Analysis and optimization of clinical pathways using timed continuous Petri nets.

César Arzola
Dept. of Computer Science and Systems Engineering
Universidad de Zaragoza, Spain
carzola@unizar.es

Cristian Mahulea, Jorge Júlvez
Aragón Institute for Engineering Research
Universidad de Zaragoza, Spain
{cmahulea,julvez}@unizar.es

***Abstract*—This paper presents a novel approach to analyze and optimize healthcare systems based on clinical pathways, using timed continuous Petri nets (TCPNs) under infinite server semantics. TCPNs allow for an efficient continuous-time analysis of the patient flow and resource utilization dynamics of these types of systems. We demonstrate the feasibility and effectiveness of our method through a case study of a hip fracture pathway at the "Lozano Blesa" University Clinical Hospital, in Zaragoza Spain. Additionally, we propose a method to optimize the behavior of the modeled system by taking a control theory approach, which enables the system to achieve maximum throughput more efficiently. Finally, we provide simulation results to demonstrate the effectiveness of the proposed controller in practical settings.**

***Index Terms*—healthcare systems; clinical pathways; modeling; timed continuous Petri nets; nonlinear analysis and design.**

I. Introduction

Healthcare systems are an essential component of any society, providing crucial medical services to individuals in need. However, the complex and dynamic nature of these systems, including tasks that often demand the coordination of multiple processes and resources, simultaneous actions, and decision-making, can pose significant challenges to their efficient operation. Consequently, the development of advanced analysis and optimization tools for healthcare systems has become a critical research area.

This contribution focuses on the analysis and optimization of healthcare systems based on *clinical pathways* [1]. A clinical pathway is a structured approach to the treatment and management of patients with a particular diagnosis or condition. It provides a standardized set of recommendations for medical staff to follow in terms of diagnostic tests, interventions, and treatments for each step of a patient's care journey. As a case study, we consider the analysis of the clinical pathway of hip fracture from the "Lozano Blesa" University Clinical Hospital in Zaragoza, Spain.

To deal with this problem, discrete event systems (DES) and, in particular, Petri nets (PNs) have proven to be highly appropriate for modeling the event-driven nature of healthcare systems [2]–[5]. The advantage of this approach is that by leveraging these formalisms, healthcare professionals can gain valuable insights into the operation of these complex systems, leading to better patient outcomes and a higher quality of care. However, it is well-known that the use of analysis techniques for DES may become inefficient in highly populated systems since they suffer from the *state explosion problem*, requiring a substantial amount of time to complete the analysis.

In order to cope with this, we propose a model to describe the structure and dynamics of the clinical pathway using *timed continuous Petri nets* (TCPNs), which are a *fluid relaxation* of discrete PNs [6], [7]. The proposed model is based on the one presented in [3] where Stochastic Well-Formed (SWN) Petri nets were used to model the clinical pathway and available SWN solution techniques, such as event-driven simulation [8], were used to obtain an estimation of different performance indices. The advantage of our approach is that it achieves a reduction in computational complexity for analyzing the system while maintaining a good level of accuracy with respect to the behavior of the discrete model. Furthermore, this approach offers the benefit of leveraging the analysis and design techniques that have been developed for TCPNs which have been extensively addressed in the literature (steady-state throughput analysis [9], fault diagnosis [10], controllability [11], observability [12], among others).

TCPNs have been previously used in the literature to model healthcare systems [13], [14]. However, these works deal with the optimization of the system differently. For instance, [14] proposes to optimize the healthcare system by solving a linear programming problem for the determination of the optimal resources, i.e., by redesigning the system by changing its initial parameters and resource availability. In this work, we use a different approach: our goal is to manage the available resources to optimize system performance, assuming redesign is not desirable or feasible.

In order to do that, we take a control theory approach by using the available literature on *controllability analysis* and *control design* for TCPN systems (*e.g.*, [11], [15], [16]). First, we deal with the controllability analysis of the proposed model by using the results developed in [11] that efficiently deal with the controllability verification. This is a crucial property of any dynamic system since if a system is not controllable, then there does not exist a controller that drives the system to the desired state. We have implemented this approach in SimHPN, an available MATLAB toolbox for the analysis of hybrid Petri

*This work was supported by the Spanish Ministry of Science and Innovation through the projects DAMOCLES-PID2020-113969RB-I00 and TED2021-130449B-I00. César Arzola was supported by Conacyt, grant No. 710039.

nets [17]. Next, we propose a control scheme, based on the one proposed in [16] that enables the modeled system to reach a state of maximum throughput more quickly. The scheme is based on an *On-Off* type control over the *firing speed* at the *controllable transitions* that reduces the *marking error*, i.e., the difference between the desired and actual state of the clinical pathway. In the context of the modeled system, the proposed control law can be interpreted as that it prioritizes activities that reduce the marking error, resulting in an optimized use of personnel and a more efficient system. The proposed control law can be computed easily online, despite the complexity of the system. Finally, we provide simulation results that demonstrate the effectiveness of the proposed control scheme in the model under study.

The paper is structured as follows: Section II provides the basic concepts of the formalism and tools utilized for the analysis of the modeling of the healthcare system. In Section III, we briefly review controllability concepts and control laws for TCPNs based on prior research. Section IV outlines the TCPN model developed to represent the clinical pathway, which is validated against the original SWN model. In Section V, we perform a controllability analysis and present the simulation results of the control scheme implementation. Finally, Section VI provides a summary of the conclusions drawn from the study.

II. Basic Concepts

This section introduces some basic concepts of PNs and TCPNs. We assume that the reader is familiar with these subjects, which can be consulted in [6], [7]. Notation $\mathbf{a}[i]$ stands for the i-th entry of vector $\mathbf{a}$. Similarly, $\mathbf{A}[i,j]$ denotes the i,j-th entry of matrix $\mathbf{A}$, $\mathbf{A}[i,\bullet]$ its i-th row, and $\mathbf{A}[\bullet,j]$ its j-th column.

Definition 1: A *Petri net* (PN) structure is a directed bipartite graph defined as $\mathcal{N} = \langle P, T, \mathbf{Pre}, \mathbf{Post}\rangle$ where: P and T are finite non-empty disjoint sets of nodes named places and transitions, respectively; $\mathbf{Pre}$ and $\mathbf{Post}$ are $|P|\times|T|$ matrices, where $\mathbf{Pre}[j,k], \mathbf{Post}[j,k] \in \mathbb{N}_{\geq 0}$ represent the weight of the *arc*, (p_j, t_k), connecting p_j to transition t_k, and the *arc*, (t_k, p_j), connecting t_k to p_j, respectively.
In net systems, each place $p_j \in P$ has a non-negative real number of tokens denoted by $m_j \in \mathbb{R}_{\geq \mathbf{0}}$. The vector $\mathbf{m} \in \mathbb{R}^{|P|}_{\geq \mathbf{0}}$ is named the *marking* (or *state*) of the system, where $\mathbf{m}[j] = m_j$.

