International Workshop on Timed Petri Nets (TPN'85), Torino, Italy, July 1–3, 1985, pp.272–278.

PERFORMANCE EVALUATION USING EXTENDED TIMED PETRI NETS

W.M. Zuberek

Department of Computer Science
Memorial University of Newfoundland
St.John's, Canada A1B 3X5

Abstract. It is shown that the behavior of extended limited-choice Petri nets with exponentially distributed firing times can be represented by probabilistic state graphs. For bounded Petri nets the corresponding state graphs are finite, stationary descriptions can thus be obtained by standard techniques used for analysis of continuous-time finite-state homogenous Markov chains. An immediate application of such a model is performance analysis of concurrent systems, and in particular queueing systems with exponentially distributed interarrival and service times. A simple model of an interactive computer system with priority scheduling is used as an illustration of performance evaluation, and a short comparison of timed Petri nets with stochastic Petri nets is given.

1. INTRODUCTION

A Petri net [1,6,12] is known as an abstract, formal model of communication between asynchronous components of a system. The properties, concepts, and techniques of Petri nets are being developed in a search of natural and simple methods for describing and analyzing systems that may exhibit asynchronous and concurrent activities. The major use of Petri nets has been the modeling of systems of events in which it is possible for some events to occur concurrently but there are constraints on the concurrence, precedence, or frequency of these occurences [1,3,11]. Such a model, however, is not complete enough for the study of systems performance since no assumption is made on the duration of systems activities. The timed Petri nets have been introduced by Ramchandani [12] by assigning firing times to the transitions of Petri nets. Sifakis [14] introduced another definition of a timed Petri net by assigning time to places of a net. Merlin and Farber [9] discussed timed Petri nets where a time threshold and maximum delay were assigned to a transition to allow the incorporation of timeouts into protocol models. Razouk [13] used a restricted class of timed Petri nets with enabling as well as firing times to derive performance expressions for communication protocols. Recently Molloy [10] introduced stochastic Petri nets in which transition firing times are exponentially distributed random variables, and the corresponding firing rates are assigned to transitions of a net. Ajmone Marsan, Conte and Balbo [2] generalized stochastic Petri nets introducing two classes of transitions, timed and immediate ones. Timed transitions (as in stochastic nets) have exponentially distributed firing times, while immediate transitions fire in zero time, i.e., they are used to represent logical conditions which do not contribute to delay times. In (basic and generalized) stochastic Petri nets, however, transition firings are intantaneous events (as in place/transition nets), and tokens, during the "firing times" actually remain in corresponding places. Therefore the analysis is based on reachability sets which are generated without timing considerations. This introduces certain restrictions on modelling of even very simple systems.

The method described in this paper is an extension of the approach originated by Ramchandani and used to model the performance of digital systems at the register transfer level [15,16] when fixed (or deterministic) firing times can be used. Basic Petri nets are extended by inhibitor arcs [1,11] in order to model priorities of events and priority-based scheduling disciplines. The use of inhibitor arc is restricted to generalized "free-choice" classes. Firing rates of exponentially distributed firing times are assigned to transitions of a Petri net, and a new state description is introduced which represents the behavior of timed nets in a way similar to homogeneous continuous-time finite-state Markov chains. This directly provides such performance measures as utilization of systems components, average queue lengths, average waiting times and turnaround times or average throughput rates, and at the same time preserves the simplicity of model specification and offers automatic generation of the state space.

This paper is organized in 4 main sections. Section 2 contains definitions of basic concepts for extended Petri nets. Timed Petri nets are introduced in section 3. Application of timed Petri nets to modelling and performance evaluation of computer systems is discussed in section 4. Section 5 compares stochastic Petri nets with M-timed Petri nets.

2. EXTENDED PETRI NETS

An extended Petri net $\mathbf{N}$ is a quadruple $\mathbf{N} = (P, T, A, B)$ where:

P is a finite, nonempty set of places,

T is a finite, nonempty set of transitions,

A is a set of directed arcs which connect places with transitions and transitions with places, and:

$$\forall t \in T\ \exists p_i, p_j \in P\ :\ (p_i, t) \in A \wedge (t, p_j) \in A,$$

B is a set of inhibitor arcs, $B \subset P \times T$, and A and B are disjoint sets.

A place p is an input (or an output) place of a transition t iff there exists an arc (p, t) (or (t, p), respectively) in the set

A. The sets of all input and output places of a transition t are denoted by $Inp(t)$ and $Out(t)$, respectively. Similarly, the sets of input and output transitions of a place p are denoted by $Inp(p)$ and $Out(p)$. A place p is an inhibitor place of a transition t iff there exists an inhibitor arc (p,t) in the set B. The set of all inhibitor places of a transition t is denoted by $Inh(t)$.

A marked Petri net $\mathbf{M}$ is a pair $\mathbf{M} = (\mathbf{N}, m_0)$ where:

$\mathbf{N}$ is an extended Petri net, $\mathbf{N} = (P, T, A, B)$,

m_0 is an initial marking function which assigns a nonnegative integer number of so called tokens to each place of the net, $m_0 : P \to \{0, 1, ...\}$.

