MODELLING AND DESIGN OF HOSPITAL DEPARTMENTS BY TIMED CONTINUOUS PETRI NETS

Mariagrazia Dotoli[(a1)], Maria Pia Fanti[(a2)], Agostino Marcello Mangini[(a3)], Walter Ukovich[(b4)]

[(a)]Dipartimento di Elettrotecnica ed Elettronica, Polytechnic of Bari (Italy)
[(b)] Dipartimento di Elettrotecnica, Elettronica e Informatica, University of Trieste (Italy)

[(1)]dotoli@deemail.poliba.it, [(2)]fanti@deemail.poliba.it, [(3)]mangini@deemail.poliba.it, [(4)]walter.ukovich@deei.units.it

ABSTRACT
This paper proposes a model to describe in a concise and detailed way the flow of patients in a hospital starting from their arrival to the emergency medical service to the assignment of beds in the suitable department and finally the discharge. The model is based on a continuous Petri nets framework, whose fluid approximation allows us to define suitable optimization problems in order to plan the system capacity, e.g., determining the medical and nursing staff dimension and the number of beds. A case study and a simulation and optimization analysis show the efficiency of the model.

Keywords: hospital department, modeling, continuous Petri nets, performance evaluation.

1. INTRODUCTION

Providing high quality healthcare calls for improved organization and management in hospital departments. In such systems the main problems to face may be classified as follows (Xiong, Zhou, Manikopoulos 1994): i) dimensioning the system, i.e., determining the type and number of resources to provide (staff, rooms, beds, etc.); ii) understanding the workflow and detecting anomalies such as bottlenecks, waiting times, etc.; iii) improving efficiency, i.e., using resources in a better way, by decreasing patients length of stay, reacting to problems such as staff absence, etc.; iv) studying the system reactivity with respect to an increased workload.

Simulation and performance evaluation provides a useful tool for capacity planning and efficiency improvement. The hospital system may be effectively described as a Discrete Event System (DES) in order to perform discrete event simulation (Gunal and Pidd 2007, Kumar and Shim 2007). Moreover, Petri Nets (PNs) may be employed to model emergency medical services and hospitals (S.S. Choi, M.K. Choi, Song and Son 2005, Xiong, Zhou, Manikopoulos 1994). Indeed, PNs are analytical and graphical tools that are suitable for modeling asynchronous, concurrent processes in communication, computer and manufacturing systems. However, PN models suffer from the so called state explosion problem. One way to deal with such a problem is to use some kind of relaxation technique, in particular applicable to some discrete event models. Since hospitals can be considered DESs whose number of reachable states is very large, PN formalisms using fluid approximations provide an aggregate formulation to deal effectively with such complex systems, reducing the dimension of the state space (Silva and Recalde 2004).

This paper proposes a model to describe in a concise framework the flow of patients in a hospital starting from their arrival to the emergency medical service to the assignment of a bed in the suitable department and finally the discharge. The model is based on timed continuous PNs (Silva and Recalde 2004). In particular, places with finite capacities model the medical and nursing staff as well as the available beds and surgery theatres, while transitions describe the flow of patients and the operation/examination/treatment actions. The model provides an effective framework to analyze and simulate the workflow in a generic hospital department. Moreover, the fluid approximation allows us to define suitable optimization problems in order to optimize the chosen performance indices. For instance, this formulation provides a tool to determine the optimal number of beds, doctors and nurses to guarantee efficiency and good flow of discharged patients in the considered hospital department.

2. BASICS ON PETRI NETS

2.1. Discrete Petri Nets

A discrete PN is a bipartite graph described by the four-tuple PN=(P, T, **Pre**, **Post**), where P is a set of places with cardinality m, T is a set of transitions with cardinality n, ***Pre***: $P\times T\rightarrow\mathbb{N}^{m\times n}$ and ***Post***: $P\times T\rightarrow\mathbb{N}^{m\times n}$ are the *pre-* and *post-incidence matrices*, respectively, which specify the arcs connecting places and transitions. More precisely, for each $p\in P$ and $t\in T$ element **Pre**(p,t) (**Post**(p,t)) is equal to a natural number indicating the arc multiplicity if an arc going from p to t (from t to p) exists, and it equals 0 otherwise. Note that $\mathbb{N}$ is the set of non-negative integers.