Definition 2: A *continuous PN* (CPN) system, $\langle \mathcal{N}, \mathbf{m}_0\rangle$, is a PN structure provided with an *initial marking*, $\mathbf{m}_0 \in \mathbb{R}^n_{\geq \mathbf{0}}$. CPNs allow for transitions to be *fired* in non-negative real-valued amounts. The *enabling degree* of the system at marking $\mathbf{m}_r$ is denoted by the $|T|$-sized vector $\mathbf{enab}(\mathbf{m}_r)$, and is defined s.t. its k-th component is: $\mathbf{enab}(\mathbf{m}_r)[k] = \min_{p_j \in {}^\bullet t_k} \{\mathbf{m}_r[j]/\mathbf{Pre}[j,k]\}$ representing how much t_k can be fired at $\mathbf{m}_r$; t_k is *enabled* at $\mathbf{m}_r$ if $\mathbf{enab}(\mathbf{m}_r)[k] > 0$. An enabled transition t_k can be fired in any real amount, $0 \leq \alpha_k \leq \mathbf{enab}(\mathbf{m}_r)[k]$, leading to a new marking $\mathbf{m}_{r+1} = \mathbf{m}_r + \mathbf{C}[\bullet, k]\alpha_k$, where $\mathbf{C} = \mathbf{Post} - \mathbf{Pre}$ is the the *token-flow* matrix of a PN. This is denoted as $\mathbf{m}_r \xrightarrow{\alpha_k} \mathbf{m}_{r+1}$. A *sequence*, $\sigma = \alpha_k \alpha_l ... \alpha_m$ such that $\mathbf{m}_0 \xrightarrow{\alpha_k} \mathbf{m}_1 \xrightarrow{\alpha_l}, ..., \xrightarrow{\alpha_m} \mathbf{m}_r$ is said to be a *fireable sequence*. A marking $\mathbf{m}$ reached after the firing of σ at $\mathbf{m}_0$ can be computed by the fundamental CPN equation:

$$\mathbf{m} = \mathbf{m}_0 + \mathbf{C}\boldsymbol{\sigma} \tag{1}$$

where the *firing count vector*, $\boldsymbol{\sigma} = [\alpha_1 \ ... \ \alpha_{|T|}]^T$, indicates the amount of firing of each transition in the sequence σ. Given $\langle \mathcal{N}, \mathbf{m}_0\rangle$, the set of all reachable markings from $\mathbf{m}_0$ is convex and it is denoted as $RS(\mathcal{N}, \mathbf{m}_0)$ [7].
If $\mathbf{x} \neq \mathbf{0}$ (resp. $\mathbf{y} \neq \mathbf{0}$) is a non-negative solution of $\mathbf{Cx} = \mathbf{0}$ (resp. $\mathbf{y}^T\mathbf{C} = \mathbf{0}$) then it is named *T-semiflow* (resp. *P-semiflow*). $\mathcal{N}$ is *consistent*, denoted as Ct (resp. *conservative*, denoted as Cv) if there exists a T-semiflow $\mathbf{x} > \mathbf{0}$ (resp. P-semiflow $\mathbf{y} > \mathbf{0}$).

The notion of time can be included in CPNs by timing the firing of their transitions, leading to the *timed continuous Petri net* systems.

Definition 3: A *timed continuous Petri net* (TCPN) system is a continuous-state system described by the tuple $\langle \mathcal{N}, \boldsymbol{\lambda}, \mathbf{m}_0\rangle$, where $\langle \mathcal{N}, \mathbf{m}_0\rangle$ is a CPN system and $\boldsymbol{\lambda} \in \mathbb{R}^{|T|}_{>0}$ is the *firing rate vector* or *timing* of the net, which assigns to each transition a positive real number representing its *firing rate*. By assuming that the marking and the firing count vector in (1) are dependent on time, τ, $\mathbf{m}(\tau) = \mathbf{m}_0 + \mathbf{C}\boldsymbol{\sigma}(\tau)$, the state equation of the timed system is obtained by taking the time derivative of $\mathbf{m}(\tau)$:

$$\dot{\mathbf{m}}(\tau) = \mathbf{C}\mathbf{f}(\tau) \tag{2}$$

where $\mathbf{f}(\tau) = \dot{\boldsymbol{\sigma}}(\tau)$ is the *firing flow* vector of the TCPN.
This work defines the flow vector using *infinite server semantics* (ISS), which offers a more accurate approximation of the performance of the system's discrete counterpart for a broad class of net systems [18]. Under ISS, the flow of the k-th transition is defined as the product of its firing rate, $\lambda_k = \boldsymbol{\lambda}[k]$, and its instantaneous enabling degree, $\mathbf{enab}(\mathbf{m}(\tau))[k]$:

$$\mathbf{f}(\tau)[k] = \lambda_k \mathbf{enab}(\mathbf{m}(\tau))[k] = \lambda_k \min_{p_j \in {}^\bullet t_k} \left\{ \frac{\mathbf{m}(\tau)[j]}{\mathbf{Pre}[j,k]} \right\} \tag{3}$$

The min operator in Eq. (3) allows describing the state evolution of a TCPN system as a *piecewise affine* system. The next concepts illustrate this:
(1) A *configuration* of the TCPN, $\mathcal{C} = \{(p_{\alpha_1}, t_1), \cdots, (p_{\alpha_{|T|}}, t_{|T|})\}$, is a set of arcs, one per transition s.t. $p_{\alpha_k} \in {}^\bullet t_k$.
(2) The $|T| \times |P|$ *configuration matrix* $\mathbf{\Pi}_i$, associated to $\mathcal{C}_i$, is defined as:

$$\mathbf{\Pi}_i[k,j] = \begin{cases} \frac{1}{\mathbf{Pre}[j,k]} & \text{if } (p_j, t_k) \in \mathcal{C}_i \\ 0 & \text{otherwise} \end{cases} \tag{4}$$

(3) A configuration $\mathcal{C}_i$ is said to be *active at* $\mathbf{m}(\tau)$ if $\mathbf{\Pi}_i \mathbf{m}(\tau) = \mathbf{enab}(\mathbf{m}(\tau))$. If $\mathcal{C}_i$ is active and $(p_j, t_k) \in \mathcal{C}_i$, it is said that p_j *constrains* the flow of t_k at $\mathcal{C}_i$.
(4) A *region* $\mathcal{R}_i$ is the convex subset of markings for which $\mathcal{C}_i$ is active: $\mathcal{R}_i = \{\mathbf{m} \in RS(\mathcal{N}, \mathbf{m}_0) | \mathbf{\Pi}_i \mathbf{m} = \mathbf{enab}(\mathbf{m})\}$. According to (3), within region $\mathcal{R}_i$, the flow vector is given by $\mathbf{f}(\tau) = \mathbf{\Lambda} \cdot \mathbf{enab}(\mathbf{m}(\tau)) = \mathbf{\Lambda}\mathbf{\Pi}_i \mathbf{m}(\tau)$ where $\mathbf{\Lambda}$ is the diagonal

matrix containing the entries of $\boldsymbol{\lambda}$ in its main diagonal.