Let any function $m : P \to \{0, 1, ...\}$ be called a marking of a net $\mathbf{N} = (P, T, A)$.

A transition t is enabled by a marking m iff every input place of this transition contains at least one token and every inhibitor place of t contains zero tokens. The set of all transitions enabled by a marking m is denoted by $T(m)$.

A place p is shared iff it is an input place for more than one transition. A shared place p is guarded iff for each two different transitions t_i and t_j sharing p there is another place p_k such that p_k is in the input set of t_i and in the inhibitor set of t_j:

$$\forall t_i \in Out(p)\ \forall t_j \in Out(p) - \{t_i\}\ \exists p_k \in P - \{p\}\ :\ (p_k, t_i) \in A \wedge (p_k, t_j) \in B\ \vee\ (p_k, t_i) \in A \wedge (p_k, t_j) \in A$$

i.e., no two transitions from the set $Out(p)$ can be enabled by the same marking m. A net is conflict-free iff all its shared places are guarded.

A shared place p is free-choice (or extended free-choice [6]) iff the input sets and inhibitor sets of all transitions sharing p are identical, i.e., iff:

$$\forall t_i, t_j \in Out(p)\ :\ Inp(t_i) = Inp(t_j) \wedge Inh(t_i) = Inh(t_j).$$

A shared place p is limited-choice iff the set of transitions sharing p, $Out(p)$, can be subdivided into two disjoint sets $Nch(p)$ and $Fch(p)$ such that p is free-choice with respect to $Fch(p)$

$$\forall t_i, t_j \in Fch(p)\ :\ Inp(t_i) = Inp(t_j) \wedge Inh(t_i) = Inh(t_j).$$

and p is guarded with respected to $Nch(p)$

$$\forall t_i \in Nch(p)\ \forall t_j \in Out(p) - \{t_i\}\ \exists p_k \in P - \{p\}\ :\ (p_k, t_i) \in A \wedge (p_k, t_j) \in B\ \vee\ (p_k, t_i) \in A \wedge (p_k, t_j) \in A.$$

In other words, for a limited-choice place p, each marking m enables either at most one transition from the set $Nch(p)$, or all transitions in the set $Fch(p)$, and then there is a "limited free choice" (i.e., free choice restricted to the set $Fch(p)$) of selecting a transition for firing. It can be observed that a free-choice place p is a special case of a limited-choice place when there is no restriction, i.e., the set $Nch(p)$ is empty, while a guarded place p is another special case when the "restricted" set $Fch(p)$ is empty.

A net is limited-choice iff all its shared places are limited-choice. Only limited-choice extended Petri nets are considered in this paper.

Every transition enabled by a marking m can fire. When a transition fires, a token is removed from each of its input places and a token is added to each of its output places. This determines a new marking in a net, a new set of enabled transitions, and so on.

A marking m_j is directly reachable from a marking m_i in a net $\mathbf{N}$, $m_j \leftarrow m_i$, iff there exists a transition t enabled by the marking m_i, $t \in T(m_i)$, such that

$$\forall p \in P\ :\ m_j(p)\ =\ \begin{cases} m(p) - 1, & \text{if } p \in Inp(t) - Out(t); \\ m(p) + 1, & \text{if } p \in Out(t) - Inp(t); \\ m(p), & \text{otherwise.} \end{cases}$$

A marking m_j is (generally) reachable from a marking m_i in a net $\mathbf{N}$, $m_j \stackrel{*}{\leftarrow} m_i$, if there exists a (possibly empty) sequence of markings $(m_{i_0} m_{i_1} m_{i_2} ... m_{i_k})$ such that $m_{i_0} = m_i$, $m_{i_k} = m_j$, and each marking m_{i_ℓ} is directly reachable from the marking $m_{i_{\ell-1}}$ for $\ell = 1, ..., k$.

A set $M(\mathbf{M})$ of reachable markings of a marked Petri net $\mathbf{M} = (\mathbf{N}, m_0)$ is the set of all markings which are reachable from the initial marking m_0, $M(\mathbf{M}) = \{m \mid m \stackrel{*}{\leftarrow} m_0\}$.

A marked net $\mathbf{M}$ is bounded if there exists a positive integer k such that each marking in the set $M(\mathbf{M})$ assigns at most k tokens to each place of the net

$$\exists k > 0\ \forall m \in M(\mathbf{M})\ \forall p \in P\ :\ m(p) \leq k.$$

If a marked net $\mathbf{M}$ is bounded, its reachability set $M(\mathbf{M})$ is finite. Only bounded Petri nets are considered in this paper.