The $m \times n$ *incidence matrix* of the net is defined as follows:

$$\mathbf{C}=\mathbf{Post}-\mathbf{Pre}. \tag{1}$$

Given a PN, for each place $p \in P$ the following sets of transitions may be defined: ${}^{\bullet}p=\{t \in T: \mathbf{Post}(p,t)>0\}$, named pre-set of p; and $p^{\bullet}=\{t \in T: \mathbf{Pre}(p,t)>0\}$, named post-set of p. Analogously, for each transition $t \in T$ the following sets of places may be defined: ${}^{\bullet}t=\{p \in P: \mathbf{Pre}(p,t)>0\}$, named pre-set of t; and $t^{\bullet}=\{p \in P: \mathbf{Post}(p,t)>0\}$, named post-set of t.

The state of a PN is given by its current marking, which is a mapping $\mathbf{m}: P \rightarrow \mathbb{N}^m$, assigning to each place of the net a nonnegative number of tokens. A PN system $\langle PN, \mathbf{m}_0 \rangle$ is a net PN with an initial marking $\mathbf{m}_0$.

A transition $t \in T$ is enabled at a marking $\mathbf{m}$ if and only if (iff) for each $p \in {}^{\bullet}t$, it holds:

$$\mathbf{m}(p) \geq \mathbf{Pre}(p,t) \tag{2}$$

and we write $\mathbf{m}[t\rangle$ to denote that $t \in T$ is enabled at marking $\mathbf{m}$. When fired, t produces a new marking $\mathbf{m}'$, denoted by $\mathbf{m}[t\rangle \mathbf{m}'$ that is computed by the PN state equation:

$$\mathbf{m}'=\mathbf{m}+\mathbf{C}\,\vec{\boldsymbol{t}}\,, \tag{3}$$

where $\vec{\boldsymbol{t}}$ is the firing vector.

Let σ be a sequence of transitions (or firing sequence). The notation $\mathbf{m}[\sigma\rangle \mathbf{m}'$ indicates that the sequence of enabled transitions σ may fire at $\mathbf{m}$ yielding $\mathbf{m}'$. We also denote $\boldsymbol{\sigma}: T \rightarrow \mathbb{N}^n$ the firing vector associated with a sequence σ, i.e., $\boldsymbol{\sigma}(t)=v$ if transition t is contained v times in σ.

A marking $\mathbf{m}$ is said reachable from $\langle PN, \mathbf{m}_0 \rangle$ iff there exists a firing sequence σ such that $\mathbf{m}_0[\sigma\rangle \mathbf{m}$. The set of all markings reachable from $\mathbf{m}_0$ defines the reachability set of $\langle PN, \mathbf{m}_0 \rangle$ and is denoted by $R(PN, \mathbf{m}_0)=\{\mathbf{m} \mid \sigma: \mathbf{m}_0[\sigma\rangle \mathbf{m}\}$.

2.2. Continuous Petri Nets

This section recalls some basic definitions on the Continuous PN (ContPN) formalism used in this paper. For additional details the interested reader is referred to (Silva and Recalde 2004).

ContPNs are a straightforward relaxation of discrete PNs. Unlike discrete PNs, the amount in which a transition can be fired in ContPNs is not restricted to a natural number. The structure of a ContPN is identical to that of a discrete PN. However, the initial marking $\mathbf{m}_0$ is a vector of non negative *real* numbers. A transition t is enabled at $\mathbf{m}$ iff $\forall p \in {}^{\bullet}t$, $\mathbf{m}(p)>0$. The enabling degree of t is:

$$\mathrm{enab}(t,\mathbf{m})=\min_{p \in {}^{\bullet}t}\left\{\frac{\mathbf{m}(p)}{\mathbf{Pre}(p,t)}\right\}=$$
$$=\max\left\{k \in \mathbb{R}_0^+ \mid k \cdot \mathbf{Pre}(\cdot,t) \leq \mathbf{m}\right\}, \tag{4}$$