(5) A *linear mode* of a TCPN system, Σ_i, is the linear-time invariant system, $\dot{\mathbf{m}}(\tau) = \mathbf{C}\mathbf{f}(\tau) = \mathbf{C}\boldsymbol{\Lambda}\boldsymbol{\Pi}_i\mathbf{m}(\tau)$, that describes the evolution of the marking inside $\mathcal{R}_i$[1].

Given a TCPN system, a marking $\mathbf{m}_{ss} \in RS(\mathcal{N}, \mathbf{m}_0)$ that fulfills that $\mathbf{m}_{ss} = \lim_{\tau\to\infty} \mathbf{m}(\tau)$, $\mathbf{m}_{ss} \in \mathcal{R}_s$ and $\dot{\mathbf{m}} = \mathbf{C}\boldsymbol{\Lambda}\boldsymbol{\Pi}_s\mathbf{m}_{ss} = \mathbf{0}$ is named a *steady state* of the system. The *steady-state flow*, or *throughput*, of the system is given by $\mathbf{f}_{ss} = \boldsymbol{\Lambda}\boldsymbol{\Pi}_s\mathbf{m}_{ss}$.

Control actions can be applied to TCPN systems to enforce a desired behavior. They consist of *local reductions of the flow through the transitions*. Transitions in which control actions can be applied are named *controllable*. The set of all controllable transitions is denoted by T_c and the set of *uncontrollable* transitions is $T_{nc} = T \setminus T_c$. The *control vector* $\mathbf{u}(\tau) \in \mathbb{R}^{|T|}_{\geq \mathbf{0}}$ is defined s.t. $\mathbf{u}(\tau)[j]$ represents the control action on the j-th transition. A control vector is called *suitably bounded* (s.b.) if it fulfills that $\mathbf{u}(\tau) \leq \mathbf{f}(\tau)$ and that $\mathbf{u}(\tau)[k] = 0$, $\forall t_k \in T_{nc}$. Only, s.b. control vectors can be applied. The *forced* state equation of a TCPN system is given by:

$$\dot{\mathbf{m}}(\tau) = \mathbf{C}[\mathbf{f}(\tau) - \mathbf{u}(\tau)] = \mathbf{C}\boldsymbol{\Lambda}\boldsymbol{\Pi}(\mathbf{m})\mathbf{m}(\tau) - \mathbf{C}\mathbf{u}(\tau) \quad (5)$$
$$0 \leq \mathbf{u}(\tau) \leq \mathbf{f}(\tau)$$

where $\boldsymbol{\Pi}(\mathbf{m}) = \boldsymbol{\Pi}_i$ when $\mathbf{m}(\tau) \in \mathcal{R}_i$. In the following, the particular case of TCPN system with $\mathbf{u}(\tau) = \mathbf{0}$ will be referred to as the *unforced* case.

An important subset of $RS(\mathcal{N}, \mathbf{m}_0)$ is the *set of equilibrium markings*, *i.e.*, the "potential steady states" of the forced system. The set of equilibrium markings in $\mathcal{R}_i$ is:
$E_i = \{\mathbf{m}_q \in \mathcal{R}_i \mid \exists \mathbf{u}_q \text{ s.b., s.t. } \mathbf{C}(\boldsymbol{\Lambda}\boldsymbol{\Pi}_i\mathbf{m}_q - \mathbf{u}_q) = \mathbf{0}\}$ The union of the sets E_i, for all $\mathcal{R}_i$, is denoted as $\mathbb{E}$.

To simplify the forced state equation, we can interpret the control inputs as local variations in the firing rate of the controllable transitions. This can be achieved by rewriting the input signal as $\mathbf{u}(\tau) = \mathbf{I}_u(\tau)\boldsymbol{\Lambda}\boldsymbol{\Pi}(\mathbf{m})\mathbf{m}(\tau)$ where the control variable is $\mathbf{I}_u(\tau) = \text{diag}[I_{u1}(\tau) \ \dots \ I_{u|T|}(\tau)]$ with $0 \leq I_{uk}(\tau) \leq 1 \ \forall t_k \in T_c$ and $I_{uk}(\tau) = 0 \ \forall t_k \in T_{nc}$. Then the matrix $\mathbf{I}_c(\tau) = \mathbf{I} - \mathbf{I}_u(\tau) = \text{diag}[I_{c1}(\tau) \ \dots \ I_{c|T|}(\tau)]$ is constructed, where $\mathbf{I}$ is the identity matrix of size $|T|$, and the forced state equation can be rewritten as:

$$\dot{\mathbf{m}}(\tau) = \mathbf{C}\mathbf{I}_c(\tau)\boldsymbol{\Lambda}\boldsymbol{\Pi}(\mathbf{m})\mathbf{m}(\tau)$$

Notice that $0 \leq I_{ck}(\tau) \leq 1 \ \forall t_k \in T_c$ and $I_{ck}(\tau) = 1 \ \forall t_k \in T_{nc}$.

III. Controllability and Control in TCPN systems

Controllability analysis of dynamic systems is crucial since if a system is not controllable, then there does not exist a controller that drives the system in the desired way. The analysis of controllability for the particular case of TCPNs has been extensively studied in the literature, obtaining adequate and efficient results to perform this task for this particular class of systems [11], [15]. Our focus in this paper, however, is not to provide an in-depth explanation of the theory, but rather to demonstrate the effectiveness of the control strategy in optimizing the system performance. We will provide a high-level overview of the analysis and control approach used in this work. In the literature, the controllability for TCPN systems under ISS has been defined as:
A TCPN system is *bounded input controllable* (BIC) over a set of markings $S \subseteq RS(\mathcal{N}, \mathbf{m}_0)$ if, for any $\mathbf{m}_1$, $\mathbf{m}_2 \in S$, there exists a s.b. $\mathbf{u}(\tau)$ that transfers the system from $\mathbf{m}_1$ to $\mathbf{m}_2$ in finite or infinite time and maintains this marking.