An enable function of a marking m in a net $\mathbf{N}$ is any function $e : T \to \{0, 1, ...\}$ such that after initiating all firings indicated by nonzero values of e, the set of enabled transitions is empty, i.e. $E(m') = \emptyset$, where:

$$\forall p \in P\ :\ m'(p)\ =\ m(p) - \sum_{t \in Out(p)} e(t)$$

and the marking m'', created by firing all transitions indicated by nonzero values of e, is reachable from m, $m'' \in M((\mathbf{N}, m))$, where:

$$\forall p \in P\ :\ m''(p)\ =\ m(p) + \sum_{t \in Inp(p)} e(t) - \sum_{t_i \in Out(p)} e(t).$$

i.e., any function which indicates (by nonzero values) all those transitions which can fire simultaneously (and some of the transitions may fire several times). For conflict-free nets, for each marking m there exists exactly one enable function which is determined by

$$\forall t \in T\ :\ e(t)\ =\ \begin{cases} 0, & \text{if } \sum_{p \in Inh(t)} m(p) > 0; \\ \min_{p \in Inp(t)}(m(p)), & \text{otherwise.} \end{cases}$$

For limited-choice nets, there may be several different enable functions for the same marking m. The set of all enable functions of a marking m is denoted by $E(m)$.

3. TIMED PETRI NETS

In a timed Petri net, each transition t takes a positive time to fire. When a transition t is enabled, a firing can be initiated by removing a token from each of t's input places. This token remains in the transition t for the "firing time", and then the firing terminates by adding a token to each of t's output places. Each of the firings is initiated in the same instant of time in which it is enabled. If a transition is enabled while it fires, a new, independent firing can be initiated. For conflict-free nets all enabled transitions immediately initiate their firings since each marking of a net uniquely determines the enable function. If a net contains conflicts, and there are several different enable functions for the same marking, the selection of an actual enable function is assumed to be a random process which can be described by corresponding probabilities.

The operation of a timed Petri net can thus be considered as taking place in "real time", and it is assumed that it starts at the time $\tau = 0$. At this moment, the firings indicated by an enable function $e \in E(m_0)$ are initiated and the tokens are removed from the input places. Then, after the time corresponding to the shortest "firing time" of the transitions which initiated firings, the tokens are deposited in appropriate output places creating a new marking, a new set of enabled transitions, and so on.

The firing times of transitions can be described in several ways. In D-timed Petri nets [7,15,16] they are deterministic (or constant), i.e., there is a positive (rational) number assigned to each transition of a net which determines the time of firing. In M-timed Petri nets [17] (and stochastic Petri nets [2,10]), the firing times are exponentially distributed random variables, and the corresponding firing rates are assigned to transitions of a net. The memoryless property of exponential distributions is the key factor in analysis of M-timed Petri nets.

An M-timed extended limited-choice Petri net $\mathbf{T}$ is a triple $\mathbf{T} = (\mathbf{M}, c, r)$ where:

$\mathbf{M}$ is an extended limited-choice marked Petri net, $\mathbf{M} = (\mathbf{N}, m_0)$, $\mathbf{N} = (P, T, A, B)$,

c is a choice function which assigns a "free-choice" probability to each transition t of the net in such a way that for each limited-choice place p of $\mathbf{N}$:

$$\sum_{t \in Fch(p)} c(t) \ = \ 1,$$

and for all transitions which do not belong to limited-choice classes, $c(t) = 1$,

r is a firing rate function which assigns a positive real number $r(t)$ to each transition t of the net, $r : T \rightarrow \mathbf{R}^+$, and $\mathbf{R}^+$ denotes the set of positive real numbers; the firing time of a transition t is a random variable $v(t)$ with the distribution function

$$\text{Prob}[v(t) > x] \ = \ e^{-r(t)*x}, \ x > 0.$$

The memoryless property of exponential distributions means that if the duration v of a certain activity (e.g., the firing time) is distributed exponentially with parameter r, and if that activity is observed at time y after its beginning, then the remaining duration of the activity is independent of y and is also distributed exponentially with parameter r:

$$\text{Prob}[v < x + y \mid v > y] \ = \ \text{Prob}[v > x] \ = \ e^{-r*x}.$$

The exponential distribution is the only continuous distribution with that property.

Also, if v and w are the durations of two independent simultaneous activities a and b, distributed exponentially with parameters q and r, respectively, then the time interval u until the first completion of an activity (a or b) is distributed exponentially with parameter $(q + r)$, and the probability that the activity a will complete first is equal to $q/(q+r)$, while the same probability for the activity b is equal to $r/(q + r)$. These results can be generalized in an obvious way to any number of activities. They are used in the description of state transitions.

A state s of an M-timed Petri net $\mathbf{T}$ is a pair $s = (m, f)$ where:

m is a marking function, $m : P \rightarrow \{0, 1, ...\}$,

f is a firing function which indicates (for each transition of the net) the number of active firings, i.e., the number of firings which have been initiated but are not yet terminated, $f : T \rightarrow \{0, 1, ...\}$.

An initial state s of a net $\mathbf{T}$ is a pair $s = (m_i, f_i)$ where f_i is an enable function from the set $E(m_0)$, and the marking m_i is defined by

$$\forall p \in P \ : \ m_i(p) \ = \ m_0(p) - \sum_{t \in Out(p)} f_i(t).$$

A limited-choice M-timed net $\mathbf{T}$ may have several different initial states.

