specific domain properties called

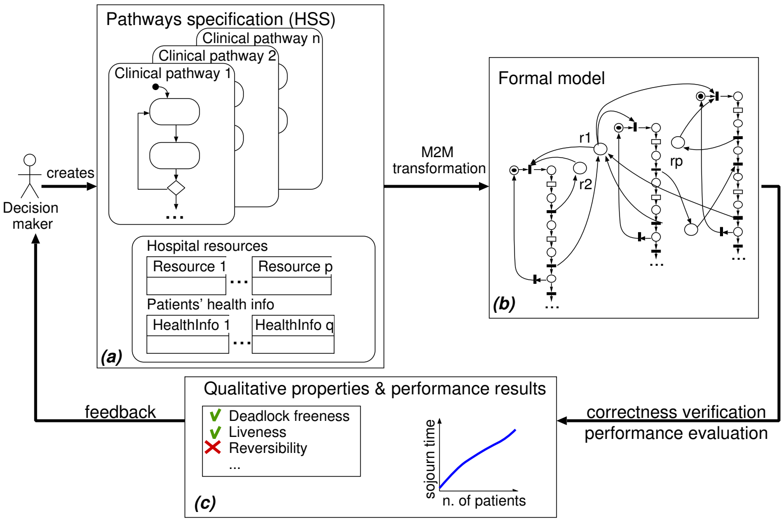


Figure 1: Modeling and analysis workflow.

tags. When a stereotype is applied to a model element, the model element inherits the tags (or the properties) of the stereotype and the value assigned to a tag is called *tagged-value*.

UML is a semi-formal modeling language (i.e., it has not a rigorous mathematical syntax) that can be used to specify the system but cannot be directly used for analysis or synthesis purposes. For this reason we propose a set of rules to transform an UML model -enriched with HSS specification- into a formal model (Fig. 1-b).

The formal model is a colored Petri Net model, concretely a Stochastic Well-formed Net (SWN) model (see the appendix for a short introduction), that can be used to estimate performance measures -e.g., patient sojourn time, utilization of hospital resources, etc.- via either state-based techniques or event-driven simulation, using current available tools (Fig. 1-c). Many software tools exists for the analysis of colored Petri net models, e.g., GreatSPN [29], TimeNET [54] or CPN tools [52]. Moreover, the SWN model can be relaxed to so called *facets* of the model by decolouring, that is by removing state variables that are unnecessary in the study of a particular property, and the properties can be verified considering just the structure of the model then avoiding the exploration of the reachability graph. For example, by neglecting the timing specifications, the net system without time (obtained by simple assuming zero time durations) can be used to check qualitative properties of the system, e.g., deadlock-freeness.

Finally, the results of the analysis and synthesis are shown to the medical doctors in a form understandable for them, e.g., by means of a dashboard or directly mapped into the initial UML-HSS pathways specification. This last step of the method is a future work and it is not considered in this paper.

3 Related works

In the recent scientific literature, healthcare systems constitute an important research topic. The literature on these systems is huge and is continuously expanding. As reported in [12], about 30 new articles per day are written on simulation and modeling in healthcare systems. The Research Into Global Healthcare Tool (RIGHT) project[1] perceived that *there is an overwhelming demand for modeling and simulation tools in the healthcare community, whilst there is a sizeable body of knowledge among the modeling community, yet there is a clear gap between what is needed and what is on offer. Bridging such a gap between the two communities will require a great deal of effort in trying to identify main linkages and successfully setting them up*.

General modeling of healthcare system. *Simulation techniques* have mainly been used in the literature for the analysis of healthcare systems. Amongst these techniques, the most frequent modeling methodologies are discrete event simulation [20, 33, 25, 36, 19, 27, 30, 31, 45, 49, 51] and system dynamics [26, 2]. Simulation techniques do not suffer from the state explosion problem and are a good alternative to state-based techniques, nevertheless the time needed to get a required accuracy is often prohibitive. The net-level analysis techniques, such as the techniques based on the computation of performance bounds [7, 8, 15, 16], offer a trade-off between the solution time and the accuracy of the results. Additionally, such techniques also provide complementary information (e.g., the slowest part of the model) beside the performance values. One of the main motivation of the proposed modeling methodology is to allow the application of net-level analysis techniques, beside the simulation ones.

One of the most frequently used approach for modeling complex information systems is *Business Process Modeling* (BPM). Several standard modeling languages exist for this purpose, such as the UML [42], Business Process Model and Notation (BPMN) [40], Integration DEFinition (IDEF) [10], Extended Enterprise Modeling Language (EEML) [28] and Computer Integrated Manufacturing Open System Architecture (CIMOSA) [34]. A business process model reflects the behavior of a complex process in terms of the involved participants, the causality/concurrence/conflict of the activities carried out by the formers as well as the data flow generated by the process. Such model is then used for performing simulation experiments and implementing analysis methods to better understand the effects of running that process. These kind of models have been applied to healthcare systems, see for example [3, 6, 22, 46]. For example, in [22], Activity Diagrams and Class Diagrams are used to define an UML model. The proposed methodology is general and can be used to obtain models corresponding to different views of the healthcare system. The modeling methodology proposed in this paper can be applied only to clinical pathways, hence describing the flow of treatments and medical tests that a patient should follow to get healthier. On the one hand, just considering clinical pathways is a limitation of our approach since parallel activities cannot be modeled. On the other hand, the resulting formal models have a well-defined structure (that is, they belong to known Petri Net classes defined in the literature), thus allowing to apply efficient structural techniques for their analysis.

The works [6] and [23] provide, as in this paper, an UML model of healthcare sys-

---

[1] `www.right.org.uk`

tems and translation to Petri nets. In particular, in [6] Statecharts and Class Diagrams are used to model healthcare systems and an (ad-hoc) conversion algorithm is proposed to get colored Petri Nets: each UML model is converted into a Petri net and then the final Petri Net model is obtained through the merging of common places and transitions (the identification of the common places/transitions of the Petri Net relies on the common names assigned to the states belonging to the different views in the UML models). Authors of [23] provide a methodology for the efficient management of hospital departments based on three elements: the modeling module (which employs UML and timed Petri Nets to detail patients flow and dynamics), the optimization module (which employs the fluid relaxation to approximate in a continuous Petri net framework the model and to optimize suitable performance indices), and the simulation and decision module (which verifies that the optimized parameters allow an effective workflow organization while maximizing the patient flow). There are many differences with our work:

- we provide a domain specific modeling language, hence we provide a *semantics* for the UML language via UML profiling techniques to enhance the UML models with timing/stochastic information and to support the consistency of the behavioral and structural views of system;
- we focus only on clinical pathways allowing us to obtain relaxations (called facets) with well defined structures making possible the use of structural analysis;
- we define local transformation rules for the transformation of the UML to a SWN and not ad-hoc rules;
- our SWN model is more general and the resource view of the system is only one facet. It is possible to obtain other "views", as for example in Subsection 5.2.

Clinical pathways modeling. In [53] general medical protocols are modeled by colored Petri nets and the analysis techniques are based on the exploration of the reachability graph. Even if the authors develop a method for limiting the exploration of an augmented reachability graph, if many clinical pathways are modeled the state explosion problem may arise. Furthermore, the models are more difficult to understand by a medical doctor than the one based on UML. The same difficulty in understanding the model by the medical doctors may appear in [37] where a Petri net modeling methodology is proposed for medical protocols in primary healthcare.

Other modeling languages have been proposed for clinical pathways, e.g., *Guideline Interchange Format* (GLIF) [11] or *pathway elements model* [5], for decision support systems. These models have friendly interfaces for the doctors and mainly can be used to simulate the system under different scenarios. This paper presents a domain modeling language, transformations to SWN and different possible relaxations (facets) useful when a particular property of the healthcare system is studied.

4 HSS: a domain specific modeling language for clinical pathways

In this Section, we define a DSML for the modeling and analysis of clinical pathways as an UML profile [47, 48], namely *Healthcare System Specification (HSS)*. For the HSS definition, we apply a well-established systematic approach in model-driven system engineering [47]. In particular, the approach consists of two main steps: first a domain model [38] is defined for the domain of interest; then, the domain concepts are mapped to UML extensions, i.e., stereotypes and tags which constitute the profile, by applying patterns [35].

4.1 The domain model

A domain model is a visual glossary, represented by a Class Diagram, that includes a set of conceptual classes (and their relationships) in a domain of interest [38]. The Class Diagram in Fig. 2 shows the domain model that has been built with the purpose of capturing those concepts in the healthcare domain that are relevant for the analysis of clinical pathways, according to the needs of the hospital managers. The domain model specifies the requirements for the domain specific language that will be defined as an UML profile in the second step. Moreover, it describes the type of information that is required to be modeled by the hospital managers in order to enable the analysis. In particular, a *healthcare analysis context* (Fig. 2) includes two modeling views: a structural view and a behavioral one.

The **structural view** (left part of the figure) is concerned with the specification of the hospital's *assets*, that are the *resource types* (e.g., beds, operation rooms, personnel) and the patients' *health information* (e.g., patients included in auto-transfusion programs). The grey classes in the figure, related to the *Resource type* and *Health info* concepts respectively, represent fine-grained data about the *resources* and the *patients' medical records* (e.g., patients' allergies and suffered diseases). In particular, the *Resource* class is an abstract concept that can be further refined according to the type of resource.

The **behavioral view** (right part) is concerned with the specification of one or more *clinical scenarios*. A clinical scenario includes a *clinical pathway* to be followed for the treatment of a disease and the *treatment requests*. The *treatment request* concept represents the workload in the healthcare domain, in terms of number of patients (*nPatients*) that need to be treated according to a given clinical pathway. Fine-grained data related to the patients that requested a treatment (e.g., patient's identification number) are represented by the *Patient* class. A *clinical pathway* includes a set of care steps that need to be carried out in a given temporal order. A *care step* represents a task (e.g., provide rules for antibiotics profilaxis) to be performed by the personnel in charge of the patient that may require the usage of resources (e.g., the doctor) or patient's health information (e.g., the patients' allergies) to be carried out. The *Resource usage step* and *Health info usage step* are particular types of care steps. The former enables to specify the type and number of hospital resources to be assigned to a (set of) step(s) through the *acquire* and *release* concepts (association-end names of the associations between

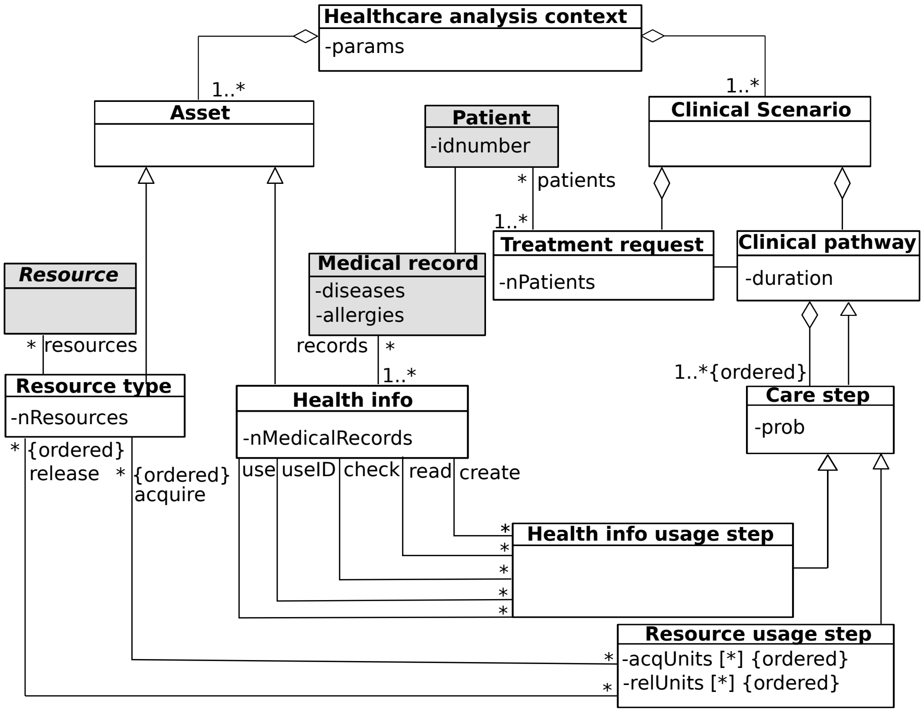


Figure 2: Domain model for the modeling and analysis of clinical pathways.

the *Resource usage step* and *Resource type* classes). In particular, in a single *Resource usage step* it is possible to allocate (*acquire*) -or deallocate (*release*)- as many resources for each type as indicated by the *acqUnits* -or *relUnits*- attribute, which represent a list of numbers ordered according to the allocated/deallocated resource types. The *Health info usage step* allows one to specify the type of operation to be performed on a given patients' health information: *create* (e.g., add the patient in the auto-transfusion program list), *read* (e.g., read the patient's data in the list), *check* (e.g., check if the patient is included in the list), *useID* and *use* (e.g., use the information of the patient's status either considering or not considering, respectively, her/his identity).