In this work, we deal with TCPN systems that contain uncontrollable transitions. Consequently, these systems are frequently not BIC over $RS(\mathcal{N}, \mathbf{m}_0)$ [15]. The commonly adopted goal for this case is to study BIC over the equilibrium markings of the system. This is particularly relevant in practical applications, where controllers are designed to guide the system toward specific stationary states. To cope with this, we will employ the techniques developed in [11], which study structural conditions (depending only on the structural information of the TCPN) for *net rank-controllability* (NRC). NRC is a structural sufficient condition for BIC over the equilibrium markings of a system and can be tested efficiently by means of polynomial time algorithms. We have implemented these algorithms in SimHPN, an available MATLAB toolbox for the analysis of hybrid Petri nets [17]. This toolbox will be utilized for the analysis of the controllability of the system under consideration.

After dealing with the controllability analysis, we turn our attention to the following control design problem:
Given a TCPN system, $\langle \mathcal{N}, \boldsymbol{\lambda}, \mathbf{m}_0 \rangle$ and a required target marking, $\mathbf{m}_r$, design a control law that drives the marking of the TCPN from $\mathbf{m}_0$ to $\mathbf{m}_r$, and then keeps the marking at $\mathbf{m}_r$ indefinitely.

To solve this problem from our particular case study, we adopt a control law presented in [16], originally designed for the case where all the transitions are controllable. This scheme is based on defining a required marking, $\mathbf{m}_r \in \mathbb{E}$ and defining the marking error as:

$$\mathbf{e}(\tau) = \mathbf{m}_r - \mathbf{m}(\tau) \quad (6)$$

Based on this, we can now define the *contribution degree* of a controllable transition. This is a measure of how much the firing of a particular transition contributes to reducing the overall error in the system, formally defined as:

The *contribution degree* of the k-th transition, $\Psi_k(\tau)$, is defined as:

$$\Psi_k(\tau) = \mathbf{e}^T(\tau)\mathbf{C}[\bullet, k] \quad (7)$$

Now, we can define the control law as follows:

Definition 4: Let $\langle \mathcal{N}, \boldsymbol{\lambda}, \mathbf{m}_0 \rangle$ be a TCPN system and $\mathbf{m}_r \in \mathbb{E}$ be the required marking. For each $t_k \in T_c$, its control input at instant τ is given by:

$$I_{ck}(\tau) = \begin{cases} 1 & \text{if } \Psi_k(\tau) > 0 \\ 0 & \text{otherwise} \end{cases} \quad (8)$$

[1]When the marking is at the border of two adjacent regions, any of the corresponding linear modes can be used interchangeably to govern the system's dynamics (thus, it is not important which one is taken).

In section V, the analysis and design techniques discussed previously will be applied to the modeled system. It will be demonstrated that the proposed scheme is capable of improving the performance of the system.

IV. Fluid model of a hip fracture clinical pathway

In this section, we present the fluid model for the study of the clinical pathway of hip fracture from the "Lozano Blesa" University Clinic Hospital in Zaragoza, Spain. It is based on the model developed in [3], as a stochastic well-formed Petri net (SWN). We validate our model by comparing the results obtained through TCPN simulation with those of the discrete system. We show how the TCPN system can be used to obtain a more efficient continuous-time analysis of the patient flow and resource utilization dynamics while maintaining a good level of accuracy, w.r.t. the original model.

The proposed methodology serves as a valuable tool for hospital managers, enabling comprehensive resource assessments and informed decision-making during the different parts of the process. It can be used to obtain insights and recommendations to optimize resource allocation and improve overall performance. By utilizing the model, managers can identify resource utilization patterns, address potential bottlenecks, and enhance patient care and operational efficiency proactively.

A. TCPN model of the clinical pathway

The proposed TCPN model is depicted in Fig. 1. It contains some modifications w.r.t. the original SWN, which will be explained in the following. As in the original discrete model, it captures the clinical pathway for hip fracture treatment: it consists of all the tasks that need to be done during the pre-operative day of hospitalization (left side of the figure), the surgery (at the bottom of figure), and the post-operative day (right side of the figure). The TCPN model includes places that represent both the patients (p_1) and the resources of the system, such as nurses (p_{41}) and doctors (p_{42}). The model shows how these entities are utilized throughout the clinical pathway.

Originally, the SWN contains *conflicts* that represent the different alternatives through the clinical pathway. For instance, p_6 models the possibility of the patient having a urinary infection or not. In the SWN (Fig. 8 in [3]), a *choice place* is used to model this scenario and a probabilistic resolution policy is used to solve the conflict. For this particular case, statistically, only 10% of the patients suffer from urinary infection and need an urgent pre-operative study (procedure modeled from p_7 to t_9) while 90% of the cases can continue the pathway without it. For simplicity of analysis, we remove the probabilistic choices in our model by introducing equivalent structures composed by *fork transitions* (t_6, t_{10}, and t_{37}) that divide the token flow proportionally to each branch's probability. For instance, for t_6, for each token that consumes, it allocates 0.9 tokens to the first branch and 0.1 tokens to the second branch, representing the behavior of the original system.

Next, to set the parameters for performance evaluation purposes, we need to determine the timing of the transitions. For the SWN, the mean time delay of each transition is defined according to the time interval it takes to accomplish the task it represents. This information is reported in Table 1 of [3]. The SWN model has two types of transitions: immediate and timed. The former type represents events that occur instantaneously, such as the allocation and release of resources or probabilistic choices (depicted as black transitions in Fig. 1). The latter represents the tasks that must be performed during the clinical path (depicted as white transitions in the figure). This information is used to define the firing rate of the transitions in the TCPN model, summarized in Table I. It is worth noticing that we assume that the timing of the immediate transitions is not instantaneous but only much faster than the rest of the transitions. This way, we can still consider the controllability analysis techniques from the literature [11], which do not consider the case of immediate transitions, while still obtaining a good approximation of the original model.

TABLE I
Firing rates of the transitions of the clinical pathway.