A state $s_j = (m_j, f_j)$ is directly (t_k, e_ℓ)-reachable from the state $s_i = (m_i, f_i)$, $s_j \leftarrow s_i$, iff the following conditions are satisfied:

1. $f_i(t_k) > 0$,
2. $\forall p \in P \ : \ m_{ij}(p) \ = \ m_i(p) + \begin{cases} 1, & \text{if } p \in Out(t_k); \\ 0, & \text{otherwise;} \end{cases}$
3. $e_\ell \in E(m_{ij})$,
4. $\forall p \in P \ : \ m_j(p) \ = \ m_{ij}(p) - \sum_{t \in Out(p)} e_\ell(t)$,
5. $\forall t \in T \ : \ f_j(t) \ = \ f_i(t) + e_\ell(t) - \begin{cases} 1, & \text{if } t = t_k; \\ 0, & \text{otherwise.} \end{cases}$

The state s_j which is directly (t_k, e_ℓ)-reachable from the state s_i is thus obtained by the termination of a t_k firing (1), updating the marking of a net (2), and then initiating new firings (if any) which correspond to the enable function e_ℓ from the set $E(m_{ij})$ (3, 4 and 5).

It can be observed that the e_ℓ functions in the definition of directly reachable states can assume the values 0 and 1 only.

Similarly as for marked nets, a state s_j is (generally) reachable from a state s_i if there is a sequence of directly reachable states from the state s_i to the state s_j. Also, a set $S(\mathbf{T})$ of reachable states is defined as the set of all states of a net $\mathbf{T}$ which are reachable from the initial states of the net $\mathbf{T}$. For bounded nets the sets of reachable states are finite.

A state graph $\mathbf{G}$ of an M-timed Petri net $\mathbf{T}$ is a labeled directed graph $\mathbf{G}(\mathbf{T}) = (V, D, b)$ where:

V is a set of vertices which is equal to the set of reachable states of the net $\mathbf{T}$, $V = S(\mathbf{T})$,

D is a set of directed arcs, $DsubsetVtimesV$, such that $(ssubi, ssubj)$ is in D iff $ssubj$ is directly reachable from $ssubi$,

b is a function which assigns the probability of transition from s_i to s_j to each arc (s_i, s_j) in the set D, $b : D \rightarrow [0, 1]$, in such a way that if s_j is directly (t_k, e_ℓ)-reachable from s_i, then

$$b(s_i), s_j) \;=\; r(t_k) * f_i(t_k) * \prod_{t \in T} c(t)^{e_\ell}(t) / \sum_{t \in T} r(t) * f_i(t).$$

Example. For the timed Petri net shown in Fig.1a (as usually, places are represented by circles, transitions by bars, inhibitor arcs by small circles instead of arrowheads, the initial marking by dots inside places, and the firing rate function and the choice function are given as an additional descriptions of transitions), the state graph is shown in Fig.1b, and the derivation of the set $S(\mathbf{T})$ of reachable states is shown in Tab.1. It can be observed that the net contains two shared places, p_1 and p_4; p_4 is a free-choice (or extended free-choice [6]) place, while p_1 is a limited-choice place with $Nch(p_1) = \{t_2\}$ and $Fch(p_1) = \{t_3, t_4\}$. The choice function c is thus effectively defined for t_3 and t_4 only (for all remaining transitions $c(t) = 1$), with $c(t_3) = 0.25$ and $c(t_4) = 0.75$. □

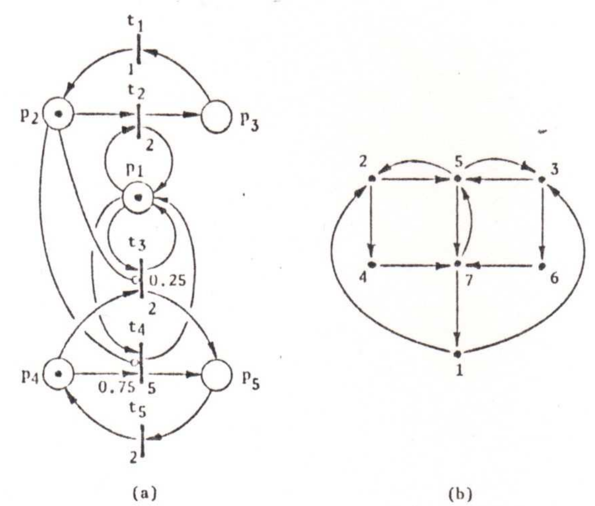

Fig.1. Timed Petri net (a) and its state graph (b).