It is worth to observe that, from the analysis perspective the proposed approach is based on the availability of summary information on the number of patient treatment requests (*nPatients*), number of resources by type (*nResources*) and of the patients' health information (*nMedicalRecords*). Therefore, the user -e.g., the hospital manager- does not need to specify explicitly fine-grained data in the structural view (grey classes in the figure).

On the other hand, in order to support the performance analysis of clinical pathways, the domain model includes also timing and stochastic concepts. The *duration* property represents the time spent to carry out a clinical pathway (i.e., typically a performance measure to be estimated with the analysis) as well as the time needed to carry out a care step (i.e., a parameter usually provided as input to the analysis). Observe that a care step is a specialization of the clinical pathway, so it inherits the properties of the latter. The *prob* property refers to the probability associated to a care step in case that alternative steps can be carried out (e.g., 10% of the patients suffer urinary infection and urgent pre-operative study is needed in such situation). Finally, the *params* property of the *healthcare analysis context* enables to specify input parameters to perform sensitivity or *what-if* analysis, e.g., the number of patients to be treated (*nPatients*) by following a given clinical pathway can be parameterized.

4.2 The HSS profile

The HSS language is defined as an UML profile that includes a set of stereotypes and imports the model library of the MARTE profile [41] -which is an OMG standard- in order to reuse the predefined data-types, namely the basic Non Functional Property (NFP) data-types.

The domain model is mapped to UML extensions by applying a set of patterns [35], which are aimed at defining a small set of new extensions to facilitate the fast learning of the language to the final users. First, a stereotype is created for those white classes of the domain model that have attributes or association-ends explicitly named (i.e., in Fig. 2, all the white classes but *Asset* and *Clinical Scenario*). In order to define the proper extension for each stereotype, we have considered the UML meta-classes whose instances are model elements of UML Class or Activity diagrams. For example, the *ClinicalGuideline* stereotype extends the *Activity* meta-class, thus it can be applied to an UML activity diagram modeling a clinical pathway. On the other hand, the *ResourceType* and *HealthInfo* stereotypes extend the *Class* meta-class, thus the stereotypes can be applied to classes of a class diagram. The generalization relation of the domain model is maintained in the mapping. Then, the *CareStep* is a sub-stereotype of *ClinicalGuideline* and inherits the tags of the stereotype as well as the extended meta-class: a care step can be applied to activity diagrams -by inheritance- activity nodes (e.g., actions) and activity edges (e.g., transitions) within an activity diagram.

The class attributes are mapped to stereotype tags: when defining a tag, a type needs to be associated to it. Indeed, at user model level, the stereotyped model element will be characterized by tagged-values, where the values should conform to the type assigned to the tag. The types assigned to the tags are basic NFP types of the MARTE library, such as *NFP_Duration* as type of *duration* tag.

We have applied the reference association pattern [35] to the associations of the domain model whose association-ends are explicitly named. In particular, *acquire* and *release* association-ends of the associations between *Resource usage step* and *Resource type* classes (Fig. 2) are mapped to homonyms tags of the *ResourceUsageStep* stereotype. The type defined for the tags is the stereotype corresponding to the referenced class, i.e., *ResourceType*. The association-end multiplicity and the *order* constraint is preserved in the mapping. The same pattern applies to *create*, *read*, *check*, *useID* and *use* association-ends (associations between *HealthInfoUsageStep* and *HealthInfo* classes), resulting in the homonyms tags of the *HealthInfoUsageStep* stereotype. Finally, the tags *resources* (*ResourceType* stereotype), *records* (*HealthInfo*) and *patients* (*TreatmentRequest*) -in Fig. 3- trace back to the homonym association ends related to the grey classes in the domain model (Fig. 2). The purpose of these tags is to keep

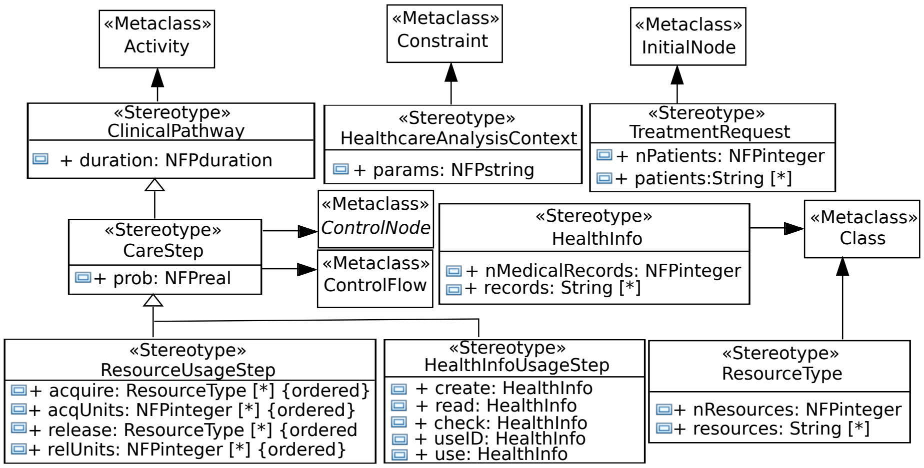


Figure 3: UML extensions for healthcare modeling and analysis.

trace of the fine-grained data by referencing external sources, so a general string type is associated to them.

5 Model-to-model transformation and analysis

5.1 Transformation to Stochastic Well-formed Net

The HSS profile described in Subsec. 4.2 permit us to obtain models for a healthcare analysis context (i.e., define the available resources, healthcare information and a set of clinical pathways). However, the obtained models cannot be directly analyzed to assess qualitative and quantitative properties using formal techniques. For this reason, we propose a Model-to-Model (M2M) transformation technique to get automatically a Stochastic Well-formed Net (SWN) model. M2M transformation techniques constitute the pillars of the Model-Driven Engineering (MDE) paradigm [39], where *target* models are automatically generated from *source* models by applying transformation rules which are defined on the source and target *meta-models*.

The input artifacts of the proposed transformation are:

1. *The source and target modeling languages*: the *source* language is the UML and the HSS profile, specified in Fig. 3; whereas the *target* language is the SWN formal language.
2. The *UML models*: constructed according to the *source* modeling language and consisting of:
    - an UML class diagram modeling the hospital assets (available resources and healthcare information);
    - a (set of) UML activity diagrams modeling the clinical pathways.

On the other hand, the output artifact is the SWN model that will be used for the qualitative and quantitative analysis (i.e., the *target model*).

As it was mentioned, clinical pathways will be defined as activity diagrams capturing the activities control flow (sequential execution, alternate paths) through the activity/control nodes and edges. Notice that the elements of the UML activity diagram ensuring the parallel execution of activities are not included, limiting the systems that can be modeled by our approach. In particular, the modeling methodology proposed in this paper can be used to model the flow of treatments, medical tests and cures that a patient should follow according to the clinical pathways in order to get healthier. Other aspects, as for example decisions of different medical test that can be taken in parallel are not possible to model. The motivation of not using these elements is to provide a modeling language with the smallest set of primitives that: (i) comply with the requirements of the final users, i.e., the medical doctors, for the specification of clinical pathways, and (ii) are easy to learn and use by the medical doctors. As by-product, the formal models generated with the transformation have well-defined structures, thus efficient structural analysis techniques can be applied. Nevertheless, this does not means that our models are sequential, the concurrency is appearing due to the parallel execution of different clinical pathways competing for common hospital resources. The activity nodes of the AD diagram that we consider are:

- *Action node* (represented as rounded rectangles), e.g., *Evaluate before hospitalization* in Fig.7;
- *Choice node* (represented as a diamond) models decision with alternate paths, e.g., *Are there any infections?*;
- *Merge node* (also represented as a diamond) is used to unify the alternate paths;
- *Initial node* (shown as a solid circle) represents the beginning of a clinical pathway;
- *Final node* (shown as a solid circle with a hollow circle outside) models the end of a clinical pathway.

The set of transformation rules is shown in Fig. 4 and 5, where the first column provides the rule identifier and its description, the second column depicts the UML-HSS model elements of the source model and the third column shows the SWN model elements of the target model. In the second column the bold-italic font is used to indicate the element identifiers, whereas in the third column the bold-italic font is used to indicate the labels associated SWN places.

The SWN model is obtained through a three steps-transformation, that will be explained in the following. The reader can refer to the appendix for the definition of the SWN formalism and the syntax used.

Step 1. Transformation rules in Fig. 4 that are divided into two groups depending on the type of diagrams on which they are applied.

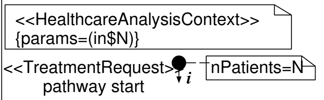
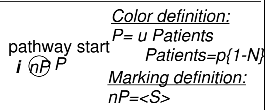
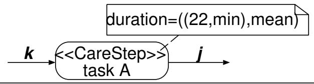
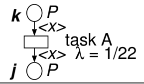
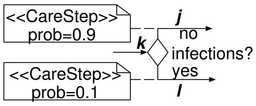
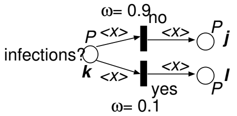
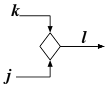
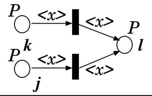
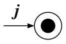
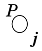
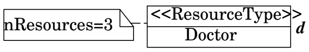
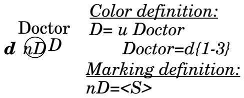
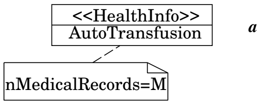
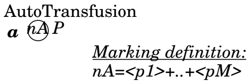
Rule	***UML-HSS model element (source)***	***SWN model element (target)***
[R1] Analysis context parameters and treatment request (quantity)		
[R2] Task (duration)		
[R3] Alternative choice (probability)		
[R4] Branch merging		
[R5] Final node		
[R6] Hospital resource (type and quantity)		
[R7] Patients' health information (type and quantity)		

Figure 4: Application of transformation rules (first step). Rules *R1-R5* are applied to the nodes of the activity diagram modeling the clinical pathway, whereas rules *R6-R7* are applied to the classes of the class diagram modeling the available resources and healthcare information.