Transitions set	Timing (1/min)
$\{t_4, t_5, t_{12}, t_{15}, t_{16}, t_{20}\}$	1
$\{t_{11}, t_{17}, t_{29}, t_{30}, t_{34}, t_{38}\}$	1/5
$\{t_3, t_{28}\}$	1/15
$\{t_8, t_{25}, t_{31}\}$	1/30
$\{t_{23}\}$	1/120
$\{t_1, t_2, t_6, t_7, t_9, t_{10}, t_{13}, t_{14}, t_{18}, t_{19}, t_{21}, t_{22}$ $t_{24}, t_{26}, t_{27}, t_{32}, t_{33}, t_{35}, t_{36}, t_{37}, t_{39}, t_{40}\}$	10

Finally, the initial marking of the system is determined by the number of available entities: number of patients nP, number of nurses nN, and number of doctors nD. Initially, the marking in the rest of the places is 0, representing that no patient is receiving care. Several cases will be simulated in order to validate the model. For more information on the modeled clinical pathway, please see [3].

B. TCPN model validation

In order to validate the proposed TCPN, we present the performance results of our model under different assumptions of patient workload and resource plan and compare them with the results obtained for the SWN in [3]. The results are summarized in Fig. 2. The performance index of interest is the cycle time of t_1, CT_{t_1}, multiplied by the number of patients, nP, i.e. $CT = CT_{t_1} nP$. This measure corresponds to the mean time spent for a patient to undergo the treatment [3]. The CT values for the discrete case were obtained by event-driven simulation using GreatSPN [8] and the SWN model.

To calculate this performance index in the TCPN system, we first obtain the cycle time of a transition by analyzing its throughput (the flow through its transitions at the steady state) [7]. This was performed by simulating the behavior of the system until it reached a steady state, using SimHPN[2]. The throughput of transition t_1 can then be obtained from the flow at the steady state, $\mathbf{f}_{ss}$, as $\chi_1 = \mathbf{f}_{ss}[t_1]$. Specifically,

[2]The simulations are performed using the algorithms implemented on SimHPN with a computer with i5-6600 CPU @ 3.30GHz and 16GB RAM.

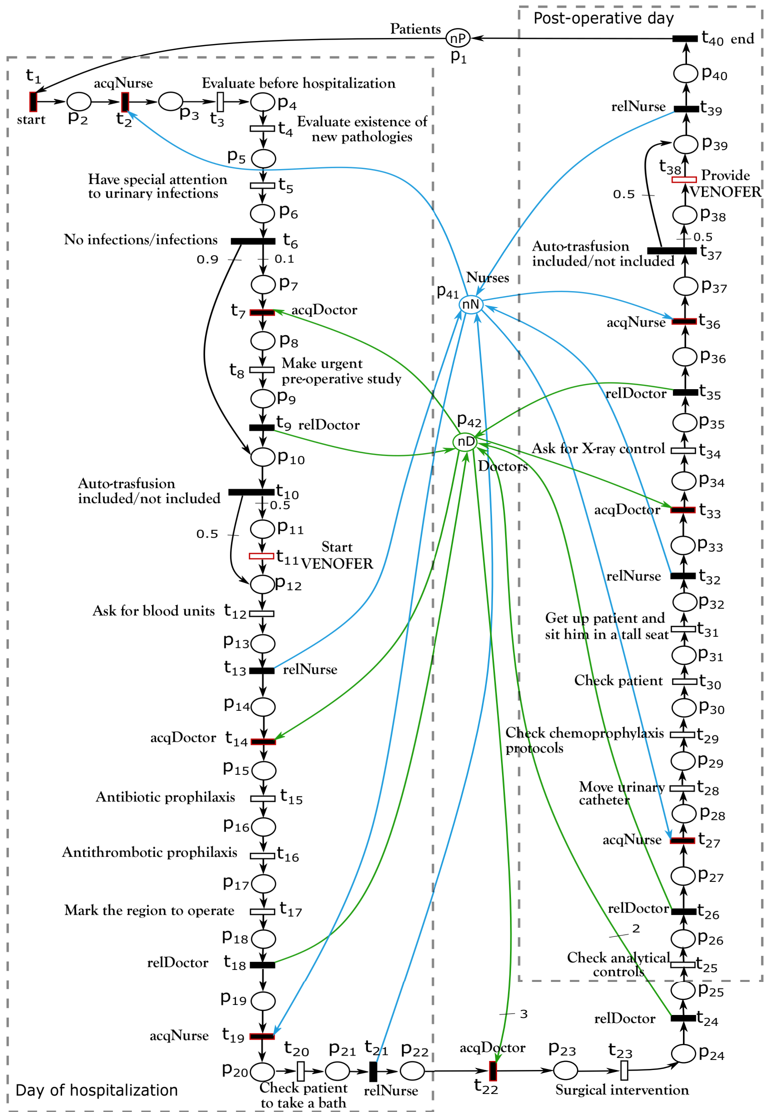

Fig. 1. A TCPN system that models a hip fracture clinical pathway. The controllable transitions of the system are represented with a red outline. The model is contained in SimHPN's models folder as Model_HipFractureClinicalPathway_SS.mat

the throughput of a transition corresponds to the inverse of its cycle time. Hence, the overall performance index can be computed as $CT = \frac{1}{\chi_1} nP \frac{1}{1440}$, where the factor $\frac{1}{1440}$ is added to convert the time units from minutes (used in the TCPN simulation) to days.

The simulation results for both models are reported in Fig. 2. We simulated the 26 different scenarios presented in [3], where the authors considered several patient workloads on the interval $[1, 100]$ and different personnel resource plans. We compared the CT obtained by using our proposed fluid model (5^{th} column) against the one obtained by the event-driven simulation (8^{th} column). The table also shows the relative error between the performance values computed with both techniques (6^{th} column). The error was computed by considering the results of the event-driven simulation as reference values: Rel.Err $= \frac{CT^{SWN} - CT^{TCPN}}{CT^{SWN}} \times 100$). Furthermore, the solution time for obtaining the performance values in each case is reported in columns 7 and 9.