with $\mathbb{R}_0^+ = \mathbb{R}^+ \cup \{0\}$ and t can fire in a certain amount $\alpha \in \mathbb{R}$, with $0 \leq \alpha \leq \mathrm{enab}(t,\mathbf{m})$ leading to a new marking $\mathbf{m}' = \mathbf{m} + \alpha \cdot \mathbf{C}(\cdot,t)$, where the incidence matrix $\mathbf{C}$ is also called the *token flow matrix*. If $\mathbf{m}$ is reachable from $\mathbf{m}_0$ by the firing of a sequence σ, the fundamental equation $\mathbf{m} = \mathbf{m}_0 + \mathbf{C} \cdot \boldsymbol{\sigma}$ can be written, where $\boldsymbol{\sigma} \in \mathbb{R}_0^{+n}$ is the firing count vector associated with σ.

2.3. Timed Continuous Petri Nets

In this subsection, timing constraints are added to ContPNs. Time can be associated with places, transitions or arcs. In this paper we assume that time is associated with transitions and the following definitions specify and characterize timed ContPNs.

Definition 1: A timed ContPN $\langle PN, \boldsymbol{\lambda} \rangle$ is the untimed ContPN PN together with a vector $\boldsymbol{\lambda} \in \mathbb{R}^{+n}$, where $\boldsymbol{\lambda}[t_i] = \lambda_i$ is the firing rate of transition t_i.

Definition 2: A timed ContPN system is a tuple $\Sigma = \langle PN, \boldsymbol{\lambda}, \mathbf{m}_0 \rangle$, where $\langle PN, \boldsymbol{\lambda} \rangle$ is a timed ContPN and $\mathbf{m}_0$ is the initial marking of the net.

The fundamental equation describing the timed ContPN system evolution explicitly depends on time τ and is as follows: $\mathbf{m}(\tau) = \mathbf{m}_0 + \mathbf{C} \cdot \boldsymbol{\sigma}(\tau)$. Taking the derivative of this equation with respect to time, we obtain $\dot{\mathbf{m}}(\tau) = \mathbf{C} \cdot \dot{\boldsymbol{\sigma}}(\tau)$. Using the notation $\boldsymbol{f}(\tau) = \dot{\boldsymbol{\sigma}}(\tau)$ to represent the flow of transitions with respect to time, the state equation becomes:

$$\dot{\mathbf{m}}(\tau) = \mathbf{C} \cdot \boldsymbol{f}(\tau)\,. \tag{5}$$

Depending on the flow definition, different semantics have been defined in the literature for continuous timed transitions, the two most important ones being the so-called *infinite server* (or *variable speed*) and *finite server* (or *constant speed*) semantics (Recalde and Silva 2001). The two semantics correspond to two different approximations of the discrete net system that the ContPN relaxes. For a broad class of nets it is formally proven by Mahulea, Recalde and Silva (2006) that the infinite server semantics always provides a better approximation than the finite server one. Hence, in this paper we consider the infinite server semantics.

Under the infinite server semantics, the flow f_i through a timed transition t_i is the product of its speed $\boldsymbol{\lambda}[t_i]$ and its instantaneous enabling degree, as follows:

$$f_i = \boldsymbol{f}\left[t_i\right] = \lambda_i \cdot \text{enab}(t_i, \mathbf{m}) = \lambda_i \cdot \min_{p_j \in {}^\bullet t_i} \left\{ \frac{m_j}{\mathbf{Pre}(p_j, t_i)} \right\} \quad (6)$$

where $m_j = \mathbf{m}(p_j)$ is the marking of place p_j.

2.4. Optimization of Timed Continuous Petri Nets

The use of timed ContPNs to model a DES allows us to consider off-line problems in which, given the system configuration, the objective is to optimally parameterize it. Among the problems belonging to this class are those devoted to the minimization of a cost function that may be formulated in linear terms with respect to the initial marking elements, i.e., as a weighting of the initial marking $\mathbf{b} \cdot \mathbf{m}_0$, where $\mathbf{b}$ represents a gain vector (e.g., if $\mathbf{m_0}(p_1)$ is to be minimized, $\mathbf{b}(p_1)$=1, while the rest of the weights of the gain vector should be zero). This kind of optimization problems, under some conditions depending by the structure of the ContPN described in (Silva and Recalde 2004), admits a particularly elegant and efficient solution by solving a linear programming problem. Indeed, it is either possible to determine the exact value of the optimal initial marking or in the worse case to obtain an upper bound of $\mathbf{m}_0$.