- Rules applied to the activity nodes of the AD modeling the clinical pathways. The identifiers associated to the activity nodes are used in the transformation to label the places of the SWN model. The SWN models obtained from the application of the rules will be composed via place merging in the second step of the transformation to get the global model. All the SWN places produced by the rules *R1-R5* have the set of patients P as color domain.
  - *Rule R1* maps the initial node of the activity diagram (stereotyped *TreatmentRequest* in Fig. 4) into a single SWN place with a label ***i***, that is the identifier of the activity edge leaving the initial node. A color domain P and an initial marking nP is also set to the SWN place, according to the *nPatients* tagged-value. In particular, P is the color class consisting of a unique static color subclass $Patients$, which includes N different colors $p_1 \ldots p_N$ -each color corresponds to a patient identifier- and the initial marking nP corresponds to a set of N tokens, one per each color p_i. Observe that N can be a natural number or a parameter.
  - *Rule R2* transforms an action node *task A* into a SWN subnet consisting of a timed transition with an input place and an output place. The input and output places are labeled with the identifiers of the *task A* incoming and outgoing edges, respectively (denoted ***k*** and ***j*** in Fig. 4). Since the *task A* is a *CareStep* with an associated mean duration (e.g., 22 minutes in Fig. 4), the firing rate of the corresponding SWN transition is set to the inverse of the duration value. The expression assigned to the input and output arcs of the transition *task A* corresponds to the projection function $\langle x \rangle$, so the firing of the transition removes a colored token from the input place and adds it to the output place.
  - *Rule R3* transforms a choice node to a pair of conflicting SWN immediate transitions. The rule represents a probabilistic choice and the *prob* tagged-values, associated to the outgoing edges of the choice node, are used to set the weights of the SWN immediate transitions. The label of the SWN places are identical to the labels of the input/output activity edge (***k***, ***j*** and ***l***, respectively).
  - *Rule R4* transforms a merge node to a SWN subnet that unifies alternative flows: in particular, the places of the subnet are labeled with the identifiers of the incoming and outgoing activity edges (***k***, ***j*** and ***l***, respectively).
  - *Rule R5* is applied to final nodes of an activity diagram. Each final node is mapped to a single SWN place labeled with the identifier of the incoming activity edge.
- Rules applied to the classes of the class diagram modeling the available resources and the healthcare information.
  - *Rule R6* maps a class, stereotyped *ResourceType*, to a single SWN place: the place name is set to the class name (e.g., *Doctor*) and the place label is set to the class identifier (e.g., ***d***). The *nResources* tagged-value is used to define the color domain D and the initial marking nD of the place, where

D is the color class consisting of a unique static color subclass $Doctor$ which includes 3 different colors d_1, d_2 and d_3 -each color corresponds to a doctor identifier- and the initial marking nD corresponds to a set of three colored tokens, one per each color d_i. For each resource type, a different color class is defined.

- Finally, the *rule R7* is similar to *R6* but applied to the classes stereotyped *HealthInfo* to map the type of patients' health information. However, in this case, the color domain of the SWN place represents the set of patients, thus it is equal to the color domain P assigned to the SWN places created with the rules *R1-R5*. The initial marking of the place *nA* is set according to the *nMedicalRecord* tagged-value. For example, in Fig. 4, the initial marking corresponds to the subset of patients included in the auto-transfusion program.

Step 2. Composition of the SWN elements from Step 1. The resulting SWN subnets are composed by merging the places with common labels [9]. The resulted SWN model represents the control flow of the patients according to the clinical pathway modeled by the activity diagram. Additionally, the SWN model includes isolated places (derived from rules *R6* and *R7*) that represent the hospital resources and patients' health information.

Step 3. Resource assignment and healthcare information manipulation by rules in Fig. 5. The composed SWN model is expanded using the rules in Fig. 5 to consider resource acquisition/release and patients' healthcare information manipulation/usage (i.e., create, read, check, useID or use).

The rules in Fig. 5 are applied to all activity edges (rules $R8$ to $R12$) or activity decision nodes (rule *R13*), in the activity diagram modeling a clinical pathway, that are stereotyped *ResourceUsageStep* or *HealthInfoStep*. Remember that these stereotypes are defined in the HSS profile, and they are applied to the AD model elements to specify the resource utilization and information interchanged. The identifier of the activity edge, in rules $R8$ to $R12$, is denoted by ***j***, and the source node of the edge can be an initial, decision or merge node, whereas the target node of the edge can be a decision, merge or final node. The graphical representation of these nodes in Fig. 5 represents all these possibilities. Finally, rule $R13$ is applied only to decision nodes stereotyped *HealthInfoStep*.

- *Rule R8* is related to an activity edge, stereotyped *ResourceUsageStep*, with *acquire* and *acqUnits* tagged-values. This rule allows the allocation of a number of *o* instances of a resource (arc expression $\langle r1\rangle + \ldots + \langle ro\rangle$) and consists in adding before place ***j*** in the input SWN model (i.e., the SWN model before the application of this rule) an immediate transition and a place. The immediate transition removes from the corresponding resource place as many tokens as specified by the *acqUnits* tagged-value, i.e., o. In Fig. 5, place ***j1*** and transition *acqRes* are added for the allocation of resource *Resource*. The place ***j1*** inherits the input arc and the color domain of the place ***j*** before the application of this rule. We

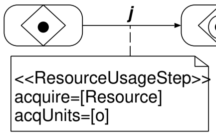
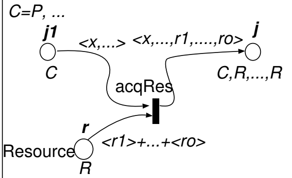
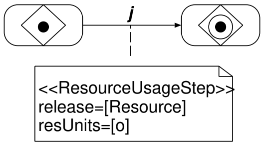
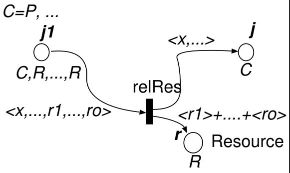
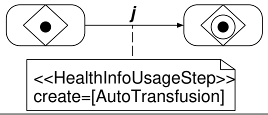
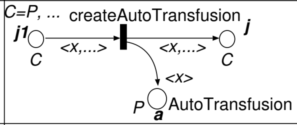
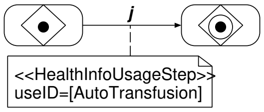
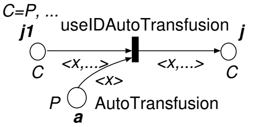
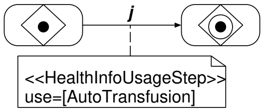
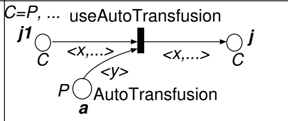
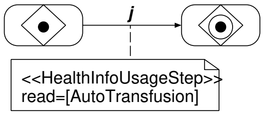
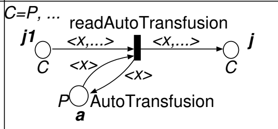
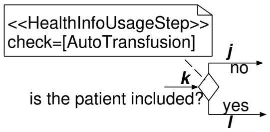
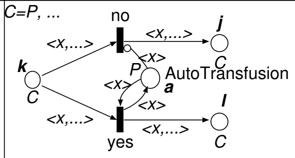
<table>
<tr><th>Rule</th><th>UML-HSS model element (source)</th><th>SWN model element (target)</th></tr>
<tr><td>[R8] Use of resources (allocation)</td><td>
</td><td>
</td></tr>
<tr><td>[R9] Use of resources (de-allocation)</td><td>
</td><td>
</td></tr>
<tr><td>[R10] Manipulation of patients' healths information (create)</td><td>
</td><td>
</td></tr>
<tr><td>[R11a] Use of patients' healths information (useID)</td><td>
</td><td>
</td></tr>
<tr><td>[R11b] Use of patients' healths information (use)</td><td>
</td><td>
</td></tr>
<tr><td>[R12] Use of patients' healths information (read)</td><td>
</td><td>
</td></tr>
<tr><td>[R13] Use of patients' healths information (check)</td><td>
</td><td>
</td></tr>
</table>

Figure 5: Application of transformation rules (second step).

denote such color domain C, a shorthand notation for a Cartesian product of color classes that includes the basic color class P (the set of patients). The color domain of place ***j*** and all the places that follow in the path will be replaced with the Cartesian product of the color domain C and the o color domains of place *Resource*.

- *Rule R9* is the complementary of rule *R8* that considers the resource release (*ResourceUsageStep* stereotype, *release* and *relUnits* tagged-values). Again, rule *R9* modifies the initial definition of the color domain associated to the places of the control flow, in order to keep track of the resources allocated to each patient.
- *Rule R10* is related to an activity edge stereotyped *HealthInfoUsageStep* with *create* tagged-value. This tagged-value is used to create a new medical record. Before the original place ***j***, a new place called ***j1*** and an immediate transition are added to the SWN model. The input arc of the original node ***j*** will be the input arc of the new node ***j1***. Moreover, the immediate transition is connected to the place corresponding to the *create* tagged-value (an *HealthInfo* place, see rule *R7*) such that the firing of this transition will produce a token in such place with a color representing the patient in place ***j1***. This rule permits for example to create a medical record containing the patients following the clinical pathway of auto-transfusion program.
- *Rules R11a* and *R11b* -usedID and use operations- permit the use of the existent information in the medical records. Both rules require the introduction of a new place ***j1*** and an immediate transition with an input arc coming from the *HealthInfo* place specified by the *useID* (or *use*) tagged-value. The difference between the rules *R11a* and *R11b* is that, in the former, the identity of the patient is considered while in the latter is not. This is reflected in the expression of the arc between the patients' health information place (*AutoTransfusion*) and the immediate transition: the rule *R11a* defines the synchronization based on the token color (i.e., the other input arc of the transition includes the same variable x in the arc expression), whereas the rule *R11b* defines the synchronization based on the presence of a token (i.e., a different variable y is used in the arc expression).
- *Rule R12* is related to a read operation and maps the *HealthInfoUsageStep* annotation to an immediate transition with a test arc (i.e., input-output arc) to the health information place specified by the *read* tagged-value. The read operation may appear in the specification of a clinical pathway when it is necessary to check if a patient followed other clinical pathway (in this case the auto-transfusion pathway).
- Finally, the *rule R13* is related to a check operation and specifies a boolean condition (e.g., whether the patient is included in the list of the auto-transfusion program or not). It is mapped to a pair of non-free choice conflicting transitions. Both transitions are connected to the place representing the patients' health information: one transition models the *true* value of the condition and it is connected with a test arc, the other transition models the *false* value and it is connected with an inhibitor arc.

5.2 Analysis of clinical pathways: an approach based on facets

Public health systems are complex systems due to their size and their intricate structure. Such systems interact with different types of stakeholders such as managers of different levels, patients, Governments or regulatory authorities. Each of these groups has different concerns, visions and expectations with respect to the behavior of the system. The relevant aspects of the behavior of the system for each stakeholder are that we call *facet* of the system, and so the overall system is a multi-faceted system.

The SWN model, automatically generated from the UML-HSS specification of the clinical pathways via M2M transformation, has the capacity to represent the multi-faceted nature of a complex healthcare system. This model represents the perspective based on the deployment and operation of a concurrent set of clinical pathways interacting throughout the competition for shared resources and interchanging information to impose causality relations. The methodology proposed is to use the SWN model as the multi-faceted model from which we are able to obtain particular models for each facet.

Facet 1: the decolorized facet. The first proposed facet is obtained from the decolorization (i.e., removing "irrelevant" differences among individuals making them "indistinguishable") of the SWN model. The approach that we use for decolorization is very simple and basically assumes that differences between individuals are removed. It can be noticed that in the PN literature exist methods for decolorization based on unfolding [18] that provide PN systems without colors providing the same behavior as the original color net. Since in the SWN model may appear synchronizations between tokens (e.g., after applying rule [R11a] the resulted immediate transition fires only if the tokens in the input places correspond to the same individual), for each possible combination of individuals a new transition is added in the decolorized PN. However, since in healthcare domain, the population (number of patients, of medical doctors, etc.) could be very high, we propose a different decolorized facet that is described as follows.

