

## MODELING AND SIMULATION BY HYBRID PETRI NETS

Hassane Alla

Latéfa Ghomri

GIPSA-LAB  
Grenoble University  
38400 Saint Martin d'Hères. FRANCE

MELT Lab  
Tlemcen University  
FSI UABB, Tlemcen, 13000, ALGERIA

### ABSTRACT

Petri nets (PNs) are widely used to model discrete event dynamic systems (computer systems, manufacturing systems, communication systems, etc). Continuous Petri nets (in which the markings are real numbers and the transition firings are continuous processes) were defined more recently; such a PN may model a continuous system (biological systems, fluid systems, etc). or approximate a discrete system (manufacturing systems, transport systems, etc). A paper presents the main ideas origin of the definition of this model. Different timings can be associated with discrete transitions leading to different derived models. When the maximal firing speeds associated with continuous transitions hybrid Petri net is obtained if one of its parts is discrete and the other part is continuous. This are constant, an elegant model is obtained allowing fast simulations.

### 1 INTRODUCTION

Carl Adam Petri is a contemporary German mathematician (C. A. Petri 1962). In the early sixties, he defined a general purpose mathematical model for describing relations existing between conditions and events. Then, he continued working on the net theory. Many other researchers have provided a large number of theoretical results concerning these nets. Engineers have appropriated this model, usually called Petri net, for modeling the discrete event dynamic systems they have to handle, particularly for computer science and automatic control applications. Various extensions have been proposed and are listed in the references. Some of them concern the discrete behavior autonomous or timed (M. Ajmone-Marsan et al), (B. Berthomieu and M. Diaz 1991), (C. Ramchandani 1973). Others address new important application domains such as transport systems (A. Di Febbraro and N. Sacco 2003) or biological systems (A. Doi, et al.) and (A. Saito et al.).

Many systems are naturally hybrid, *i.e.*, their modeling needs at least one continuous state variable and at least one discrete state variable. In some cases, a discrete system, or part of a system, can be approximated by a continuous model.

Petri nets (PNs) are widely used to model discrete event dynamic systems (computer systems, manufacturing systems, communication systems, etc). Continuous Petri nets (in which the markings are real numbers and the transition firings are continuous processes) were defined more recently; such a PN may model a continuous system or approximate a discrete system. A hybrid Petri net can be obtained if one of its parts is discrete and the other part is continuous. This paper is basically a brief survey of the work of the author's team on hybrid PNs.

In a discrete PN, the marking of a place may correspond either to the Boolean state of a device (for example a resource is available or not), or to an integer (for example the number of parts in a buffer). A general analysis method is to compute the set of reachable states and deduce the different properties of the system. But when a PN contains a large number of tokens, the number of reachable states explodes and this is a practical limitation of the use of Petri nets. To illustrate this point, consider a manufacturing line

shown in Figure 1 and composed of three machines  $M_1$ ,  $M_2$  and  $M_3$  in order, and two intermediate buffers  $B_1$  and  $B_2$  with respective finite capacities,  $C_1$  and  $C_2$ . Each machine can be operational or down and the parts move on the machines, and wait in the intermediate buffers if required. We assume that there are always raw parts upstream  $M_1$  and available space downstream  $M_3$ . The number of reachable states of this system is  $N = 2^3(C_1 + 1)(C_2 + 1)$ ; then  $N = 1\,352$  for  $C_1 = C_2 = 12$ . For a set composed of 10 machines and 9 buffers each with capacity 12,  $N = 2^{10} \times 13^9$  which is greater than  $10^{13}$  states!

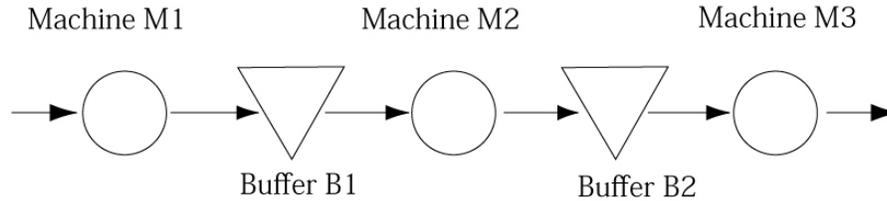


Figure 1: A manufacturing line.