3. THE CONTINUOUS PETRI NET MODEL OF THE HOSPITAL DEPARTMENT

3.1. The System Description

Figure 1 shows the scheme of the basic hospital workflow model. Patients arrive at the emergency department at random time instants. In general, the emergency department serves various patient categories, each characterized by a different degree of urgency. Typically, patients are classified according to four degrees of urgency: life threatening, urgent, serious, non urgent.

Incoming patients are immediately registered and subsequently redirected to a suitable department on the basis of a performed diagnosis. However, life threatening cases are permitted to by-pass the registration and are generally treated in the surgical department. These patients are operated by the doctors and, if the operation is successful, they are subsequently considered as urgent cases and have to wait for the assignment of a bed place.

Usually, other patients wait in a waiting area until the staff is available for registration. After registration, registered patients have to wait in the patient waiting area until the treatment area and associated staff are available. Several non urgent patients do not need to be hospitalized, while other patients are assigned to a suitable department where they wait for a bed to be assigned. After the bed assignment, patients wait in another queue to receive the examination and prescription by the doctors.

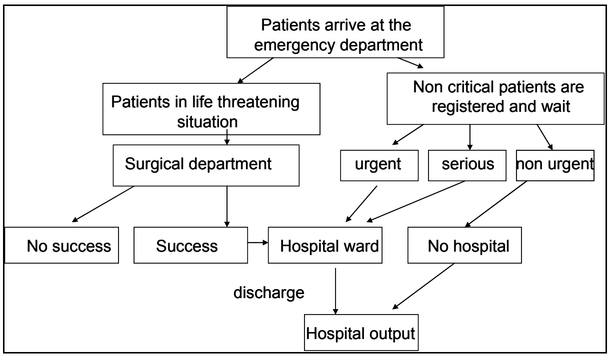

Figure 1. The basic hospital workflow.

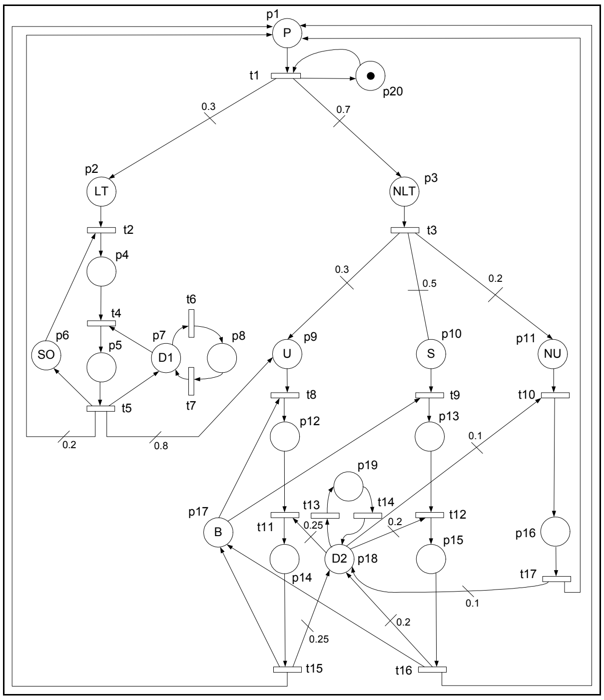

Figure 2. The ContPN model of the patient flow in a typical hospital department.

3.2. The System Model by Timed ContPN

We model the hospital department system in the timed ContPN framework (Silva and Recalde 2004), where places with finite capacities model the staffs of doctors and nurses as well as the available beds. In addition, transitions describe the flow of patients and the actions of doctors and nurses.