- All the places added by applying rules [R1]–[R12], will not have any color domain assigned (all of them will contain indistinguishable tokens) and all arcs inscriptions are removed. Notice that this means that rule [R11a] and [R11b] will provide the same output and the synchronization appearing after [R11a] in the SWN model is removed. Observe that the synchronization is also lost by decolorizing rule [R12].
- The decolorization of [R13] considers that the choice between the two transitions is probabilistic according to the transition weights. Additionally, inhibitor and normal arcs connecting place modeling the health information (i.e., AutoTransfusions) are removed. Indeed, in the example discussed in Sub-section 6.1, the rule [R13] has been decolorized by considering the initial marking of the places "start" (number of patients N) and "Auto-transfusion" (the number of patients M included in the auto-transfusion program), i.e., the immediate transitions have weights M/N and $1 - M/N$ (that, in the particular case, are both equal to 0.5, since $M = 50\%N$).

This facet can be used for quantitative analysis purposes, such as to compute performance bounds for the cycle time as in Sub-section 6.1.

Facet 2: the resource management facet. If one wants to study the use of resources (quantitatively or qualitatively) a relaxation can be obtained from the *Facet 1* by removing all places (together with the connected arcs) modeling the health information (e.g., auto-transfusion in the considered case study). Notice that these places model message channels and, from the point of view of resource utilization, can be ignored when the model is analyzed. This facet can be used to study the utilization of the resources, the correct assignment and release of resources, etc.

Facet 3: the handshake between clinical pathways facet. Clinical pathways are interacting between them by using the health information, normally through medical records. Health information used in a clinical pathway is translated to a colored place in the M2M transformation according to rule [R7]. This facet considers only the flow of patients and the information about the health information. Hence, this facet can be obtained from the *Facet 1* by simply removing all the places (together with the connected arcs) modeling resources.

An example of the use of the last two facets is considered in Sub-section 6.2.

6 Application of the methodology

We first apply, the steps to obtain a SWN from the pathway used for hip fracture in the University Hospital “Lozano Blesa” of Zaragoza (Spain). The *Facet 1* (decolorized facet) is obtained and used in Subsec. 6.1 for the performance analysis. Then, we exemplify the application of the *Facet 2* (resource management facet) and *Facet 3* (handshake between clinical pathways) for the qualitative analysis in Sub-section 6.2 using a simpler example.

6.1 Performance analysis of the case study

In the following, we consider the case study of the clinical pathway used for the hip fracture in the University Hospital “Lozano Blesa” of Zaragoza (Spain). First, we apply the HSS profile, defined in Section 4, to specify the performance parameters and resource constraints. Then, we show the SWN model obtained from the M2M transformation, defined in Sub-section 5.1. Finally, we analyze the performance model, obtained by decoloring the SWN model, using net-level efficient techniques.

Use of the HSS profile to specify clinical pathways. The purpose of the HSS profile is to model clinical scenarios -which rely upon hospital assets- using UML class and activity diagrams, with the aim of analyzing them from a qualitative and a quantitative perspectives.

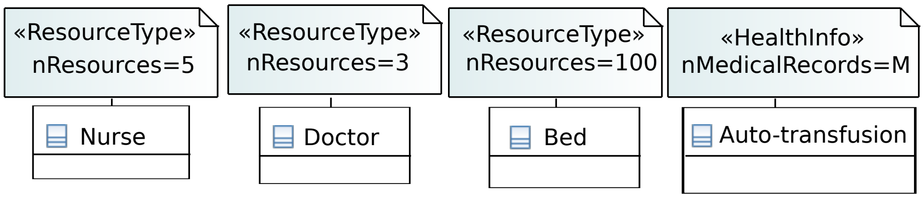


Figure 6: Structural view: hospital assets.

Fig. 6 and Fig. 7 exemplify the application of the HSS profile for the case study. In particular, Fig. 6 shows the set of hospital assets, that includes the hospital resource types -the three classes stereotyped *ResourceType* i.e., bed, doctor, and nurse- and the patients' health information types -the class stereotyped *HealthInfo* i.e., auto-transfusion.

We use the UML note symbol to show explicitly the tagged-values associated to a given model element. However, when an UML tool with profiling facilities is used, the tagged-values can be easily set via a GUI. This is the case, for example, of the Eclipse Papyrus tool [1] which has been used to implement the HSS profile as a plug-in and to model the case study.

Fig. 7 models the clinical pathway used for hip fracture: for readability reasons, only a subset of tagged-values are shown as annotations. This pathway includes all the tasks to be accomplished during the day of hospitalization (left side of the figure) and the post-operative day in the hospital (right side of the figure). Between the two parts, there is the day-D of the surgical intervention.

By following the pathway -from the pre-operative day to the post-operative day- it is possible to estimate the mean time spent for a patient to undergo the treatment. Such performance index is specified as a model parameter CT in the *duration* tagged-value of the *ClinicalGuideline* (note symbol at the top-left side). A parameter N is also used to set the number of patients to be treated (*TreatmentRequest* annotation attached to the initial node of the activity diagram). The model input and output parameters are declared in the *HealthAnalysisContext* constraint (note symbol at the top-center).

Considering the day of hospitalization (left part), a prerequisite to perform the pathway for a given patient is the availability of a bed that will be assigned to the patient during the hospitalization stay. A nurse is also required to perform several care steps; then, both the bed and nurse requirements are specified by stereotyping as *ResourceUsageStep* the arc connecting the initial node to the "Evaluate before hospitalization" step and assigning the corresponding values to the *acquire* and *acqUnits* tags. The first three steps, carried out by the nurse, consist in monitoring the health state of the patient waiting for the surgery. In particular, the nurse has to determine if the patient has new pathologies or not. The pathway considers the case that there are not pathologies or they are compatible with the intervention for hip fracture and controlled, and then the next step performed by the nurse is to check the presence of urinary infections in the previous 4-6 weeks before hospitalization. The two arcs outgoing the decision node "Are there any infections?" are annotated with a probability modeling the case where 10% of the treated patients suffer urinary infections. If in-

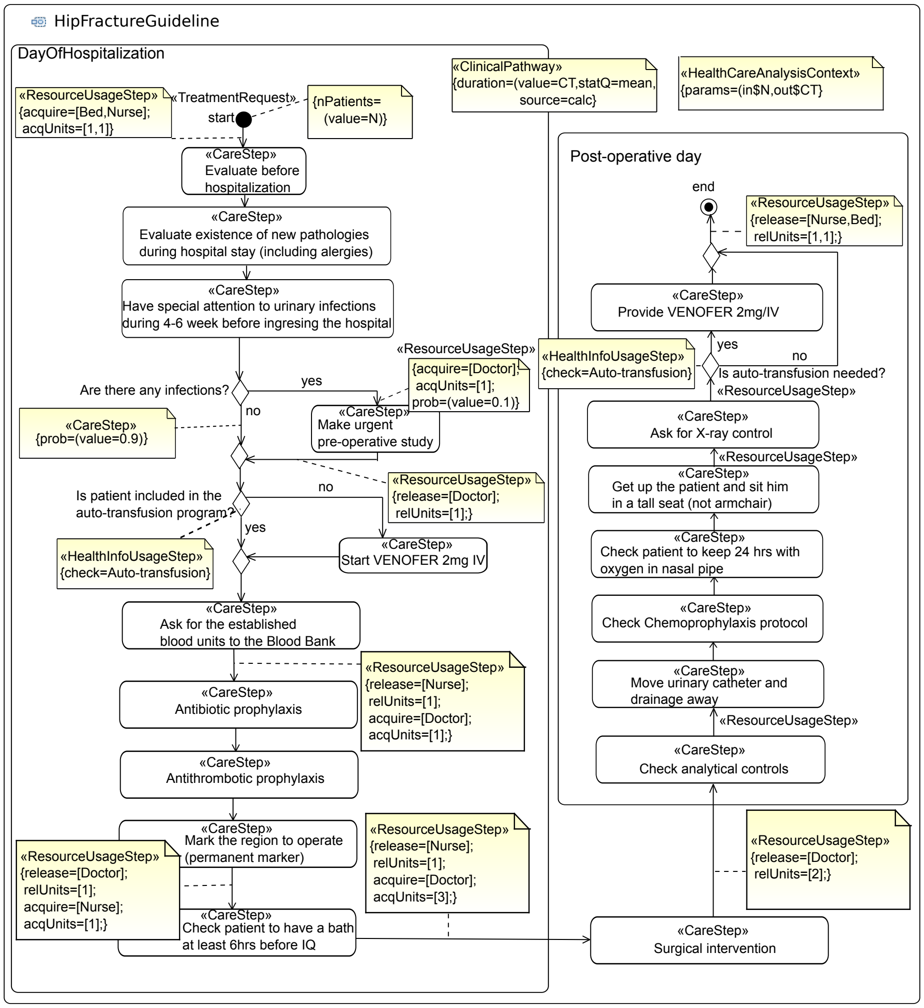


Figure 7: Behavioral view: an hip fracture clinical scenario.

fections are detected ("yes" arc), then a medical doctor does an urgent pre-operative study of the patient; which can include anamnesis, physical exam and analytical tests (e.g., urine exam). Observe that the "no" arc is stereotyped as *CareStep* since only the *prob* tagged-value needs to be specified, while the "yes" arc is stereotyped as *ResourceUsageStep* to specify -besides the *prob* tagged-value- the assignment of a doctor to the *Make urgent pre-operative study* step. The *acquire* and *release* tagged-values are used to allocate and deallocate, respectively, a doctor to this care step.

The next step carried out (by the nurse) is to determine whether the patient is included in the auto-transfusion program: the decision node "Is patient included in the auto-transfusion program?" is stereotyped as *HeathInfoUsageStep* to model a decision step based on the available information (*check* tagged-value). In case that the patient is not included in the auto-transfusion program, the nurse has to start the treatment with Venofer 200 mg intravenous. The nurse also asks for the reserved blood units from the Blood Bank of the hospital. Then, a medical doctor is in charge of providing antibiotic prophylaxis and anti-thrombotic prophylaxis with HBPM (Low molecular weight heparin) as well as to mark the region to be operated with a permanent marker. Finally, the last task to be performed by a nurse in the pre-operative day is to check that the patient has had a shower in the previous six hours before the surgery. The day-D of surgical intervention requires three medical doctors.

The second part of the pathway is the post-operative day (Fig. 7 - right part). The patient has been operated and now she/he is recuperating in the hospital bed that was assigned to her/him the day of hospitalization. The steps in the second part are: checking analytical controls (doctor), moving urinary catheter and all drainage away (nurse), checking the accomplishment of the chemoprophylaxis protocol (nurse), administrating oxygen in nasal pipe for 24 hours (nurse), mobilization of the patient (nurse), and asking for a control X-ray (doctor). In the case that the patient was no suitable for auto-transfusion, the nurse provides the second dosis of Venofer 200 mg intravenous.

Automatic generation of an analyzable SWN model. The SWN model of the clinical pathway of Fig. 8 has been obtained through M2M transformation by applying the rules of Figs. 4 and 5. The PN subnets enclosed in the dotted rectangles correspond to the two parts of the pathway, while the places outside the rectangles represent the initial available resources- i.e., 100 beds, 5 nurses and 3 medical doctors (cf. rule *R6* in Fig. 4)- and the patients' health information -i.e., M medical records of patients included in auto-transfusion program (cf. rule *R7*). PN transitions, depicted as thin black bars, may represent: allocation or release of resources (cf. respectively, rules *R8* and *R9* in Fig. 5), probabilistic choices (cf. rule *R3* in Fig. 4) or informed choices according to patients' health information (cf. rule *R13* in Fig. 5), the finalization of the protocol, i.e., place *end* (cf. rule *R5* in Fig. 4).