Tab.1. The set of reachable states for m_0=[1,1,0,1,0].

i	m_i 1 2 3 4 5	f_i 1 2 3 4 5	t_k	m_{ij} 1 2 3 4 5	j	$b(s_i, s_j)$
1	0 0 0 1 0	0 1 0 0 0	2	1 0 1 1 0	2	0.750
					3	0.250
2	0 0 0 0 0	1 0 0 1 0	1	0 1 0 0 0	4	0.167
			4	1 0 0 0 1	5	0.833
3	0 0 0 0 0	1 0 1 0 0	1	0 1 0 0 0	6	0.333
			3	1 0 0 0 1	5	0.667
4	0 1 0 0 0	0 0 0 1 0	4	1 1 0 0 1	7	1.000
5	1 0 0 0 0	1 0 0 0 1	1	1 1 0 0 0	7	0.333
			5	1 0 0 1 0	2	0.500
					3	0.167
6	0 1 0 0 0	0 0 1 0 0	3	1 1 0 0 1	7	1.000
7	0 0 0 0 0	0 1 0 0 1	2	1 0 1 0 0	5	0.500
			5	0 0 0 1 0	1	0.500

It can be observed that the state graph of a limited-choice bounded M-timed Petri net is a continuous-time finite-state homogenous Markov chain in which the average time $h(s)$ spent in the state $s = (m, f)$, $s \in S(\mathbf{T})$, (or the sojourn time) is equal to

$$h(s) \;=\; \frac{1}{\sum_{t \in T} r(t) * f(t)}.$$

Therefore, the stationary probabilities $y(s)$ of the states s can be obtained by solving a system of simultaneous linear equations [5,7]

$$\begin{cases} \sum_{(s_i, s_j) \in D} b(s_i, s_j) * y(s_j)/h(s_j) \;=\; y(s_i)/h(s_i), \\ \qquad\qquad\qquad\qquad\qquad\qquad i = 1, ..., K-1; \\ \sum_{1 \le i \le K} y(s_i) \;=\; 1 \end{cases}$$

where K is the number of states in the set $S(\mathbf{T})$.

For the net from the example, the sojourn times $h(s)$ and the stationary probabilities $y(s)$ are given in Tab.2.

Tab.2. Stationary probabilities for $m_0 = [1, 1, 0, 1, 0]$.

i	$h(s_i)$	$y(s_i)$
1	0.500	0.155
2	0.167	0.136
3	0.333	0.091
4	0.200	0.027
5	0.333	0.391
6	0.500	0.045
7	0.250	0.155

4. PERFORMANCE EVALUATION

A very simple closed-network model of an interactive system with 2 classes of users (and jobs) and nonpreemptive priority scheduling is shown in Fig.2a. The system contains one central server P_c with two queues of waiting jobs Q_1 and Q_2 (for class-1 and class-2 jobs, respectively), and n_1 terminals in class-1 and n_2 terminals in class-2. All class-1 jobs have higher priority than the class-2 ones, i.e., if class-1 and class-2 jobs are waiting for the central server, the class-1 jobs will receive the service first. The class-1 jobs are statistically identical while there are two types of class-2 jobs, "short" and "long" ones, with probabilities a and $(1 - a)$, respectively. To simplify the solution, it is assumed that the jobs of the same class are served by the First-Come-First-Served discipline, and that all the service times and the terminal times are exponentially distributed. Under these assumptions the number of jobs in the system (i.e., in the waiting queues and in the server) is a finite-state continuous-time homogeneous Markov chain [5,7]. For n_1=n_2=1 there are 7 states of the Markov chain:

0: no jobs in the system;
1: a class-1 job in P_c;
2: a short class-2 job in P_c;
3: a long class-2 job in P_c;
4: a class-1 job in P_c, 1 job in Q_2;
5: a short class-2 job in P_c, 1 job in Q_1;
6: a long class-2 job in P_c, 1 job in Q_1.

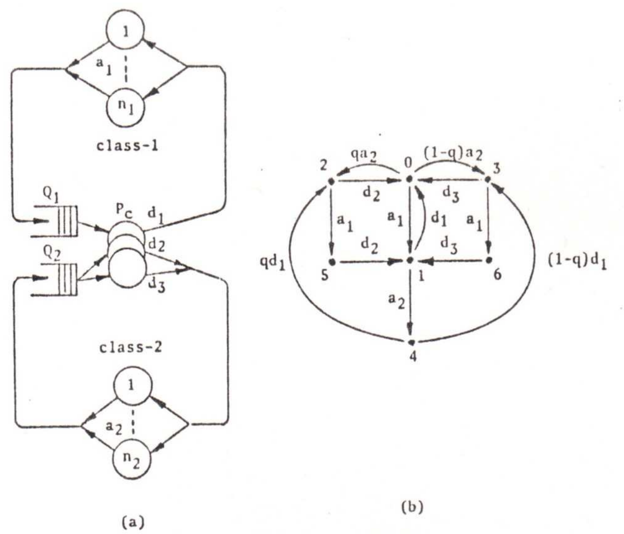

Fig.2. Closed-network model of an interactive system (a) and its transition-rate diagram (b):

The corresponding transition-rate diagram [4] is shown in Fig.2b where a_1 and a_2 denote the terminal rates for class-1 and class-2, respectively, and d_1, d_2 and d_3 denote the service rates for class-1, and short and long class-2 jobs, respectively.