The ContPN represented in Figure 2 models the process steps of the patient flow. Marking m_1 represents the number of patients that enter the hospital department. Place p_{20} is added to make the flow through the timed transition t_1 constant and equal to λ_1 that represents the speed with which patients enter the department. Some patients (in our case a percentage of the total that is assigned equal to 30%) are in a life threatening situation (marking m_2), others (marking m_3) have to wait for the registration that is represented by the timed transition t_3. The life threatening patients wait for the operating room to be ready (transition t_2) and successively for the availability of doctors (transition t_4). Markings m_7 and m_8 represent respectively the number of ready to operate and absent surgeons, whereas m_6 represents the number of available beds in the operating room. When the operation is finished (transition t_5) some patients can be dead (in our case a percentage of 20% of the total) while other patients, no longer in a life threatening situation, are now considered as urgent.

As regards the patients that are not in a life threatening situation, after registration they are divided into three degrees of urgency: urgent (marking m_9), serious (marking m_{10}), non urgent (marking m_{11}). The average number of urgent, serious and non urgent patients is given by the weights of the arcs exiting from transition t_3 (in our case corresponding to percentages of 30%, 50% and 20%). The urgent and serious patients have to be hospitalized, hence they have to wait for a bed place (transition t_8 or t_9) and have to be assigned to a doctor to be examined (transition t_{11} or t_{12}). A doctor can be assigned either to urgent patients (in our case to 4 patients at most, since the weight of the arc exiting from place p_{18} is assumed 1/4=0.25) or to serious patients (5 patients at most). When patients recover, they exit from the department (transition t_{15} or t_{16}).

In addition, non urgent patients do not need to be hospitalized, but they only have to wait for the assignment of a doctor (transition t_{10}). In the considered department, a doctor can be assigned to 10 non urgent patients at most. Transition t_{17} represents the exit of non urgent patients from the department.

Finally, markings m_{17} and m_{18} represent the number of available beds and doctors in the department, respectively, whereas marking m_{19} represents the number of absent doctors.

3.3. The Simulation and Optimization of the System

This section presents a system capacity design problem that aims to optimize suitable performance indices on the basis of the defined ContPN model of the hospital. A simulation analysis applies and verifies the obtained policy.

The objective of the considered design problem is establishing the suitable number of doctors, surgeons, beds and surgery theatres in order to obtain good values of some selected performance indices. In our model this means to determine the initial markings of the capacity places p_{18}, p_7, p_{17} and p_6, respectively.

The chosen performance indices are the following: the number of operations per time unit (t.u.), the number of discharged urgent patients per t.u., the number of discharged serious patients per t.u. and the number of discharged non urgent patients per t.u. Hence, we impose a lower bound of the cycle time of the corresponding transitions, i.e., t_5, t_{15}, t_{16} and t_{17}, respectively.

We show that the considered design problem can be solved by defining the following programming problem (Silva and Recalde 2002):

$$\min \mathbf{b}\cdot \mu_0 \tag{7}$$

$$\text{s.t.} \begin{cases} \mu = \mu_0 + \mathbf{C}\cdot\boldsymbol{\sigma} \\ \phi[t] \le \lambda[t]\cdot\dfrac{\mu[p]}{\mathbf{Pre}(p,t)} & \forall t\in T_S,\ \forall p \in {}^\bullet t \\ \phi[t] \le \lambda[t]\cdot\dfrac{\mu[p]}{\mathbf{Pre}(p,t)} & \forall t\in T_U,\ p = {}^\bullet t \\ \mathbf{C}\cdot\phi = 0 \\ \sigma, \mu_0, \phi \ge 0 \\ \phi[t_i] \ge 1/\Gamma_i \end{cases} \tag{8}$$

where $\mathbf{b}$ is the cost vector of the objective function, while μ, μ_0 and $\phi[t]$ are respectively the approximations of marking $\mathbf{m}$, of the initial marking $\mathbf{m}_0$ and of flow $\boldsymbol{f}[t]$. T_U is the set of transitions with one input place, while T_S is the set of remaining transitions, in which synchronizations are present. Moreover, Γ_i represents the cycle time of transition t_i.

In particular, the constraints (8) follow from the definition of the enabling conditions and of the state equations (Silva and Recalde 2002).