The remainder transitions, depicted as thick white bars, model tasks (cf. rule *R2*). Observe that the SWN model is parametric with respect to the number of patients N treated according to this pathway (initial marking of place *start*) and the number of patients M included in the auto-transfusion program (initial marking of place *Auto-transfusion*).

Performance analysis using the decolorized facet. The purpose of the performance analysis is to provide a support to the hospital manager in the resource planning. In particular, in this study we are interested in finding how many doctors and nurses are needed to guarantee that the sojourn time for a patient does not exceed three days (i.e., the pre-operative day, the day of the surgical intervention and the post-operative day) considering different patients workload assumptions.

To this aim, we use net-level efficient performance techniques for timed and stochastic Petri Net models based on the computation of bounds [15, 16], that allow computing the mean time spent for a patient to undergo the treatment. In particular, the mentioned techniques compute bounds of basic performance indexes of a timed or stochastic Petri Net model, such as the transitions throughput and the transition cycle time, through the generation and solution of a linear programming problem. The latter includes a set of linear constraints that are defined considering the structural specification of the net (i.e., incidence matrix and initial marking) and the timing and flow-routing specification. The LP formulation is provided in the Appendix 8.3. Therefore, such techniques are a viable alternative to state-based and Monte Carlo event-driven simulation techniques especially in case of highly populated models, such as the clinical pathways models, often characterized by thousands of patients to be treated and hundreds of resources to be allocated. Although extensions of such techniques to SWN models have been proposed [16], the current available implementations [29, 8] are restricted to uncolored timed (or stochastic) and *closed* models. Thus, we apply first the decolorization facet, explained in Subsec. 5.2, to convert the SWN model of Fig. 8 into a GSPN model and we add an immediate transition that connects the final place *end* with the initial place *start*.

Table 1 summarizes the parameters setting for performance evaluation purposes: according to the transformation rules *R2-R3* of Fig. 4, mean durations have been associated to the transitions modeling tasks and probabilities have been assigned to transitions representing alternative steps. The table also includes the probabilities associated to the two pairs of free-conflicting transitions (i.e., the last two rows emphasized in gray), that have been derived by applying the decolorization facet to the deterministic conflicts involving the patients' health information place *Auto-transfusion*, where the initial marking is set to $M = 50\%N$ and N is the patients workload (cf. Fig. 8).

The performance index considered in the analysis is the cycle time of transition t_{acqBed} (cf. Figure 8 and Appendix 8.3) multiplied by the number of patients N -i.e., $CT = CT^{acqBed}N$- that in the clinical pathway corresponds to the mean time spent for a patient to undergo the treatment. We have considered, as basic value of reference, the CT value when there is only one patient to be treated, i.e, $N = 1$. In such a case, the value computed by the bound solver [8] using the GSPN model is 0.169 days, i.e., ≈ 4.062 hours, whereas the value estimated by the event-driven simulator using the SWN model [29] is 0.170 days. Indeed, the former represents a lower bound due to the decolorization facet, nevertheless the relative error is not very relevant, i.e., 0.41%.

Transition	**Duration (min)**
Evaluate before hospitalization	00:15
Evaluate existence of new pathologies	00:01
Have a special attention to urinary infections	00:01
Make urgent pre-operative study	00:30
Start VENOFER	00:05
Ask for blood units	00:01
Antibiotic prophylaxis	00:01
Antithrombotic prophylaxis	00:01
Mark the region to operate	00:05
Check patient to take a bath	00:01
Surgical intervention	02:00
Check analytical controls	00:30
Move urinary catheter and drainage away	00:15
Check chemoprophylaxis protocol	00:05
Administrate O_2 in nasal pipe	00:05
Get up patient and sit him in a tall seat	00:30
Ask for X-ray control	00:05
Provide VENOFER	00:05
Transition	**Probability**
infections / no infections	$10\%/90\%$
included / not included	$50\%/50\%$
auto-transfusion / no auto-transfusion	$50\%/50\%$

Table 1: Timing/probabilistic specification.

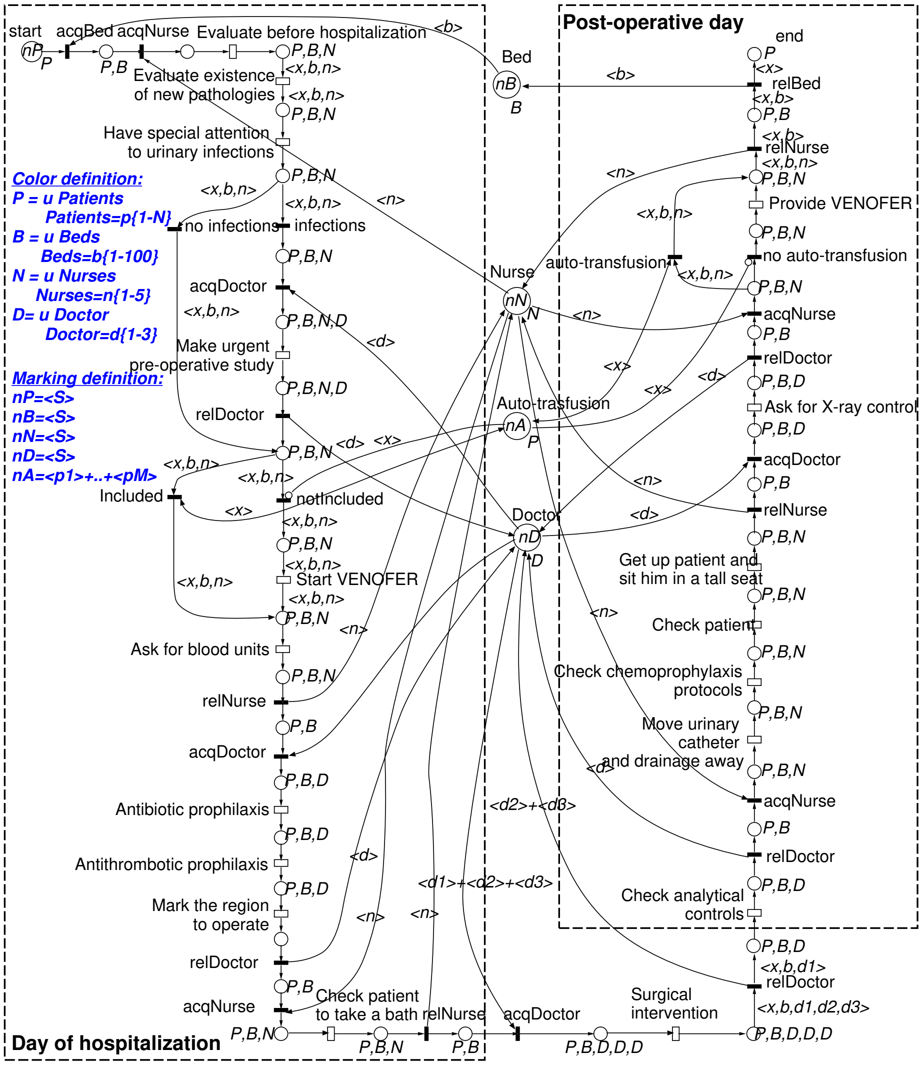


Figure 8: SWN model of the clinical pathway.

N.row	Patients	Doctors	Nurses	GSPN bound technique CT (days)	Critical resource	Sol. time (min:sec.ms)	SWN simulation CT (days)	Sol. time (min:sec.ms)	Rel. Err (%)
1	100	3	5	9.369	doctor	0:00.006	10.620	0:11.037	11.78%
2	76	3	5	7.121	doctor	0:00.006	8.105	0:08.027	12.14%
3	50	3	5	4.685	doctor	0:00.007	5.353	0:06.015	12.50%
4	26	3	5	2.436	doctor	0:00.007	2.744	0:03.047	11.21%
5	2	3	5	0.187	doctor	0:00.006	0.238	0:00.035	21.29%
6	1	3	5	0.169	pathway	0:00.011	0.170	0:00.019	0.41%
7	100	6	5	4.684	doctor	0:00.006	5.105	0:59.002	8.24%
8	76	6	5	3.560	doctor	0:00.006	3.986	0:40.084	10.69%
9	50	6	5	2.342	doctor	0:00.006	2.588	0:29.060	9.48%
10	26	6	5	1.218	doctor	0:00.006	1.371	0:15.013	11.14%
11	2	6	5	0.169	pathway	0:00.006	0.172	0:00.049	1.60%
12	100	9	5	3.123	doctor	0:00.006	3.367	2:12.084	7.25%
13	76	9	5	2.373	doctor	0:00.005	2.596	1:38.079	8.57%
14	50	9	5	1.561	doctor	0:00.006	1.686	1:05.088	7.41%
15	26	9	5	0.812	doctor	0:00.006	0.869	0:36.084	6.60%
16	2	9	5	0.169	pathway	0:00.006	0.169	0:03.073	0.00%
…	…	…	…	…	…	…	…	…	
17	100	15	5	1.874	doctor	0:00.006	1.970	18:43.046	4.88%
18	76	15	5	1.424	doctor	0:00.006	1.511	15:36.086	5.78%
19	50	15	5	0.937	doctor	0:00.006	0.997	6:56.012	6.03%
20	26	15	5	0.487	doctor	0:00.007	0.513	3:20.072	5.01%
21	2	15	5	0.169	pathway	0:00.006	0.169	0:02.035	0.00%
…	…	…	…	…	…	…	…	…	
22	100	25	5	1.135	nurse	0:00.006	1.193	21:33.068	4.86%
23	76	25	5	0.863	nurse	0:00.007	0.889	09:38.084	2.93%
24	50	25	5	0.568	nurse	0:00.007	0.585	11:47.026	2.93%
25	26	25	5	0.295	nurse	0:00.007	0.325	01:46.066	9.25%
26	2	25	5	0.169	pathway	0:00.007	0.170	01:28.020	0.19%

Table 2: Performance results under different assumptions of patient workload (2^{nd} column) and resource plan (3^{rd}-4^{th} columns): the performance index of interest (i.e., CT) is the mean time spent for a patient to undergo the treatment. CT values have been estimated using GSPN bounding techniques (5^{th} column) and SWN simulation (8^{th} column). The columns 7^{th} and 9^{th} report the corresponding solution times, and the last column shows the relative error between the CT values estimated with the two techniques. Besides, the GSPN bound techniques enables to identify the slowest subnet of the model (6^{th} column). Finally, the rows are highlighted with a different color to indicate whether the corresponding resource plan is not acceptable (red), acceptable (green) or further analysis is needed (yellow).

It is worth noticing that the bound solver computes, besides the optimal value of the LP problem, also an optimal solution that enables to determine the slowest subnet of the GSPN model (i.e., indeed such a subnet is the one generated by the minimal P-invariant that is an optimal solution of the LP problem). Thus, considering the structural characterization of the GSPN model (i.e., its minimal P-invariants), the slowest subnets can correspond either to a given type of resource or to the clinical pathway. In particular, when $N = 1$ the slowest subnet is the clinical pathway, indicating that the CT basic value is not affected by the resource (e.g., beds, doctors and nurses) unavailability.

The CT value computed in case of one patient (i.e., $N = 1$) constitutes the baseline for the sensitivity analysis. Indeed, we are interested in analyzing the increase of CT values considering different patients workload assumptions. In particular, we consider the patient workload in the interval $[2, 100]$ and different personnel resource plans: Table 2 shows the CT values obtained by applying the bound technique on the decolored model (column 5) and the event-driven simulation on the original SWN model (column 8).