This observation led us to define continuous PNs and hybrid PNs. In a continuous PN, the markings of places are real numbers and the firing of transitions is a continuous process. For the example considered above, the flow of parts on the machine may be approximated by a continuous flow and the numbers of parts in the buffers may be approximated by real numbers. However, the state of each machine (operational or not) is necessarily discrete. Hence, a hybrid model can be used for this system. This paper will present the basic concepts mainly in an intuitive way. For a formal presentation, the reader may refer to the given reference of the book “Discrete, Continuous, and Hybrid Petri Nets” (David and Alla, 2010). Numerous works in the hybrid PNs field have been developed in the literature, some results are listed in the references.

The paper is organized as follows. In Section 2, the basic continuous model known as constant continuous PN is presented. Adding a discrete part leads to Hybrid PN model described in Section 3, and a conclusion is given in Section 4.

## 2 CONTINUOUS PETRI NETS

A continuous PN may be either autonomous (no time is involved) or with firing speeds associated with transitions. A timed model may be used for the performance evaluation of systems. Various timed continuous PN models have been defined which differ by the calculation of the instantaneous firing speeds of the transitions. They provide good approximations for performance evaluation when a PN contains a large number of tokens. All the models work on the same basic rule. The only difference is the way in which the instantaneous firing speeds are defined; it follows that other definitions of this firing speed can be chosen. First, we give hereunder the definition of the basic model called Constant speed Continuous PN (CCPN), then the main ideas will be presented via an example.

**Definition 1** A CCPN is a tuple  $\langle P, T, \text{Pre}, \text{Post}, \mathbf{m}_0, \text{Spe} \rangle$  such that  $P = \{P_1, P_2, \dots, P_n\}$  is a finite, not empty, set of places;  $T = \{T_1, T_2, \dots, T_m\}$  is a finite, not empty, set of transitions;  $P \cap T = \emptyset$ , i.e., the sets  $P$  and  $T$  are disjoint;  $\text{Pre}: P \times T \rightarrow \mathbb{Q}_+$  is the input incidence application;  $\text{Post}: P \times T \rightarrow \mathbb{Q}_+$  is the output incidence application;  $\mathbf{m}_0: P \rightarrow \mathbb{R}_+$  is the initial marking.  $\text{Spe}$  is a function from the set  $T$  of transitions to  $\mathbb{Q}_+ \cup \{\infty\}$ . For  $T_j$ ,  $\text{Spe}(T_j) = V_j$  is maximal speed associated with transition  $T_j$ . □

Let us consider a simple production system composed of two machines and two buffers (Figure 2.a). Each part is processed first by Machine M1, then by Machine M2 with respectively 1/2 and 1/3 times units (t.u.) as processing times. The parts are carried by pallets recycled at each end of production. We

suppose that initially, there are two pallets in the entrance buffer of Machine M2. The timed PN model of this system is given in Figure 2.b. Performances can be deduced from the simulation of this model.

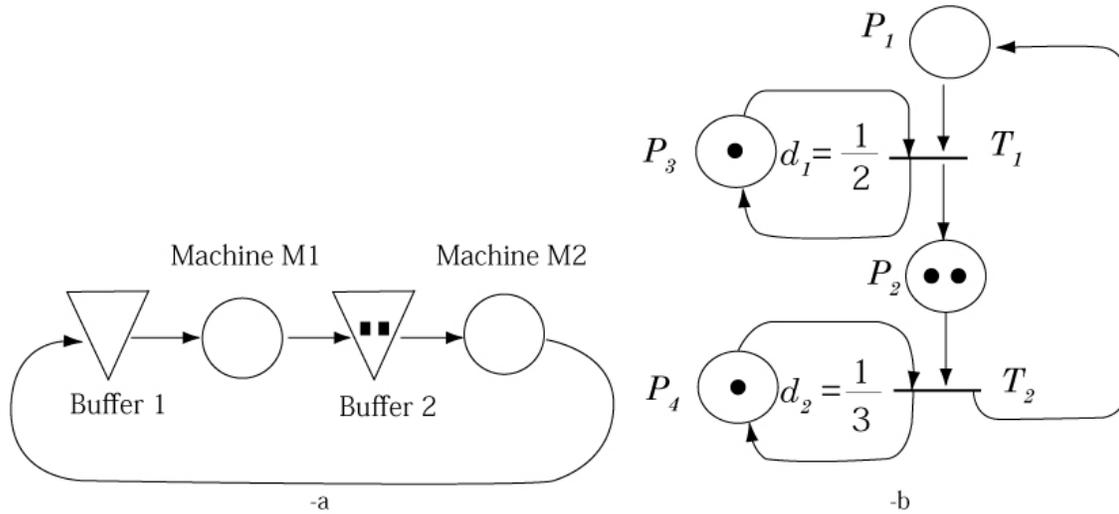


Figure 2: a- A simple production system; -b- Timed PN model.