The same system can be modeled by the timed Petri net shown in Fig.1a. The transitions t_2, t_3 and t_4 correspond to the central server processing class-1 jobs (t_2), long (t_3) and short (t_4) class-2 jobs with the service rates (or the firing rates) equal to 2, 2 and 5, respectively. The places p_2 and p_4 model the waiting queues (for class-1 and class-2 jobs, respectively). The transitions t_1 and t_5 correspond to the class-1 and class-2 terminals, respectively (with the terminal rates or the firing rates equal to 1 for class-1 and 2 for class-2 jobs). The initial number of tokens in the places p_2 and p_3 represents the number of class-1 terminals, n_1, and the initial number of tokens assigned to places p_4 and p_5 indicates the number of class-2 terminals, n_2. The place p_1 and its initial number of tokens model the number of servers (or server channels), in this case 1.

For the initial marking $m_0 = [1, 1, 0, 1, 0]$ there are 7 states of the net (Tab.1) and 7 states of the Markov chain (Fig.2b). The correspondence between the Petri net states and the states of the Markov chain is shown in the Tab.3.

Tab.3. Petri net states and Markov chain states.

Petri net	Markov chain
1	4
2	2
3	3
4	5
5	0
6	6
7	1

It can be observed that the Markov chain from Fig.2b (with a_1=1, $a_2 = 2$, $d_1 = d_2 = 2$, $d_3 = 5$, and q=0.75) is isomorphic to the state graph from Fig.1b augmented by state-transition rates, determined by the state-transition probabilities $b(s_i, s_j)$ and the average sojourn times $h(s)$, $s \in S(\mathbf{T})$. For example, the $(ssub7, ssub5)$ transition rate is equal to 0.5/0.25=2, and is the same as the Markov (1,0) transition rate $dsub2$=2. The solutions of both models must thus be equivalent.

Since the server is idle in the state s_5 only ($m_5(p_1) = 1$, and also Markov state "0"), the equilibrium probability that the system is idle is equal to the stationary probability $y(s_5) = 0.391$ (Tab.2). Then the utilization of the system is immediately 1-0.391=0.609 which is composed of 0.310 for class-1 jobs ($y(s_1) + y(s_7)$) and 0.299 for class-2 jobs. The average throughput rates can be obtained from the server's load. Since the average service time for class-1 jobs is equal to 0.5 time units, and the server utilization for this class is 0.310, then the average throughput rate for class-1 jobs is equal to 0.310/0.5=0.62 jobs per time unit, and the average turnaround time is equal to 1/0.62=1.613 time units. Since for class-1 jobs the average terminal time and the average service time are equal to 1.0 and 0.5 time unit, respectively, the class-1 jobs spend, on average, 1.613-1.0-0.5=0.113 time units in the waiting queue. Similarly, for class-2 jobs the average throughput rate is equal to 0.299/(0.25*0.5+0.75*0.2)=1.087 jobs per time unit, and the average turnaround time is equal to 0.920 time units. Since, for class-2 jobs, the average terminal time and the average service time are equal to 0.5 and 0.275 time units, respectively, the average waiting time is equal to 0.145 time units.

For $n_1 = n_2 = 2$, there are 19 states in the set $S(\mathbf{T})$, and the same performance measures are as follows:

stationary probability that the system is idle	0.100
utilization of the system	0.900
class-1 utilization of the system	0.537
average class-1 throughput rate	0.931
average class-1 turnaround time	2.148
average class-1 waiting time	0.648
class-2 utilization of the system	0.363
average class-2 throughput rate	1.320
average class-2 turnaround time	1.515
average class-2 waiting time	0.740

Many other results can be obtained in a very similar way.

5. STOCHASTIC AND M-TIMED PETRI NETS

In stochastic Petri nets [2,10], the firing times are assigned to transitions, however, for the whole period of firing, tokens remain in corresponding places and the actual "transition" occurs at the end of firing time. In consequence, the state space is easily determined as the set of reachable markings, and the probabilities of transitions between the states (or markings, in fact):

$$b(m_i, m_j) \;=\; \begin{cases} r(t_k) * h(m_i), & \text{if } m_j \stackrel{t_k}{\leftarrow} m_i; \\ 0, & \text{otherwise.} \end{cases}$$

The average sojourn times are defined on the basis of firing rates assigned to enabled transitions [2,10]:

$$h(m) \;=\; \frac{1}{\sum_{t \in T(m)} r(t)}$$

where, as before, $b(m_i, m_j)$ is the probability of transition from the state m_i to the state m_j, $h(m)$ is the average time spent in the state m, r is the firing-rate function, $r : T \to \mathbf{R}^+$, and $T(m)$ denotes the set of transitions enabled by the marking m.

Stochastic Petri nets do not consider "multiple" firings, i.e., states in which a transition is enabled several times. Since the set of reachable markings is constructed by "firing one transition at a time", the stochastic approach should be either restricted to "1-enable" (or singular) nets, or all such cases must be taken into account separately. Fig.3 shows 3 different Petri nets which model three different queueing systems, $M/M/1//3$ (a), $M/M/2//3$ (b), and $M/M/inf//3$ (c) [7].