The table also reports the time to get the performance values (columns 7 and 9) and the relative error between the performance values computed with the two techniques (last column). The latter has been calculated considering the results estimated with the event-driven simulation of the SWN model as reference values (i.e., rel.err = $\frac{CT_{sim}-CT_{bound}}{CT_{sim}}100\%$). Concerning the computation time, the bound technique is more efficient than the simulation one, since it is not influenced by the initial configuration of the model (initial marking). On the other hand, from the relative error values, we can infer that the bound technique provides a lower bound for the performance index, with respect to the simulation results, which should be taken into account in the final assessment. The relative error is in the worst case 21.29%, which occurs in one situation (i.e., $N = 2$, 3 doctors and 5 nurses), for the rest of the experiments it is below 13%.

With the help of the domain experts, we have defined a qualitative indicator that allows them to understand the performance results at a first glance and, therefore, to decide whether a given resource planning fulfills the patient sojourn time restriction (i.e., at most three days). The indicator can take one of the following values:

- *Fail*, when the CT value is greater than three days. The corresponding resource planning is considered not acceptable.
- *Uncertain*, when the CT is between one and two days. The corresponding resource planning deserves further investigation in this case, due to the fact that the bounding technique provides a lower bound for the performance index, as commented previously.
- *Ok*, when CT value is less than one day. The corresponding resource planning can be accepted.

In the Figure 2, each row corresponds to a different model configuration, i.e., patient workload and number of resources, and it has been highlighted with a different color according to the value of the performance indicator, i.e., red, yellow and green indicate fail, uncertain and ok values, respectively.

From the analysis results, the hospital manager can discard the *fail* cases and accept the *ok* ones. However, since one of the goal of the analysis is to decide a cost effective resource plan, (s)he should consider in more detail the *uncertain* cases to decide whether accept or discard them. For example, when $N = 26$ patients have to be treated, 6 doctors (row 19) are not enough to comply with the restrictions and 9 doctors are needed (row 15). However the CT value, computed in the case of 6 doctors, is close to one day (i.e., $CT = 1.218$ days, that considering the relative error due to the lower bound approximation sums to $CT = 1.354$ days), which is the acceptance threshold.

Finally, observe that the experiments have been conducted by varying only the number of doctors for what concern the number of personnel (and not the number of nurses), since they represent the critical resource for most of the cases. When the critical resource is the clinical pathway, the best performance case is obtained. Indeed, the sensitivity analysis should be driven by the identification of the critical resources, that is the analyst should increase the number of the critical resource to improve the model performance. The last experiments (rows 22-25) show that the critical resource has changed from the doctors to the nurses, meaning that to comply with the performance requirements in the case $N = 100$ (row 22), the analyst needs to increment the number of nurses in the resource plan.

6.2 Qualitative assessment of clinical pathways

The analysis scenario is composed of two clinical pathways represented in Fig. 9 and Fig. 10. By applying the M2M transformation, we obtain the SWN in Fig. 11.

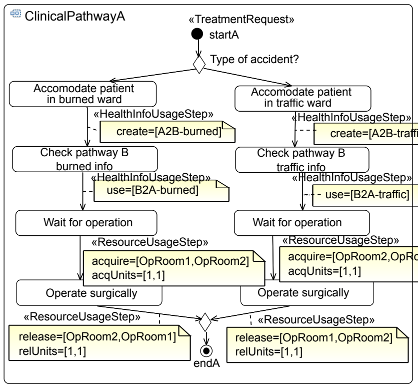


Figure 9: Clinical pathway A

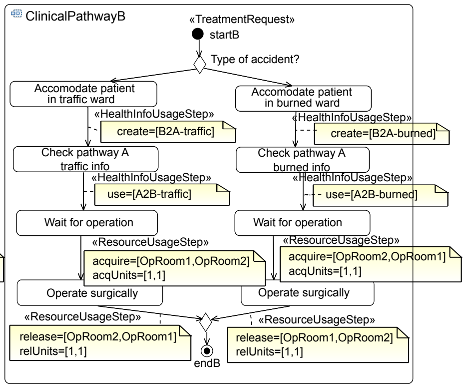


Figure 10: Clinical pathway B

Patients following each clinical pathway are accommodated in places $A0$ and $B0$, respectively. Tokens in place $A0$ are patient identifiers (from $p1$ to $p3$). The same is true for the $B0$ place, except that in this case we have two patients following this clinical pathway ($p4$ and $p5$). Both pathways are characterized by two alternative treatments according to the type of condition that has led a patient to the emergency room: burned in a fire or traffic accident. These two clinical pathways interact in two different ways. First, they compete for two operating rooms represented by the resource places *OperatingRoom1* and *OperatingRoom2*. How these resources are used by each one of the pathways is reflected by the structure of the SWN model depicted in the figure. Another present interaction requires the synchronization of both clinical pathways in such a way that both pathways treat patients with the same condition: burnt or traffic accident. Moreover, to begin the corresponding pathway, each patient expects other patient following the other pathway to start simultaneously. This synchronization is based on the fact that the two operating rooms must be prepared for the same type of patients to reduce the costs since the preparation is different in each case.

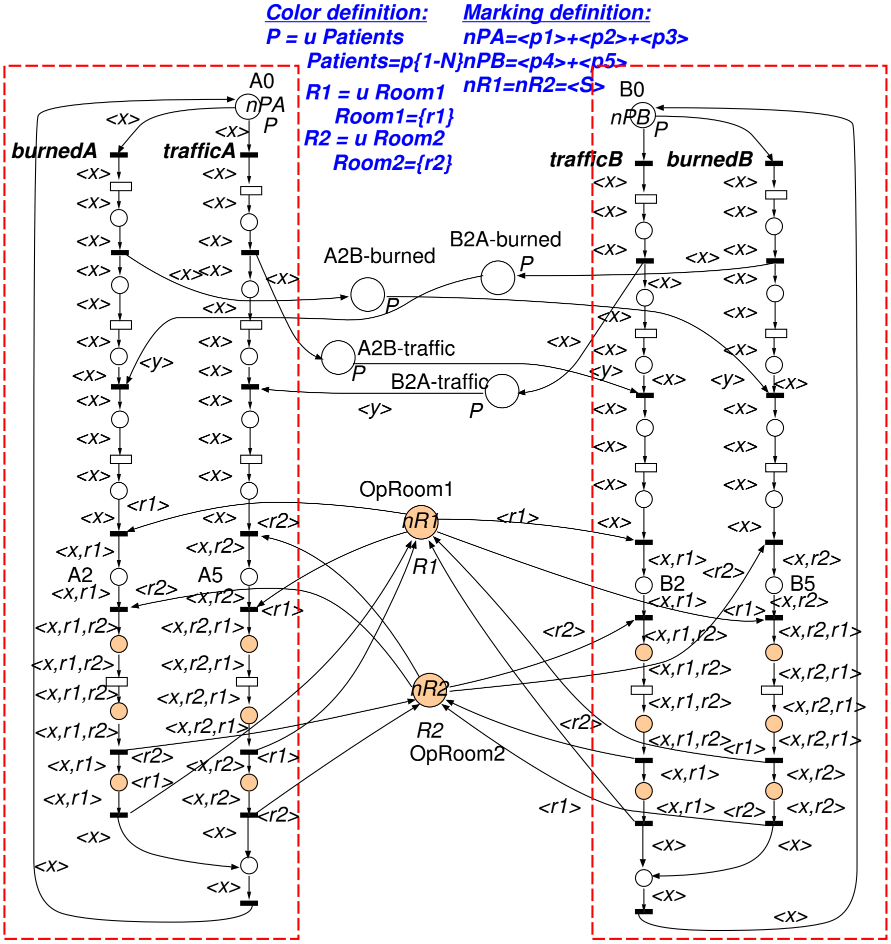


Figure 11: Two clinical pathways competing for the shared resources *OpRoom1* and *OpRoom2*, and interchanging information through places *A2B-burned*, *A2B-traffic*, *B2A-burned* and *B2A-traffic*.

Herein, we will consider two possible facets to be extracted from the model of the figure. The first aspect refers to the management of resources and the second refers to the synchronization protocol of both clinical pathways that makes both acts similarly on patients with the same condition. For the first case, *Facet 2* described in Subsec. 5.2 is used while for the second case, *Facet 3* described in the same Subsection is considered.

Resource management using facet 2. This aspect is obtained from the decoloured net of the figure by removing places *A2B-burned*, *A2B-traffic*, *B2A-burned* and *B2A-traffic* (representing the patients' health information exchanged between pathways - see rules *R10-11* of Fig. 5). The net obtained has a behavior equivalent to an S^4PR [50], a well known class of PN used in Resource Allocation Systems. Applying the theory of S^4PR nets, a deadlock state is identified due to the existence of a bad siphon defined by the orange places. This means that there are reachable states where the siphon becomes empty and remains empty forever. Therefore all output transitions of the places of the siphon are dead.

A way of correcting this problem consists of the addition of a place (monitor) to the net that prevents the emptiness of the siphon. This place should guarantee mutual exclusion amongst the places $A2$, $A5$, $B2$, $B5$. After this correction, liveness problems only can be caused by the interaction protocol but cannot be produced by the resource allocation.

The introduced correction has as disadvantage that sequentializes the use of operating rooms, which from the perspective of the rate of utilization of resources is not a good solution. For this reason, and to increase the implementation of concurrent operations and thus increase the efficiency of the system we can increase the number of operating rooms. Increasing a token in the resource place *OperatingRoom1* or *OperatingRoom2*, prevents the deadlocks, but only in the case of having a single token in $A0$ and a single token in $B0$. If there are more tokens in $A0$ and/or $B0$ then a deadlock state may be reached again.

This means that increasing the number of operating rooms only is effective in removing deadlocks if these new resources are dedicated exclusively to one pathway. Therefore, the choice of the correction strategy depends on the stakeholder who should take the decision.

Use of handshake between clinical pathways facet. This facet is obtained from the decoloured net by eliminating the places *OperatingRoom1* and *OperatingRoom2*. The resulting net belongs to the well known class of DSSP [44]. The analysis of this net allows us to determine that the protocol is bad designed. Indeed, if the clinical pathway on the left treats only burned patients (i.e., transition *burnedA* fires for all the patients) and the clinical pathway on the right treats only patients of traffic accident (i.e., transition *trafficB* fires for all the patients), the net reaches a deadlock since the pathways cannot be synchronized to treat both the same type of patient. The structural cause is that the two pathways take independent decisions before they synchronize. As decisions are free this is always possible. Correction is based on modifying the protocol so that decisions that take each clinical pathways are non-free, i.e., both decisions share patients' health information so that both resolve the choice in the same manner (see rule *R13* of Fig. 5).

The incorporation of the corrections -if needed- made in each one of the facet models to the original model ensures a live and correct model. These corrections will be propagated back to be included in the UML description in order to contain the correct specifications for the stakeholders.

7 Conclusions

This paper presents a methodology for the hospital management based on the modeling and analysis of clinical pathways. First, a graphical UML based model with few primitives is proposed allowing the medical doctors to model clinical pathways in an easy way. To support simulations of the system, the UML model is transformed into a mathematical one (by using SWN models) defining local rules. Finally, in order to use formal analysis and study different aspects of the healthcare system, the resulted model is decolorized and eventually relaxed to facets.

The methodology presented in the paper has been used to develop a software platform called HEAT, currently installed in the hospital [43] in order to monitor the patients following a clinical pathway and to model a new clinical pathway in HSS language. The main objective of a clinical pathway is for the medical doctors to do everything there is to do and nothing more, but all without increasing the time or complicating or hindering the work performed. The use of information technology to hospital clinical practice is common, since the medical history is now computerized in most hospitals, so the physicians are currently accustomed to use computer tools, hence, this new tool will be useful and also easy to learn and perform.

As future research lines, a deeper study is necessary to provide a methodology of obtaining facets in order to apply structural analysis. Another future work is to investigate on how to make automatic the last step of the proposed methodology, that is the presentation of analysis results in a form understandable for the medical doctors. Last but not least, the development of new software tools to help physicians create new clinical pathways is an objective that becomes important.