The results in Fig. 2 show that the TCPN system provides a suitable approximation for the problem under consideration. Notably, the error tends to decrease as the number of tokens (resources) increases. This is likely due to the fact that fluidization typically provides a good approximation of the original system in highly marked systems [18], which are often the most challenging cases to analyze for the SWN model. Overall, the relative error between the TCPN approximation and the SWN model is relatively low, with a maximum error of 21.05% occurring in only one scenario (case 5), in which the number of tokens is relatively low. For the rest of the scenarios, the error is below 12.25%, indicating that *the TCPN approximation provides an accurate representation of the system behavior*. Moreover, we observed that the simulation time of the SWN model worsens as the number of resources increases. In contrast, the TCPN simulation can handle larger

	Resources			TCPN Sim			SWN Sim [3]	
Case	nP	nD	nN	CT	Rel.Err.	Sol. Time	CT	Sol. Time
1	100	3	5	9.396	11.53%	303.19	10.62	11.03
2	76	3	5	7.141	11.89%	227.83	8.105	8.02
3	50	3	5	4.698	12.24%	128.77	5.353	6.01
4	26	3	5	2.443	10.97%	61.26	2.744	3.04
5	2	3	5	0.1879	21.05%	9.25	0.238	0.03
6	1	3	5	0.1712	0.71%	6.89	0.17	0.02
7	100	6	5	4.698	7.97%	134.29	5.105	59
8	76	6	5	3.5703	10.43%	99.06	3.986	40.08
9	50	6	5	2.349	9.23%	67.34	2.588	29.06
10	26	6	5	1.2214	10.91%	29.22	1.371	15.01
11	2	6	5	0.1708	0.70%	9.23	0.172	0.05
12	100	9	5	3.1318	6.99%	79.23	3.367	132.08
13	76	9	5	2.3803	8.31%	64.08	2.596	98.07
14	50	9	5	1.566	7.12%	39.78	1.686	65.08
15	26	9	5	0.8143	6.29%	21.53	0.869	36.08
16	2	9	5	0.1708	1.07%	9.45	0.169	3.07
17	100	15	5	1.879	4.62%	63.84	1.97	1123.04
18	76	15	5	1.428	5.49%	37.24	1.511	936.08
19	50	15	5	0.94	5.72%	24.37	0.997	416.01
20	26	15	5	0.488	4.87%	12.20	0.513	200.07
21	2	15	5	0.1707	1.01%	7.61	0.169	2.03
22	100	25	5	1.1503	3.58%	274.14	1.193	1293.06
23	76	25	5	0.8742	1.66%	228.07	0.889	578.08
24	50	25	5	0.575	1.71%	248.67	0.585	707.02
25	26	25	5	0.299	8.00%	202.34	0.325	106.06
26	2	25	5	0.1708	0.47%	9.79	0.17	88.02

Fig. 2. Cycle time (CT) values estimated using TCPN (5^{th} column) and SWN simulation (8^{th} column). The relative error between both techniques is also reported (6^{th} column) and the solution time for each of the different scenarios (7^{th} and 9^{th} columns). CT values are given in days and the solution time in seconds.

systems with less computational time, making it a more scalable approach for complex systems.

V. Control and Optimization of the Fluid Model

The use of the TCPN model offers the advantage of enabling the application of various analysis and design techniques developed for this formalism. In this section, we demonstrate how this feature can be leveraged to improve the performance of the system by analyzing its controllability and designing an appropriate control law. In particular, we show the implementation of a control law that enables the system to reach its state of maximum throughput more quickly. We provide simulation results to demonstrate the effectiveness of our controller in practical settings.

A. Structural controllability analysis using SimHPN

In this section, we analyze the controllability of the fluid model employing the techniques developed in [11], which we have implemented in the toolbox SimHPN. The model and parameters of this system are included in the toolbox (to load the model select *Model* $\rightarrow$ *Import from .mat file* $\rightarrow$ *models\Model_HipFractureClinicalPathway_SS.mat*). The controllability test can be carried out by selecting *Continuous* $\rightarrow$ *Structural controllability analysis* $\rightarrow$ *Net rank-controllability test* (Fig. 3). This test verifies sufficient conditions for net rank-controllability efficiently. The transitions that can be controlled are depicted with a red outline in Fig. 1.

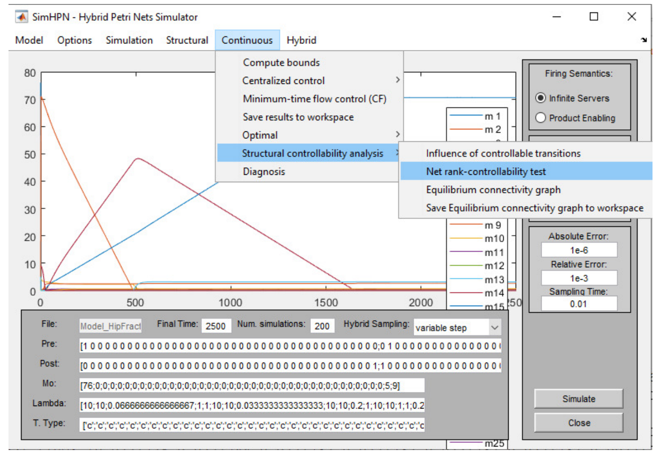

Fig. 3. SimHPN, the toolbox used for the performance and structural controllability analysis of the system.

They represent the events of the patients entering the pathway, assigning nurses and doctors to the different stages of the process and some tasks within the pathway. We present 3 control scenarios:

1) $T_c = \{t_1, t_{11}, t_{38}\}$, i.e., *the events of assigning nurses and doctors cannot be controlled*: For this case, we obtain the message "*Influence is not total! Therefore, the set of controllable transitions does not guarantee net rank-controllability. The only influenced nodes are: Places = [1 2 3 4 5 6 7 10 11 12 13 14 38 39 40 41] Transitions = [1 2 3 4 5 6 10 11 12 13 38 39 40]*". From a system point of view, total influence means that the behavior of all states in the system can be influenced by a set of controllable transitions, which is a necessary condition for NRC [11]. In the context of a clinical pathway, if the property of influence is not fulfilled, it means that there are some modes of operation (configurations) in which the behavior of several parts of the pathway cannot be affected by means any control action.

2) $T_c = \{t_1, t_7, t_{11}, t_{14}, t_{22}, t_{38}\}$, i.e., *the events of assigning nurses cannot be controlled*: By entering this set of controllable transitions we obtain the message "*It is not possible to decide if the timed net is net rank-controllable. The condition related to the choice places is not fulfilled.*". In this case the test cannot conclude if the system is NRC, since one of the structural sufficient conditions does not hold. In particular, the one related to the choice places (see [11] for more details). This serves to give indications to the user about where the problem may be in order to guarantee controllability.

3) $T_c = \{t_1, t_2, t_7, t_{11}, t_{14}, t_{19}, t_{22}, t_{27}, t_{33}, t_{36}, t_{38}\}$, i.e., *the allocation of these resources can be decided upon*: Here we choose the set that we will consider in our case study, which is such that it satisfies all the structural conditions for controllability and is indicated by the message: "*The timed net is net rank-controllable.*"

It is worth noting that, in general, the controllability analysis is crucial to ensure the existence of a control law for a system. In the literature, the synthesis of controllers for TCPNs has often been addressed without tackling the controllability analysis, considering particular cases in which this property trivially holds (usually only when all transitions can be controlled). However, the implementation of the results in [11] allows for an efficient controllability analysis.