Now, let us suppose that instead of two pallets, the system contains 75 pallets (Figure 3.a). Then it seems obvious that the simulation of such system will become longer. Another alternative is to consider the number of pallets not as integers but as real numbers, and then the dynamic behavior of the parts will be approximated as a flow. We then obtain naturally a continuous model given in Figure 3.b. where the continuous places are represented with double circles and the continuous transitions with rectangles. With each transition is associated a maximal firing speed  $V_i$  equal to the inverse of durations  $d_i$ . The markings of the continuous places are real numbers and then place  $P_2$  is marked with 75.

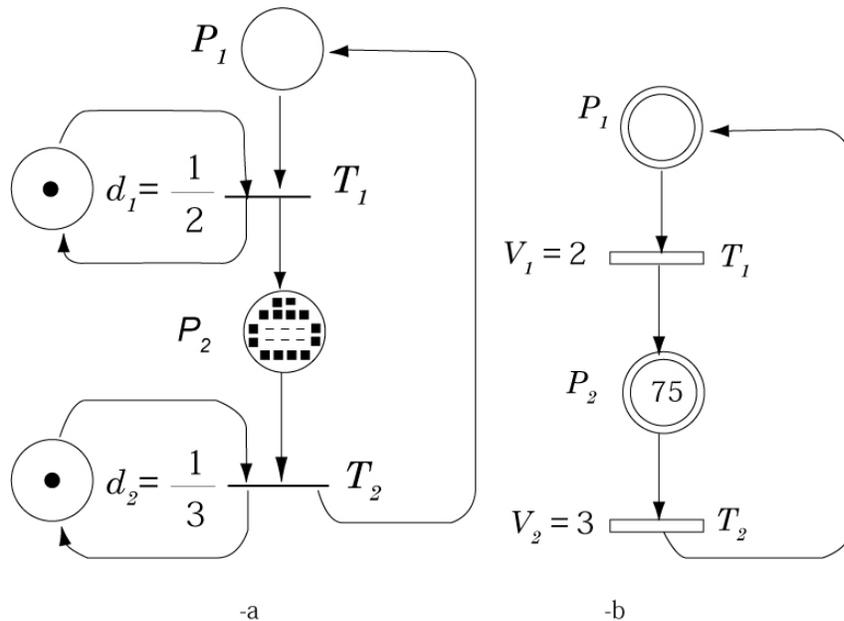


Figure 3: a- Timed PN with 75 pallets; -b- Corresponding CCPN.

Let us study the dynamic behavior of this system. As  $P_2$  is marked, transition  $T_2$  is fired with its maximal speed  $V_2$ . The instantaneous firing speed is equal to its maximal value, *i.e.* Machine M1 has enough parts in its entrance buffer and works at its maximal speed (marking of place  $P_2$  is positive). Let us consider now buffer 1, it is fed with a flow equal to 3 and its maximal speed is 2, then the instantaneous firing speed of transition  $T_1$  is  $V_1 = 2$ .

It follows that for each marking  $m_1$  and  $m_2$  of places  $P_2$  and  $P_2$ :

$$dm_1/dt = (3 - 2), \text{ and then } m_1(t) = t \tag{1}$$

$$dm_2/dt = (2 - 3), \text{ and then } m_2(t) = 75 - t \tag{2}$$

Relations (1) and (2) remain true as long as  $m_2 > 0$ .

At time  $t = 75$ ,  $m_2 = 0$  and  $m_1 = 75$ . Transition  $T_1$  can still be fired at its maximal speed since  $m_1 > 0$ , but  $T_2$  cannot. As a matter of fact  $m_2 = 0$  (buffer 2 is empty). However place  $P_2$  is fed at speed  $V_1$  by firing of  $T_1$  (machine M1 feeding). Then transition  $T_2$  can be fired at speed  $v_2(t) = 2$  which is no longer the maximal speed. In terms of production systems, it is the slowest machine which imposes its speed to the whole system.

Maximal speeds are noted with a capital  $V$ , and instantaneous speed with a small  $v$ . Then,

$$v_1(t) = V_1 = 2 \quad \text{and} \quad v_2(t) = V_2 = 3 \quad \text{for } 0 \leq t < 75 \tag{3}$$

$$v_1(t) = V_1 = 2 \quad \text{and} \quad v_2(t) = V_1 = 2 \quad \text{for } t \geq 75 \tag{4}$$

The corresponding markings are illustrated in Figure 4.