References

[1] Papyrus. `http://www.eclipse.org/papyrus/`

[2] Special issue on system dynamics for policy, strategy and management education. Journal of the Operational Research Society **50**(4) (1999)

[3] Aggarwal, V.: The Application of the Unified Modeling Language in Object-Oriented Analysis of Healthcare Information Systems. J. Med. Syst. **26**(5), 383–397 (2002)

[4] Aristovnik, A.: Measuring relative efficiency in health and education sector: The case of East European countries. Actual Problems of Economics **136**, 305 – 314 (2012)

[5] Ashiffman, R.N., Karras, B.T., Agrawal, A., Chen, R., Marenco, L., Nath, S.: GEM: A proposal for a more comprehensive guideline document model using XML. Journal of the American Medical Informatics Association **7**(5), 488 – 498 (2000)

[6] Augusto, V., Xie, X.: A modeling and simulation framework for health care systems. IEEE T. Systems, Man, and Cybernetics: Systems **44**(1), 30–46 (2014)

[7] Bernardi, S., Campos, J.: Computation of Performance Bounds for Real-Time systems using Time Petri Nets. IEEE Trans. Industrial Informatics **5**(2), 168–180 (2009)

[8] Bernardi, S., Campos, J.: A min-max problem for the computation of the cycle time lower bound in interval-based Time Petri Nets. IEEE Transactions on Systems, Man, and Cybernetics: Systems **43**(5), 1167–1181 (2013). DOI 10.1109/TSMCA.2012.2226442

[9] Bernardi, S., Donatelli, S., Horváth, A.: Implementing compositionality for stochastic Petri nets. STTT **3**(4), 417–430 (2001)

[10] Bosilj-Vuksic, V., Giaglis, G., Hlupic, V.: IDEF Diagrams and Petri Nets for Business Process Modeling: Suitability, Efficacy, and Complementary Use. In: B. Sharp, J. Filipe, J. Cordeiro (eds.) Enterprise Information Systems II, pp. 143–148. Springer, Dordrecht (2001)

[11] Boxwala, A.A., Peleg, M., Tu, S., Ogunyemi, O., Zeng, Q.T., Wang, D., Patel, V.L., Greenes, R.A., Shortliffe, E.H.: Glif3: a representation format for sharable computer-interpretable clinical practice guidelines. Journal of Biomedical Informatics **37**(3), 147 – 161 (2004)

[12] Brailsford, S.C., Harper, P.R., Patel, B., Pitt, M.: An analysis of the academic literature on simulation and modelling in health care. Journal of Simulation **3**(3), 130–140 (2009)

[13] Büyüközkan, G., Çifçi, G., Güleryüz, S.: Strategic analysis of healthcare service quality using fuzzy {AHP} methodology. Expert Systems with Applications **38**(8), 9407 – 9424 (2011)

[14] C.-P. Barros A.-G. Menezes, J.C.V.: Measurement of hospital efficiency, using a latent class stochastic frontier model. Applied Economics **45**(1), 47 – 54 (2013)

[15] Campos, J., Silva, M.: Structural techniques and performance bounds of stochastic Petri net models. Lecture Notes in Computer Science **609**, 352–391 (1992)

[16] Chiola, G., Anglano, C., Campos, J., Colom, J., Silva, M.: Operational analysis of timed Petri nets and application to the computation of performance bounds. In: O. Boxma, G. Koole (eds.) Performance Evaluation of Parallel and Distributed Systems: Solution Methods, *Tract*, vol. 106, pp. 197–213. Centrum voor Wiskunde en Informatica, Amsterdam (1994)

[17] Chiola, G., Dutheillet, C., Franceschinis, G., Haddad, S.: Stochastic well-formed colored nets and symmetric modeling applications. IEEE Trans. on Computers **42**(11), 1343–1360 (1993)

[18] Chiola, G., Franceschinis, G.: A structural colour simplification in well-formed coloured nets. In: Proc. of the 4^{th} Int. Workshop on Petri Nets and Performance Models, pp. 144–153 (1991)

[19] Connelly, L.G., Bair, A.E.: Discrete event simulation of emergency department activity: a platform for system-level operations research. Academic Emergency Medicine **11**(11), 1177–1185 (2004)

[20] Davies, R.: An assessment of models of a health system. Journal of the Operational Research Society **36**(8), 679–686 (1985)

[21] Dotoli, M., Epicoco, N., Falagario, M., Sciancalepore, F.: A cross-efficiency fuzzy data envelopment analysis technique for performance evaluation of decision making units under uncertainty. Computers & Industrial Engineering **79**, 103 – 114 (2015)

[22] Fanti, M.P., Iacobellis, G., Ukovich, W.: A Metamodelling Approach to Healthcare System management. In: A. Testi, E. Ivaldi, G. Carello, R. Aringhieri, V. Fraghelli (eds.) XXXVI ORHAS conference, Operation Research for Patient-Centered health care delivery, pp. 110–121 (2010)

[23] Fanti, M.P., Mangini, A.M., Dotoli, M., Ukovich, W.: A three-level strategy for the design and performance evaluation of hospital departments. IEEE Trans. On Systems, Man, And Cybernetics: Systems **43**(4), 742–756 (2013)

[24] Field, M.J., Lohr, K.: Guidelines for Clinical Practice: from Development to Use. Washington, DC: National Academy Press (1992)

[25] Fone, D., Hollinghurst, S., Temple, M., Round, A., Lester, N., Weightman, A.: Systematic review of the use and value of computer simulation modelling in population health and health care delivery. Journal of Public Health Medicine **24**(4), 325–335 (2003)

[26] Forrester, J.W.: The impact of feedback control concepts on the Management Sciences. In: Collected Papers of J.W. Forrester, pp. 45–60. Wright-Allen Press (1960)

[27] Fruggiero, F., Lambiase, A., Fallon, D.: Computer simulation and swarm intelligence organisation into an emergency department: a balancing approach across ant colony optimisation. International Journal of Services Operations and Informatics **3**(2), 142–161 (2008)

[28] Gao, S., Krogstie, J.: A Combined Framework for Development of Business Process Support Systems. In: A. Persson, J. Stirna (eds.) The Practice of Enterprise Modeling. Lecture Notes in Business Information Processing, vol. 39, pp. 115–129. Springer, Berlin, Heidelberg (2009)

[29] GreatSPN: `http://www.di.unito.it/~greatspn` (2002). University of Torino

[30] Günal, M.M., Pidd, M.: Discrete event simulation for performance modelling in health care: a review of the literature. Journal of Simulation **4**(1), 42–51 (2010)

[31] Harper, P.R.: A framework for operational modelling of hospital resources. Health Care Management Science **5**(3), 165–173 (2002)

[32] IBM: ILOG CPLEX Optimizer (2011)

[33] Jun, J.B., Jacobson, S.H., Swisher, J.R.: Applications of discrete event simulation in health care clinics: a survey. Journal of the Operational Research Society **50**(2), 109–123 (1999)

[34] Kosanke, K., Vernadat, F., Zelm, M.: CIMOSA process model for enterprise modelling. In: J. Goossenaerts, F. Kimura, H. Wortmann (eds.) Information Infrastructure Systems for Manufacturing. IFIP The International Federation for Information Processing, pp. 59–68. Springer, Boston, MA (1997)

[35] Lagarde, F., Espinoza, H., Terrier, F., Gérard, S.: Improving UML profile design practices by leveraging conceptual domain models. In: R.E.K. Stirewalt, A. Egyed, B. Fischer (eds.) 22nd IEEE/ACM International Conference on Automated Software Engineering (ASE 2007), Atlanta (USA), pp. 445–448. ACM (2007)

[36] Lehaney, B., Hlupic, V.: Simulation modelling for resource allocation and planning in the health sector. Journal of Royal Society Health **115**(6), 382–385 (2008)

[37] Mahulea, C., Mahulea, L., Garcia-Soriano, J., Colom, J.: Modular Petri Net Modeling of Healthcare Systems. Flexible Services and Manufacturing Journal **30**(1-2), 329–357 (2018)

[38] Martin, J., Odell, J.: Object-Oriented Methods: a Foundation. Prentice Hall (1997). 2nd Edition

[39] Mens, T., Gorp, P.V.: A taxonomy of model transformation. Electronic Notes in Theoretical Computer Science **152**, 125 – 142 (2006)

[40] OMG: Business Process Model and Notation. Object Management Group (2011). Version 2.0, formal/11-01-03

[41] OMG: A UML profile for Modeling and Analysis of Real Time Embedded Systems (MARTE). Object Management Group (2011). Document formal/11-06-02

[42] OMG: Unified Modelling Language: Superstructure. Object Management Group (2015). Version 2.5, formal/15-03-01

[43] Parrilla, L., García, J., Albareda, J., Mahulea, C.: HEAT: A Tool to Develop, Analyze and Monitor Clinical Pathways. In: ICNSC'2017: 14th IEEE International Conference on Networking, Sensing and Control. Calabria, Italy (2017)

[44] Recalde, L., Teruel, E., Silva, M.: Structure theory of multi-level deterministically synchronized sequential processes. Theoretical Computer Science **254**(1), 1 – 33 (2001)

[45] Rohleder, T.R., Lewkonia, P., Bischak, D.P., Duffy, P., Hendijani, R.: Using simulation modeling to improve patient flow at an outpatient orthopedic clinic. Health Care Management Science **14**(2), 135–145 (2011)

[46] Roux, O., Combes, C., Duvivier, D.: A modeling methodology dedicated to simulation and based on generic meta-models using MDA-UML: an application to a chronic renal dialysis unit. In: Int. Conf. on Service Systems and Service Management, vol. 1, pp. 692–697 (2006)

[47] Selic, B.: A Systematic Approach to Domain-Specific Language Design Using UML. In: Tenth IEEE International Symposium on Object-Oriented Real-Time Distributed Computing (ISORC2007), 7-9 May 2007, Santorini Island, Greece, pp. 2–9. IEEE Computer Society (2007)

[48] Silingas, D., Vitiutinas, R., Armonas, A., Nemuraite, L.: Domain-Specific Modeling Environment Based on UML Profiles. In: Proc. 15th International Conference on Information and Software Technologies (IT 2009), pp. 167–177 (2009)

[49] Swisher, J.R., Jacobson, S.H., Jun, J.B., Balci, O.: Modeling and analyzing a physician clinic environment using discrete-event (visual) simulation. Computers & Operations Research **28**(2), 105 – 125 (2001)

[50] Tricas, F., García-Valles, F., Colom, J., Ezpeleta, J.: A Petri net Structure-Based Deadlock Prevention Solution for Sequential Resource Allocation Systems. In: Int. Conf. on Robotics and Automation. Barcelona, Spain (2005)

[51] Wang, T., Guinet, A., Belaidi, A., Besombes, B.: Modelling and simulation of emergency services with ARIS and Arena. Case study: the emergency department of Saint Joseph and Saint Luc Hospital. Production Planning & Control **20**(6), 484–495 (2009)

[52] Wells, L.: Performance analysis using CPN tools. In: L. Lenzini, R.L. Cruz (eds.) Proceedings of the 1st International Conference on Performance Evaluation Methodolgies and Tools, VALUETOOLS 2006, Pisa, Italy, October 11-13, 2006, *ACM International Conference Proceeding Series*, vol. 180, p. 59 (2006)

[53] Whittaker, S.J., Rudie, K., McLellan, J.: An Augmented Petri Net Model for Health-Care Protocols. IEEE Transactions on Automatic Control **60**(9) (2015)

[54] Zimmermann, A.: Modeling and Evaluation of Stochastic Petri Nets With TimeNET 4.1. In: VALUETOOLS12: 6th Int. Conf. on Performance Evaluation Methodologies and Tools, pp. 1–10. Cargese, France (2012)

8 Appendix

The appendix provides a background on Unified Modeling Language (UML), Stochastic Well-formed Nets (SWN) and bounding techniques for Timed and Stochastic Petri nets for readers that are not familiar with them.