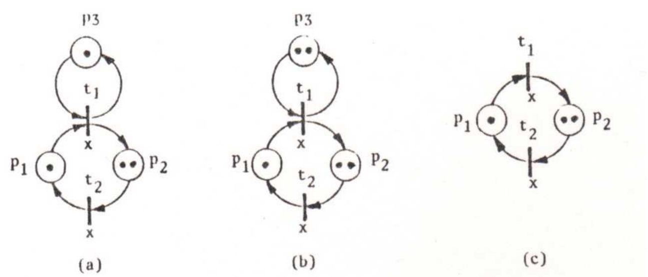

Fig.3. Petri net models of M/M/1/3 (a), M/M/2//3 (b), and M/M/inf//3 (c) systems.

The state graphs obtained by the stochastic approach are shown in Fig.4a,b,c, where the state-transition rates correspond to all firing rates arbitrarily assumed to be equal to x (which does not really matter). The graphs are isomorphic, the behavior of these three stochastic nets must thus be identical, while the models are definitely different.

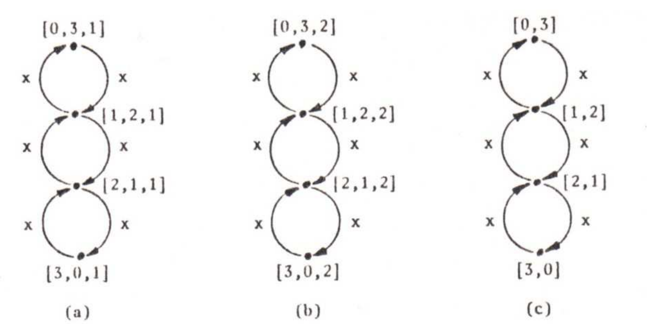

Fig.4. Stochastic transition-rate diagrams of M/M/1/3 (a), M/M/2//3 (b) and M/M/inf//3 systems.

Fig.5a,b,c show the state-transition-rate diagrams obtained by the analytical approach [5,7].

The results of M-timed Petri net analysis are exactly the same. It can be observed that (in this case) the stochastic net approach can easily be modified by introducing an "enable function" $e : T \to \{0, 1, ...\}$ (as in section 2) and extending the formulas:

$$b(m_i, m_j) \;=\; \begin{cases} e_{m_i}(t_k) r(t_k) * h(m_i), & \text{if } m_j \stackrel{t_k}{\leftarrow} m_i; \\ 0, & \text{otherwise}; \end{cases}$$

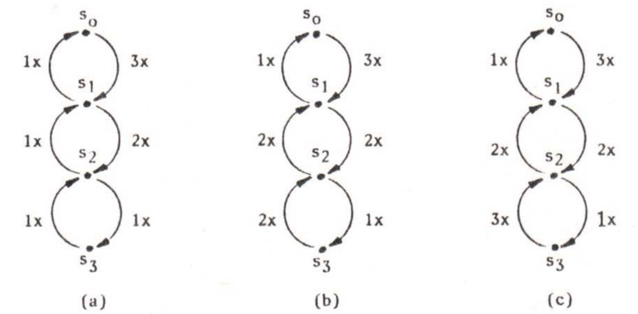

Fig.5. Analytical transition-rate diagrams of M/M/1/3 (a), M/M/2//3 (b) and M/M/inf//3 systems.

$$h(m) \;=\; \frac{1}{\sum_{t \in T(m)} e_m(t) * r(t)}.$$

The second difference between stochastic and M-timed Petri nets is in the "interpretation" of free-choice (extended free-choice, limited-choice, etc.) classes. In M-timed Petri nets the "choice" is separated from the (selected) firing because during firing tokens remain "in" (selected earlier) transitions. Consequently, all free-choice classes are described by "independent" probabilities which are not related to firing rates of transitions. In stochastic nets, during firing periods tokens remain in places, all free-choice transitions are enabled at the same time, and consequently, the "choice" probabilities are determined by the firing rates of free-choice transitions. Moreover, this is the only way to determine the (relative) frequencies of random events, so all probabilities do depend upon firing rates (in generalized stochastic nets [2] the free-choice classes are "independent" only at the "immediate" level).

Fig.6a,b show Petri net models of two different queueing systems, $M/M/1//1$ (a), and $M/H_2/1//1$ (b) [4,7].

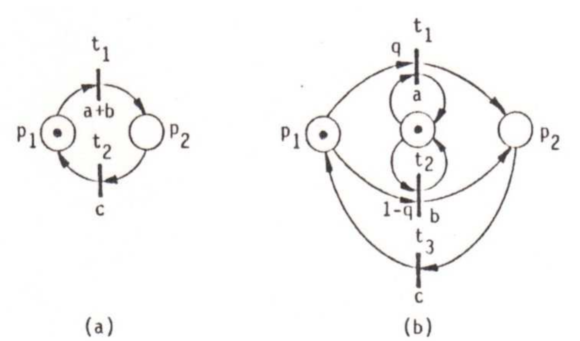

Fig.6. Petri net models of M/M/1//1 (a) and $M/H_2/1//1$ (b) systems.