Moreover, we add the following constraints on the initial markings: i) $\mu_0[p_1]$ is equal to P, i.e., the population of patients using the medical service (we assume P=5000); ii) $\mu_0[p_{20}]$=1 to impose at each time instant that flow $\boldsymbol{f}[t_1]$ is constant and equal to λ_1; iii) the remaining initial markings are set equal to zero. Formally, the following constraints are added to (8):

$$\begin{cases} \mu_0[p_i] = 0 \quad \text{for i=2,...,5,8,...,16,19} \\ \mu_0[p_1] = \text{P} \\ \mu_0[p_{20}] = 1 \end{cases} \tag{9}$$

Furthermore, the last constraints of (8) impose lower bounds of the transitions that represent the selected performance indices of the model. Hence, the flows of transitions t_5, t_{15}, t_{16} and t_{17} are forced in (8) as follows:

$$\begin{cases} \phi[t_5] \ge 0.90 \\ \phi[t_{15}] \ge 1.35 \\ \phi[t_{16}] \ge 1.05 \\ \phi[t_{17}] \ge 0.40. \end{cases} \tag{10}$$

As previously specified, the aim of the design problem is choosing the minimum number of available doctors, surgeons, beds and surgery theatres in order to satisfy the chosen constraints, in particular equations (10). Consequently, the objective function (7) has to minimize the initial markings $\mu_0[p_6]$, $\mu_0[p_7]$, $\mu_0[p_{17}]$ and $\mu_0[p_{18}]$ that are weighted by a suitable cost vector $\mathbf{b}$. In particular, we assume that the cost of an operating theatre is higher than that of a surgeon, which is in turn higher than that of a doctor, or of a bed setting b_6=20, b_7=10, b_{17}=1, b_{18}=5. Hence, the objective function (7) is specified as follows:

$$\min\left(20\mu_0[p_6]+10\mu_0[p_7]+\mu_0[p_{17}]+5\mu_0[p_{18}]\right). \quad (11)$$

Table 1 reports the firing rates of the transitions. The solution of the programming problem (11)-(8)-(9)-(10) provides the following initial markings:

$$\begin{cases}\mu_0[p_6]=2.4750\\ \mu_0[p_7]=2.2500\\ \mu_0[p_{17}]=43.1667\\ \mu_0[p_{18}]=9.7230\end{cases} \quad (12)$$

Obviously, since the initial markings m_6, m_7, m_{17} and m_{18} have to be defined by integer quantities, we set:

$$\begin{cases}\mathbf{m}_0[p_6]=3\\ \mathbf{m}_0[p_7]=3\\ \mathbf{m}_0[p_{17}]=44\\ \mathbf{m}_0[p_{18}]=10\end{cases} \quad (13)$$

Table 1: Transition firing rates.

Transition	Firing rate λ [time units/day]
t_1	3
t_2	4
t_3	3
t_4	2
t_5	0.5
t_6	1
t_7	10
t_8	3
t_9	2
t_{10}	0.8
t_{11}	1
t_{12}	0.9
t_{13}	1
t_{14}	10
t_{15}	0.05
t_{16}	0.08
t_{17}	4

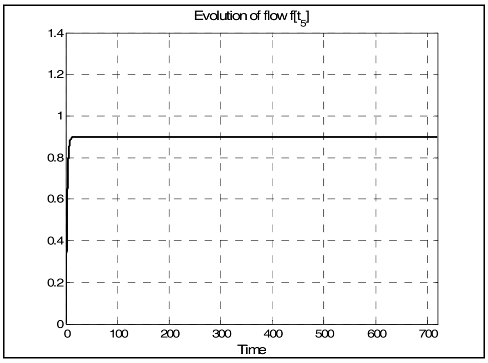

Figure 3: Evolution of flow $f[t_5]$ of finished operations (transition t_5).