8.1 Unified Modeling Language

The Unified Modeling Language (UML) [42] is a general purpose, OMG standard, modeling language for system specification. The semantics of UML diagrams is expressed in natural language, while their abstract syntax is provided in terms of UML meta-models[2], which are Class Diagrams (CD) that define the modeling constructs.

The CD are used in this paper both at meta-modeling and modeling levels. In particular, at meta-modeling level, we used them to define: (a) the domain model (Fig. 2 - subsection 4.1), (b) the UML extensions of the HSS profile (Fig. 3 - subsection 4.2). At modeling-level, we used the CD for specifying the hospital resources and patients' health information (Fig. 6 - section 6.1).

The main model elements of a CD are classes and associations. A class is the basic unit that encapsulates all the information of an object (an object is an instance of a class). Through classes we can model the environment under study (a resource, a care step, etc.). A class is graphically represented by a rectangle containing different compartments stacked vertically: the top compartment shows the class's name (e.g., *Healthcare analysis context* in Fig. 2) and the bottom compartment lists the class's attributes (e.g., *params*)[3]. Two classes can be related to each other with association, aggregation, generalization or extension relationships.

In UML associations are, by default, bidirectional and they are graphically depicted as a line between the two classes. Cardinalities can be assigned to both the association-ends. The cardinality of an association-end $A \rightarrow B$, indicates the number of instances of class B associated with an instance of class A and the following syntax is used: "$1..*$" for one or many and "$*$" for zero or many. If no cardinality is shown then it is assumed to be one. Apart from the cardinality, the association-ends can be also characterized by role names (e.g., *release* role assigned to the association-end *ResourceUsageStep→Resource*) and other properties such as constraints stated between curly brackets (e.g., *ordered*).

For example, in Fig. 2, each of two associations between *ResourceType* and *ResourceUsageStep* specifies that one *resource type* instance can have associated zero or more *resource usage step* instances and one *resource usage step* instance can have zero or more *resource type* instances. Two associations are used, since the role played by the *resource* instances in the two associations is different: for one association, the resource instances play the role of *release* and for the other, the resource instances play the role of *acquire*.

The aggregation (represented as a hollow diamond shape) is a particular kind of association that is used to specify that a class instance is a collection of instances of

---

[2] A meta-model is a model of a modeling language.

[3] Class's operations are not considered in this paper.

another class. For example, the aggregation between *Clinical pathway* and *Care Step* specifies that a *clinical pathway* instance is a collection of one or more *care step* instances.

The generalization is a binary taxonomic (i.e., related to classification) directed relationship between a more general class (super-class) and a more specific class (subclass). For example, *ResourceUsageStep* is a specific *Care step*. The generalization is graphically represented as hollow triangle shape on the superclass end of the line.

Finally, the extension relationship is used -at meta-modelling level- to indicate that the properties of a meta-class are extended through a stereotype. For example, in Fig. 3 the stereotype *Care step* extends the metaclass *ControlFlow*.

In this paper, we proposed a customization of UML for the healthcare domain through *profiling*. The UML profiling is a *lightweight* meta-modeling technique to extend UML, since the standard semantics of UML model elements can be refined in a strictly additive manner. Stereotypes and tags are the main extension mechanisms used to define a UML profile. In particular, a stereotype extends one or more UML meta-classes and can be applied to those UML model elements that are instantiations of the extended meta-classes. For example, in Fig. 3, the *ResourceType* stereotype extends the *Class* meta-class, then the former can be applied to a class in a CD (Fig. 6). Just like classes, a stereotype can have properties which are referred as tags: in the previous example *nResources* is a tag of type *NFP_Integer*. When a stereotype is applied to a model element, the value assigned to a stereotype property is called *tagged-value*.

The *UML Profile for Modeling and Analysis of Real-Time and Embedded systems (MARTE)* [41] is an OMG-standard profile that customizes UML for the modeling and analysis of non-functional properties (NFP) of real-time embedded systems, such as timing (e.g., durations) or performance-related properties (e.g., response time, utilization). In particular, MARTE provides a library of predefined NFP data-types (e.g., NFP_Duration) to be associated to tags representing NFP properties (e.g., duration, response time) and the Value Specification Language (VSL). An NFP data-type is characterized by several features, such as the origin which enables the modeler to specify whether an NFP is a requirement or a metric to be estimated and the type of statistical measure associated to an NFP (e.g., mean, min, max). At model specification level, NFP properties are expressed using the VSL well-defined syntax as tagged-values. The annotations of Fig. 7 provides several examples of tagged-values specification using VSL. For example, the *duration* tagged-value (NFP_Duration type) associated to the *ClinicalGuideline* (top-left annotation) indicates that it is a mean (*statQ=mean*) value parameter (*value*=CT) to be estimated (*source=calc*).

8.2 Stochastic Well-formed Net

A Stochastic Well-formed Net (SWN) [17] is a high level Petri net $\mathcal{N} = \langle P, T, C, \mathcal{D}, W^-, W^+, W^h, \Phi, \Pi, \Omega, M_0 \rangle$, where P is the set of places, T is the set of transitions, $C = \{C_1, \ldots, C_n\}$ is the set of basic color classes. Basic color classes are finite and disjoint sets, and each class C_i can be partitioned into several static (disjoint) subclasses $C_i = C_i^1 \cup \ldots \cup C_i^{K_i}$ when it is necessary to make a distinction among groups of colors of the class. Place color domains and variable names of the arc expressions are written in italic fonts in the nets of Fig. 4, 5 and 8.

$\mathcal{D}$ is a function that associates a color domain to each place and transition of the net. Color domains are expressed as Cartesian product of basic color classes (repetition of the same class is allowed): tokens in a place $p \in P$ incorporate information and they can be seen as instances of a data structure whose type is the color domain of p.

SWN transitions can be considered as procedures with formal parameters, where the latter range in the *transition color domain*: the classes in the color domain define the types associated with the transition parameters. The color domain of $t \in T$ is implicitly defined by the color domains of its input, output and inhibitor places, and the relation between transition and place color domains is defined through the input, output and inhibitor arc functions W^-, W^+, W^h. A transition t whose formal parameters have been instantiated to actual values is called *transition instance*, denoted as $[t, c]$, where the assignment c is a color *tuple* belonging to the transition color domain of t. Only transition instances can fire and their enabling and firing depend on the expression of the arcs connected to the transitions.

An arc expression is a sum of weighted tuples of elementary functions defined on the basic color classes. The simplest elementary function is the *projection* one -used in the nets of Fig. 4, 5 and 8- that selects one element from the tuple of value assignments defining the transition instance. The variables used for specifying the function can be chosen arbitrarily, e.g., x, y. When the same variable appears in many arc expressions related to the same transition, the different occurrences actually denote the same object. On the other hand, if the same variable is used in several arc expressions, each related to different transitions, there is no relation between the objects represented by the different variable occurrences.

Φ is a function that associates to each transition $t \in T$ a guard expression: guards are used to restrict the set of admissible color instances of a transition to those satisfying a given predicate. A predicate is expressed in terms of *standard predicates* and it is a boolean expression. By default, $\Phi(t) = true$ is assumed.

Π is the priority function that assigns a priority level to each transition. *Timed* transitions are graphically represented by white tick boxes, such as transition *task A* of Fig. 4 -rule *R2*- and they are characterized by zero priority. Priority levels greater than zero are reserved, instead, for immediate transitions, graphically represented as black thin boxes, such as transitions *yes* and *no* of Fig. 4 -rule *R3*.

Ω is a function that associates to each timed transition a (mean) firing rate, that is the parameter of the negative exponential probability distribution function characterizing the random firing delay of the transition, and to each immediate transition a weight. Transition weights are used for the probabilistic resolution of conflicts among immediate transitions with the same priority.

Finally M_0 is the initial marking function that assigns to each place either a multiset over its color domain or a parameter. In Fig. 4 -rule *R1*- an initial marking parameter nP is assigned to place *pathway start*. The parameter nP is set to the symbolic marking value $\langle S \rangle$, that corresponds to the formal sum $\langle p_1 \rangle + \ldots + \langle p_N \rangle$. The place *pathway start* initially contains N tokens, one per color in the color domain *P*.

8.3 Timed and Stochastic Petri Net bounding techniques

Techniques for performance bound computation have been defined for different types of Petri Nets, in particular for Timed [16] and Stochastic Petri Nets [15]. Later, the bounding techniques have been also extended to Interval-based Time Petri Net [7, 8]. In particular, in [8] a min-max problems is formulated, that generalizes the max-LP problem stated in [15]. In the following, we consider the Petri Net notation and the max-LP problem introduced in [15], which has been applied for the analysis of the case study. The analysis has been carried out using the tool implemented in [8] based on the CPLEX engine [32].

Notation. Let $\mathcal{N} = \langle P, T, Pre, Post, M_0 \rangle$ be a Petri Net system, where P is the set of places, T is the set of transitions, Pre and $Post$ are the pre and post incidence functions, and M_0 is the initial marking function. Let us denote by $\mathbf{Pre}$, $\mathbf{Post}$, and $\mathbf{C} = \mathbf{Post} - \mathbf{Pre}$ the $|P| \times |T|$ matrices representing the Pre, $Post$ and global incidence functions, and by $\mathbf{M_0}$ the row vector representing the initial marking function. Vectors $\mathbf{Y} \geq \mathbf{0}, \mathbf{Y^T} \cdot \mathbf{C} = \mathbf{0}$, where $\mathbf{0}$ is the null column vector, represent P-semiflows, whereas vectors $\mathbf{X} \geq \mathbf{0}, \mathbf{C} \cdot \mathbf{X} = \mathbf{0}$ represent T-semiflows.

Let $\mathcal{T} = (\mathcal{N}, S, R)$ be a Time (Stochastic) Petri Nets, where $\mathcal{N}$ is a Petri Net system, S is the function that assigns the mean service times to transitions (i.e., $S(t_k) = s_k$), and R is the function that assigns relative routing rates at conflicts. Let us denote by $\mathbf{R}$ the matrix of relative routing rates.

For live and structurally bounded Freely Related T-semiflow (FRT) nets [15], the column vector of the visit ratios $\mathbf{v^{(init)}}$, normalized for transition $t_{init} \in T$, can be obtained from the incidence matrix $\mathbf{C}$ and the matrix $\mathbf{R}$ by solving the linear system of equations: $\mathbf{C} \cdot \mathbf{v^{(init)}} = \mathbf{0},\ \mathbf{R} \cdot \mathbf{v^{(init)}} = \mathbf{0}, v_{init}^{(init)} = 1$.

LP problem statement. Let $\mathcal{T} = (\mathcal{N}, S, R)$ be a live and structurally bounded FRT net where all the timed transitions are persistent, that is, once enabled they eventually fire. A cycle time lower bound of transition $t_{init} \in T$ (i.e., the inverse of its throughput upper bound) can be computed by solving the following linear programming problem:

$$\begin{aligned} CT^{init} \quad = \quad & max\ \mathbf{Y^T} \cdot \mathbf{Pre} \cdot \mathbf{D^{(init)}} \\ s.t. \quad & \mathbf{Y^T} \cdot \mathbf{C} = \mathbf{0} \\ & \mathbf{Y^T} \cdot \mathbf{M_0} = 1 \\ & \mathbf{Y} \geq \mathbf{0_P} \end{aligned} \tag{1}$$

where $\mathbf{D^{(init)}}$ is the column vector of the service demands, i.e., $D_k^{(init)} = v_k^{(init)} s_k$.