For  $t \geq 75$ , transition  $T_1$  is said to be **strongly enabled** (all the input places are marked), and transition  $T_2$  is said to be **weakly enabled** (all the input places which are empty are fed by firing of other transitions). We have here an original characteristic of a CCPN : *a place with a null marking enables its output transition*.

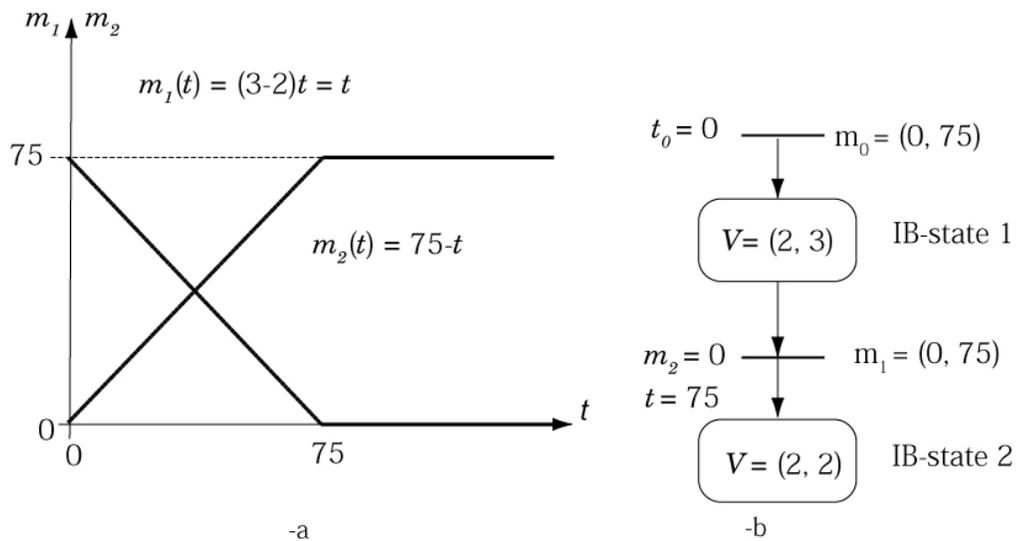


Figure 4: a- Marking evolution of the CCPN in Figure3.b; b- Evolution graph.

It can be noticed that the above constant CCPN has only one event to consider in the simulation, *i.e.* the instant when the marking of place  $P_2$  becomes null.

The dynamic behavior of a CCPN can also be described by an evolution graph as shown in Figure 4.b. In this graph a node is called an invariant behavior states (IB-state). A speed vector is associated with each node which is constant over a certain time period since the evolutions of the markings are linear time functions. The description of an evolution graph will be described in detail for a hybrid PN. Formal algorithms for the computation of instantaneous firing speeds and of evolution graph are presented in detail in the book (David and Alla 2010). They cannot be presented here especially the case of conflicts structure.

The reduced number of nodes is the most important property of this model. The number of nodes in this example is 2, it remains the same what may the number pallets be. Algorithms for computing the instantaneous firing speeds and evolution graphs can be found in the book (David and Alla 2010). In general case, it is a complex problem because the conflict resolution is hard to solve in general configurations. The problem has been solved in case of specific policies such as priority or sharing. Taking into account conflicts has made the algorithms very complex, however in our knowledge it is the first time that conflicts are deeply studied.

### 3 HYBRID PETRI NET

In order to take into account logical conditions in a continuous PN, Hybrid PNs were defined. They combine a continuous part with a discrete one. The basic hybrid PN associates a continuous PN with a discrete PN. The discrete model is a timed PN where fixed real values are associated with transitions. The formal definition of a hybrid PN is given below.