B. Control law implementation

In this section, we implement the controller presented in Section III to the TCPN system. The goal is to reach a state that guarantees maximum throughput, i.e., allocate the available resources to ensure the optimal performance of the system. To establish the target marking, we can use SimHPN to compute such a state using: *Continuous* $\rightarrow$ *Optimal* $\rightarrow$ *Optimal Control*. In order to compare the performance of the controlled system against the unforced system ($\mathbf{u}(\tau) = \mathbf{0}$), we choose the state of maximal throughput of the unforced system as our target. To do this, we set the value of Uncontrolled Transitions as = [1:40] (as if there are no controllable transitions), in the optimal control menu and use the default values for the "Gain Vector w.r.t. Flow: w" and "Cost Vector Due to Immobilization to Maintain the Product Flow z" options. Although other target markings could be chosen by applying additional constraints, we select this one to enable a direct comparison. The obtained target marking is $\mathbf{m}_r = [0.0022\ 0.0022\ 0.3326\ 0.0222\ 0.0222\ 0.0022\ 0.0002\ 0.0665\ 0.0002\ 0.0022\ 0.0554\ 0.0222\ 0.0022\ 0.0022\ 0.0222\ 0.0222\ 0.1109\ 0.0022\ 0.0022\ 0.0222\ 0.0022\ 70.5474\ 2.6607\ 0.0022\ 0.6652\ 0.0022\ 0.0022\ 0.3326\ 0.1109\ 0.1109\ 0.6652\ 0.0022\ 0.0022\ 0.1109\ 0.0022\ 0.0022\ 0.0022\ 0.0554\ 0.0022\ 0.0022\ 3.1659\ 0.0067]^T$. Once an optimal steady state has been computed, the objective is to reach and maintain the desired marking in the shortest amount of time possible.

Before implementing the proposed control law (Eq. 8), it is important to notice that due to its nature, it will generate a high-frequency control input [16], switching between two possible states: a transition working at maximum capacity (when the value is 1) and stopping its activity (when the value is 0). Therefore, the control input may lack physical meaning for the discrete system, making its interpretation and implementation difficult. Nevertheless, we propose a solution by implementing the control scheme depicted in Fig. 4 that adapts the high-frequency switching input to a smoother, more physically meaningful signal, ensuring that our approach is not only feasible but also suitable for the intended application.

In particular, we filtered the On-Off control input of each transition by computing the *running mean* of the corresponding control signal and applying a first-order low-pass filter to it (using off-the-shelf functions in MATLAB). The result of this procedure is depicted in Fig. 5 for the control signal of t_7. The upper subplot shows the high-frequency control input computed with Eq. (8). The zooms on the plot show how the duty cycle of the signals changes over time. The lower subplot shows the filtered signal, which captures the control input as a smoother control signal for the controllable transitions of the system.

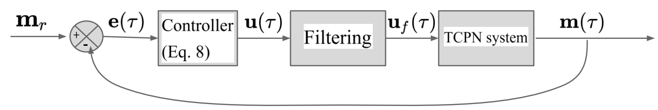

Fig. 4. Control scheme for the TCPN system.

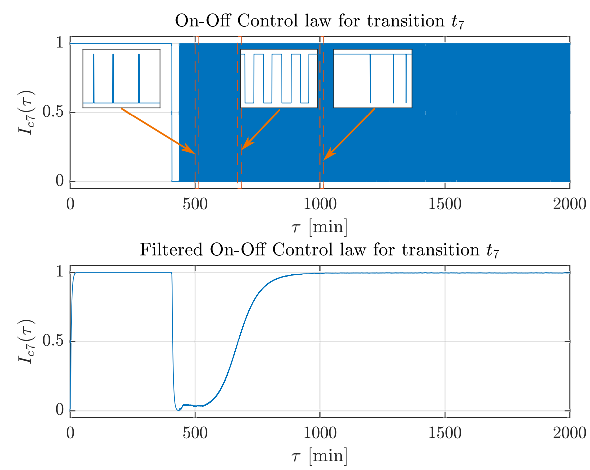

Fig. 5. Filtering control input of transition t_7.

Figure 6 shows the simulation results of the implemented control: The upper subplot depicts the marking error (the difference between the desired and the current state) of the unforced case; the lower subplot depicts the marking error for the forced case. Notice that it takes more than 1700 minutes for the unforced case to reach the target. On the contrary, for the forced case it takes around 700 minutes to reach it. The filtered control signal on the different transitions is depicted in Fig. 7.

To interpret the physical meaning of the simulation results in the context of the clinical pathway, it is important to

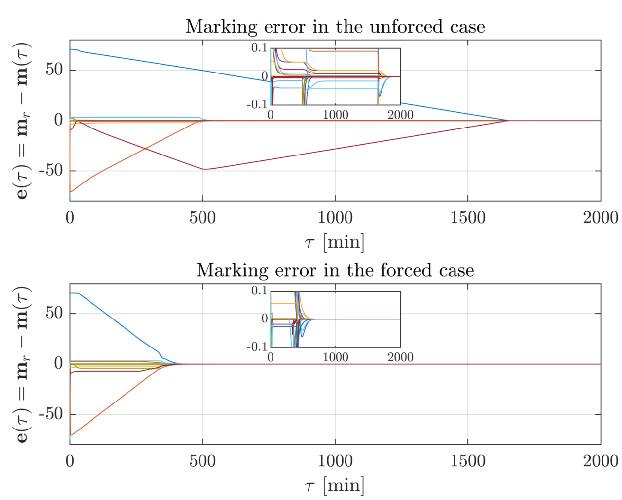

Fig. 6. Marking error for the unforced and the forced case.