For both nets, stochastic approach results in the state-transition-rate diagram shown in Fig.7a, while M-timed Petri net state-transition diagrams are different (Fig.7a,b) and, again, correspond to the analytical solutions [7]. Stochastic and M-timed Petri nets must thus use quite different modelling methods.

Finally, for analysis of bulk (or burst) arrivals and/or services [7], i.e., for systems with multiple simultaneous

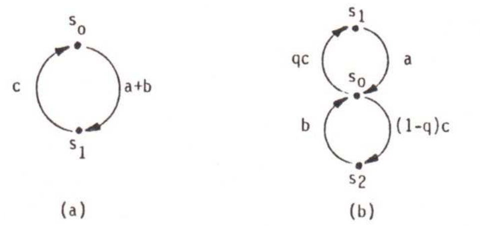

Fig.7. Transition-rate diagrams of M/M/1//1 (a) and $M/H_2/1//1$ (b) systems.

events, M-timed approach can be generalized in a rather straightforward way [17] while stochastic approach may require a rather substantial extension.

6. CONCLUDING REMARKS

Even the very simple example of an interactive system modelling illustrates the characteristic features of timed Petri nets. Models are usually quite simple, and their parameters correspond in a very natural way to components or activities of the modelled systems (e.g., the number of users, the number of processors). The state space can easily be generated from model specifications, and since the states of the modelling net directly correspond to the states of the modelled system, a verification step is provided which is not available in analytical modelling. A pilot version of a program which analyzes extended and generalized M-timed Petri nets is given in [17].

The class of timed Petri nets discussed in the paper is restricted in several ways (limited-choice, bounded nets), some of the restrictions, however, can be removed easily by appropriate extensions of the formalism. In fact, nets with more general conflicts can be handled in a very similar way provided the probabilities of conflicting firings are known and included in the state description.

REFERENCES

1. T. Agerwala, “Putting Petri nets to work”; IEEE Computer Magazine, vol.12, no.12, pp.85-94, 1979.
2. M. Ajmone Marsan, G. Conte, G. Balbo, “A class of generalized stochastic Petri nets for the performance evaluation of multiprocessor systems”; ACM Trans. on Computer Systems, vol.2, no.2, pp.93-122, 1984.
3. M. Diaz, “Modeling and analysis of communication and cooperation protocols using Petri net based models”; Computer Networks, vol.6, no.6, pp.419-441, 1982.
4. D. Ferrari, “Computer systems performance evaluation”; Prentice-Hall 1978.
5. E. Gelenbe, I. Mitrani, “Analysis and synthesis of computer systems”; Academic Press 1980.
6. M. Jantzen, R. Valk, “Formal properties of place/-transition nets”; in **Net Theory and Applications**, W. Brauer (ed.), Lecture Notes in Computer Science 84, pp.165-212, Springer-Verlag 1980.
7. L. Kleinrock, “Queueing systems”, vol.1: Theory, vol.2: Computer Applications; J. Wiley & Sons 1975, 1976.
8. J. Magott, “Performance evaluation of concurrent systems using Petri nets”; Information Processing Letters, vol.18, no.1, pp.7-13, 1984.
9. P.M. Merlin, D.J. Farber, “Recoverability of communication protocols - implications of a theoretical study”; IEEE Trans. on Communications, vol.24, no.9, pp.1036-1049, 1976.
10. M.K. Molloy, “Performance analysis using stochastic Petri nets”; IEEE Trans. on Computers, vol.31, no.9, pp.913-917, 1982.
11. J.L. Peterson, “Petri net theory and the modeling of systems”, Prentice-Hall 1981.
12. C. Ramchandani, “Analysis of asynchronous concurrent systems by timed Petri nets”; Project MAC Technical Report MAC-TR-120, Massachusetts Institute of Technology, Cambridge MA, 1974.
13. R.R. Razouk, “The derivation of performance expressions for communication protocols from timed Petri nets”; Computer Communication Review, vol.14, no.2, pp.210-217, 1984.
14. J. Sifakis, “Use of Petri nets for performance evaluation”; in **Measuring, modelling and evaluating computer systems**, pp.75-93, North-Holland 1977.
15. W.M. Zuberek, “Timed Petri nets and preliminary performance evaluation”; Proc. IEEE 7-th Annual Symp. on Computer Architecture, La Baule, France, pp.89-96, 1980.
16. W.M. Zuberek, “Application of timed Petri nets to analysis of multiprocessor realizations of digital filters”; Proc. 25 Midwest Symp. on Circuits and Systems, Houghton MI, pp.134-139, 1982.
17. W.M. Zuberek, “M-timed Petri nets, modelling and performance evaluation of systems”; Technical Report #8503, Department of Computer Science, Memorial University of Newfoundland, St. John’s, Canada A1C 5S7, 1985.