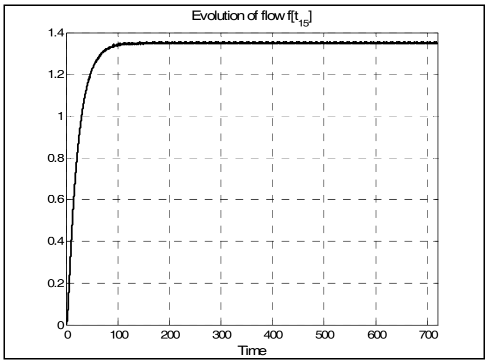

Figure 4: Evolution of flow $f[t_{15}]$ of urgent discharged patients (transition t_{15}).

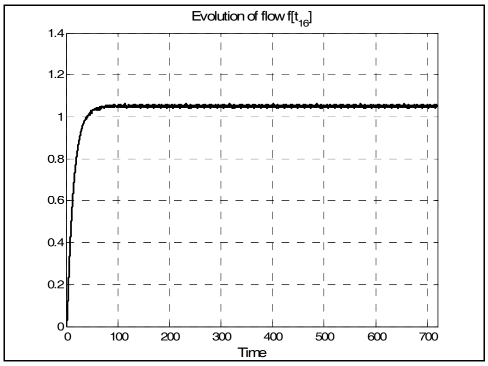

Figure 5: Evolution of flow $f[t_{16}]$ of serious discharged patients (transition t_{16}).

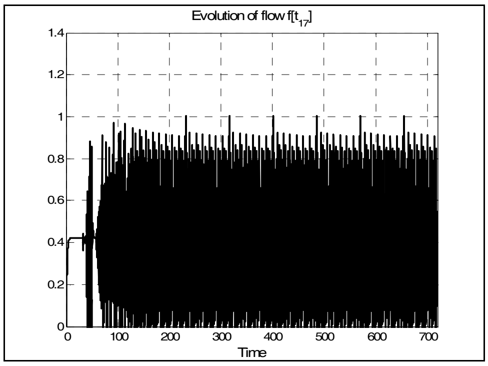

Figure 6: Evolution of flow $f[t_{17}]$ of non urgent discharged patients (transition t_{17}).

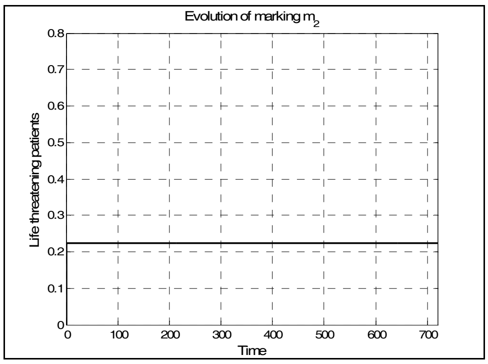

Figure 7: Marking $\mathbf{m}[p_2]$, number of patients in life-threatening situation waiting for treatment.

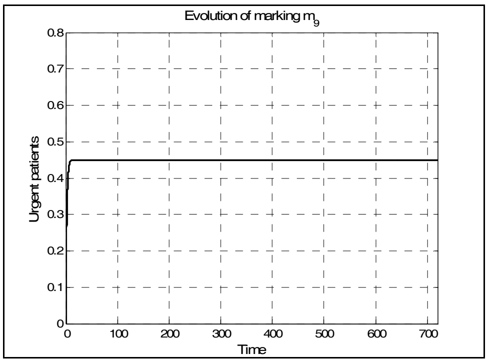

Figure 8: Marking $\mathbf{m}[p_9]$, number of urgent patients waiting for treatment.

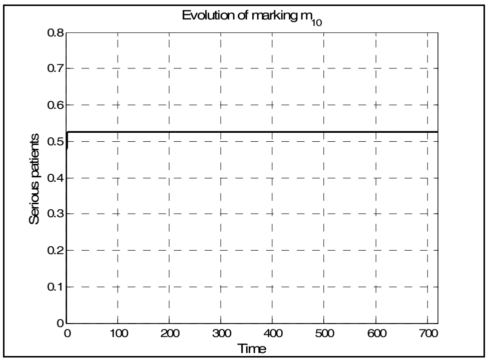

Figure 9: Marking $\mathbf{m}[p_{10}]$, number of serious patients waiting for treatment.