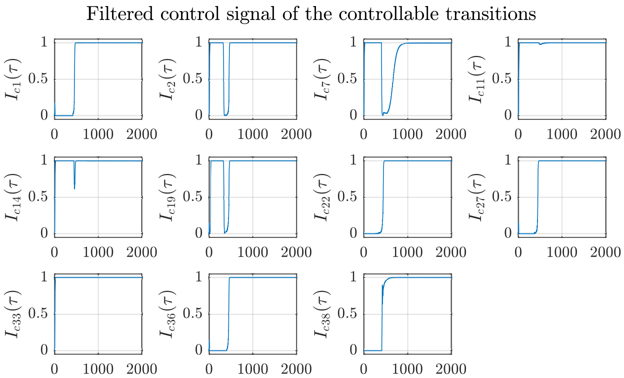

Fig. 7. Filtered control signal of the controllable transitions.

remember that the control inputs in our model correspond to delays in the firing rate of certain transitions. In a clinical pathway model with limited resources, the delays represent the addition of extra time to certain activities to prioritize others, thus optimizing the use of the available resources to ensure that the system achieves its maximum productivity as quickly as possible. For instance, in Fig. 7, the delay added to transition t_1 (which represents the entrance of patients to the pathway), can be seen as the rate at which patients enter the pathway is being regulated to avoid bottlenecks in the early parts of the process and prioritize the efficient use of the available personnel. Another example is the control actions of t_7 and t_{22} which represent the use of doctors in a pre-operative phase and during the surgery, respectively: at the beginning of the process, we may prioritize the use of doctors during the preoperative stage (t_7), to ensure that patients are adequately prepared for surgery. As more patients enter the pathway and become ready for surgery, we can then shift the focus to the surgical stage (t_{22}), where doctors are needed to perform the operations. These are examples of how, by carefully controlling the flow through the controllable transitions, we can ensure that the available resources are utilized efficiently and that the system reaches its steady state as quickly as possible, optimizing the overall throughput of the system.

VI. Conclusions

This paper proposes a novel approach using timed continuous Petri nets (TCPNs) to analyze and optimize healthcare systems through clinical pathways. Our approach is demonstrated through a case study of a hip fracture clinical pathway at the “Lozano Blesa” University Clinical Hospital, in Zaragoza Spain. First, it was shown that the TCPN model can be used to perform a more efficient analysis of the patient flow and resource utilization dynamics of the system while maintaining a good level of accuracy, w.r.t. the discrete techniques. Next, a control scheme that enables the modeled system to reach its state of maximum throughput more quickly, was proposed. The effectiveness of the proposed method is demonstrated using simulation results. Moreover, it is worth noting that the presented method can be generalized to the different types of systems that can be modeled by TCPNs, encompassing a wide range of domains such as traffic systems, flexible manufacturing systems, and more.

Finally, while a formal proof of convergence for the proposed control scheme is not provided, the paper highlights its potential and the need for further investigation to understand its limitations. Future research can also explore the practical implementation of the proposed control scheme into the real discrete system.

References

[1] T. Rotter, L. Kinsman, E. L. James, A. Machotta, H. Gothe, J. Willis, P. Snow, and J. Kugler, “Clinical pathways: effects on professional practice, patient outcomes, length of stay and hospital costs,” *Cochrane database of systematic reviews*, no. 3, 2010.

[2] V. Augusto and X. Xie, “A modeling and simulation framework for health care systems,” *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 44, no. 1, pp. 30–46, 2014.

[3] S. Bernardi, C. Mahulea, and J. Albareda, “Toward a decision support system for the clinical pathways assessment,” *Discrete Event Dynamic Systems*, vol. 29, pp. 91–125, 2019.

[4] C. W. Kang, M. Imran, M. Omair, W. Ahmed, M. Ullah, and B. Sarkar, “Stochastic-Petri net modeling and optimization for outdoor patients in building sustainable healthcare system considering staff absenteeism,” *Mathematics*, vol. 7, no. 6, 2019.

[5] J. Wang, “Patient flow modeling and optimal staffing for emergency departments: A Petri net approach,” *IEEE Transactions on Computational Social Systems*, pp. 1–11, 2022.

[6] R. David and H. Alla, *Discrete, Continuous, and Hybrid Petri Nets*. Springer-Verlag Berlin Heidelberg, 2010.

[7] M. Silva, J. Júlvez, C. Mahulea, and C. Vázquez, “On fluidization of discrete event models: observation and control of continuous Petri nets,” *Discrete Event Dyn. Syst.*, vol. 21, no. 4, pp. 427–497, 2011.

[8] “GreatSPN2.0.” Available at http://www.di.unito.it/~greatspn/index.html, 2008. Dipartamento di information, Università di Torino.

[9] M. Navarro-Gutiérrez, A. Ramírez-Treviño, and M. Silva, “Dual perspectives of equilibrium throughput properties of continuous mono-t-semiflow Petri nets: Firing rate and initial marking variations,” *Automatica*, vol. 136, p. 110074, 2022.

[10] R. C. Carrillo, O. B. Mendoza, A. R. Treviño, and J. R. León, “Sufficient structural conditions for diagnosability and heuristic diagnoser design in timed continuous Petri nets,” *International Journal of Control, Automation and Systems*, vol. 19, no. 11, pp. 3588–3597, 2021.

[11] C. Arzola, C. R. Vázquez, A. Ramírez-Treviño, and M. Silva, “Structural controllability in timed continuous Petri nets,” *Automatica*, vol. 153, p. 111005, 2023.

[12] C. Mahulea, L. Recalde, and M. Silva, “Observability of continuous Petri nets with infinite server semantics,” *Nonlinear Analysis: Hybrid Systems*, vol. 4, no. 2, pp. 219–232, 2010. IFAC World Congress 2008.

[13] M. Dotoli, M. P. Fanti, A. M. Mangini, and W. Ukovich, “A continuous Petri net model for the management and design of emergency cardiology departments,” *IFAC Proceedings Volumes*, vol. 42, no. 17, pp. 50–55, 2009. 3rd IFAC Conference on Analysis and Design of Hybrid Systems.

[14] M. P. Fanti, A. M. Mangini, M. Dotoli, and W. Ukovich, “A three-level strategy for the design and performance evaluation of hospital departments,” *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 43, no. 4, pp. 742–756, 2012.

[15] C. R. Vázquez, A. Ramírez-Treviño, and M. Silva, “Controllability of timed continuous Petri nets with uncontrollable transitions,” *International Journal of control*, vol. 87, no. 3, pp. 537–552, 2014.

[16] R. Ross-León, A. Ramírez-Treviño, J. Ruiz-León, and E. Aguayo-Lara, “Local control law for live and bounded timed continuous Petri nets,” *IFAC Proceedings Volumes*, vol. 47, no. 2, pp. 129–134, 2014.

[17] J. Júlvez, C. Mahulea, and C.-R. Vázquez, “Simhpn: A matlab toolbox for simulation, analysis and design with hybrid Petri nets,” *Nonlinear Analysis: Hybrid Systems*, vol. 6, no. 2, pp. 806 – 817, 2012.

[18] C. Mahulea, L. Recalde, and M. Silva, “Basic server semantics and performance monotonicity of continuous Petri nets,” *Discrete Event Dyn. Syst.*, vol. 19, no. 2, pp. 189–212, 2009.