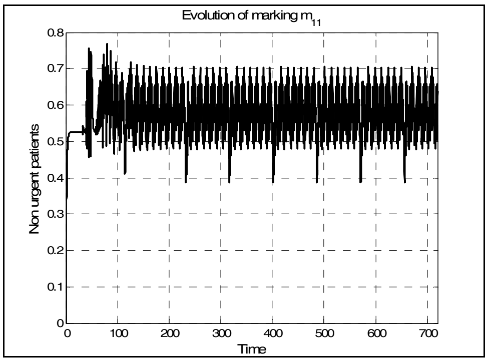

Figure 10: Marking $\mathbf{m}[p_{11}]$, number of non urgent patients waiting for treatment.

Successively, the system dynamics is analyzed via numerical simulation in the MATLAB environment (The Mathworks 2006), a well-known and efficient software that allows us to model systems with a large number of places and transitions. Moreover, such a matrix-based software appears particularly appropriate for simulating the dynamics of ContPNs based on the matrix formulation of the marking update. Furthermore, the MATLAB software is able to integrate modeling and simulation of dynamical systems with the execution of control and optimization algorithms.

A one month simulation (with run time 720 t.u. if we associate one hour to one t.u.) of the ContPN model leads to determine the flows $\boldsymbol{f}[t_i]$ with i=5,15,16,17. We show that each flow reaches a value close to the corresponding imposed bound (10) (see Figures 3 to 6). Note that $\boldsymbol{f}[t_{17}]$ oscillates at steady state, but its average value equals 0.4, in accordance with (10). Hence, the simulation shows that the initial markings provided by the solution of the programming problem are appropriate to reach the objective values of the performance indices. Figures 7 to 10 show the average number of patients (respectively in life threatening situation, urgent, serious and non-urgent) still to be treated. Note that in each category there is on average always less than one waiting patient, showing the success of the system capacity design procedure.

4. CONCLUSION

We propose a continuous Petri net model for analyzing and simulating a generic hospital department workflow, starting from the arrival of patients to their discharge. The fluid approximation allows us to define suitable optimization problems in order to determine the optimal value of key hospital parameters. In particular, we consider the planning of the optimal number of operating theatres, beds, doctors and nurses to guarantee efficiency and minimize waiting times.

5. REFERENCES

Choi, S. S., Choi, M.K., Song, W.J., Son, S.H., 2005. Ubiquitous RFID Healthcare Systems Analysis on PhysioNet Grid Portal Services Using Petri Nets. *Proceedings of IEEE ICICS*, 1254-1258.

Gunal, M.M., Pidd, M., 2007. Interconnected DES Models of Emergency, Outpatient and Inpatient Departments of a Hospital. *Proceedings of the 2007 Winter Simulation Conference*, 1461- 1465.

Kumar, A., Shim S.J., 2007, Eliminating Emergency Department Wait by BPR Implementation. *Proceedings of the 2007 IEEE IEEM,* 1679-1683.

Mahulea, C., Recalde, L., Silva, M. 2006. On Performance Monotonicity and Basic Servers Semantics of Continuous Petri Nets. *Proceedings of WODES'06: 8th International Workshop on Discrete Event Systems*, Ann Arbor, USA, July, 345–351.

Recalde, L., Silva, M. 2001. Petri Nets Fluidification Revisited: Semantics and Steady State. *European Journal of Automation* APII-JESA 35, 435–449.

Silva, M., Recalde, L., 2004. On the Fluidification of Petri Nets: from Discrete to Hybrid and Continuous Models. *Annual Reviews in Control*, 28 (2), 253-266.

Silva, M., Recalde, L., 2002. Petri Nets and Integrality Relaxations: a View of Continuous Petri Nets. *IEEE Transactions on Systems, Man, and Cybernetics*, 32 (4), 314–327.

The Mathworks, 2006. *MATLAB Release Notes for Release 14*. Natick, MA: The Mathworks.

Xiong, H.H., Zhou, M.C., Manikopoulos, C.N., 1994. Modeling and Performance Analysis of Medical Services Systems Using Petri Nets. *Proc. IEEE International Conference on Systems, Man and Cybernetics,* 2339-2342.