**Definition 2** A hybrid Petri net is a tuple  $\langle P, T, h, \text{Pre}, \text{Post}, \mathbf{m}_0, \text{Tempo} \rangle$  fulfilling the following conditions:  $P = \{P_1, P_2, \dots, P_n\}$  is a finite, not empty, set of places;  $T = \{T_1, T_2, \dots, T_m\}$  is a finite, not empty, set of transitions;  $P \cap T = \emptyset$ , *i.e.* the sets  $P$  and  $T$  are disjointed;  $h: P \cup T \rightarrow \{D, C\}$ , called "hybrid function", indicates for every node whether it is a discrete node (sets  $P^D$  and  $T^D$ ) or a continuous one (sets  $P^C$  and  $T^C$ );  $\text{Pre}: P \times T \rightarrow Q_+$  or  $\mathcal{N}$  is the input incidence application;  $\text{Post}: P \times T \rightarrow Q_+$  or  $\mathcal{N}$  is the output incidence application;  $\mathbf{m}_0: P \rightarrow R_+$  or  $\mathcal{N}$  is the initial marking. In the definitions of  $\text{Pre}$ ,  $\text{Post}$ , and  $\mathbf{m}_0$ ,  $\mathcal{N}$  corresponds to the case where  $P_i \in P^D$ , and  $Q_+$  or  $R_+$  corresponds to the case where  $P_i \in P^C$ .  $\text{Tempo}$  is a function from the set  $T$  of transitions to  $Q_+$ : if  $T_j \in T^D$ ,  $d_j = \text{tempo}(T_j) = \text{timing}$  associated with  $T_j$ ; if  $T_j \in T^C$ ,  $V_j = \frac{1}{\text{tempo}(T_j)} = \text{Maximal firing speed}$  associated with  $T_j$ .  $\text{Pre}$  and  $\text{Post}$  functions must meet the following criterion: if  $P_i$  and  $T_j$  are such that  $P_i \in P^D$  and  $T_j \in T^C$ , then  $\text{Pre}(P_i, T_j) = \text{Post}(P_i, T_j)$  must be verified. □

$Q_+$  and  $R_+$  are respectively the sets of positive rational and real numbers

Here again we introduce in an intuitive way the main characteristics of this model. The followings elementary patterns give the basic modeling schemas. In Figure 5.a, the discrete part controls the continuous one. When the machine is working, the production is allowed. By firing transition  $T_3$  it is stopped. Figure 5.b presents the opposite influence: the continuous part controls the discrete one. When the level of the output buffer reaches the value 14.8, transition  $T_4$  is fired and the production is stopped. We can have more complex behaviors such as the transformation of a discrete marking into a continuous one and vice-versa.

Let us take again the example of the simple manufacturing system, and let us suppose now that the two machines can be stopped for example for maintenance reasons. Machines M1 and M2 work respectively during 90 and 45 t.u. then they are stopped. They can work again after 72 t.u. for M1 and 58 t.u. for M2. This system can be modeled by the hybrid PN given in Figure 6.a.

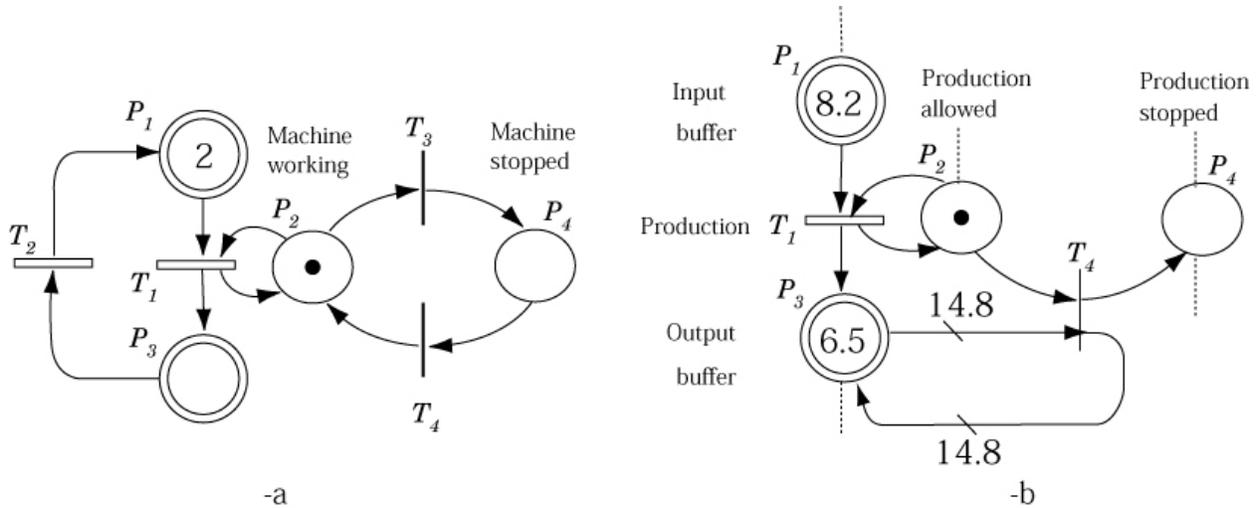


Figure 5: Hybrid PN a- Influence of the discrete part on the continuous part; b- Influence of the continuous part on the discrete part.

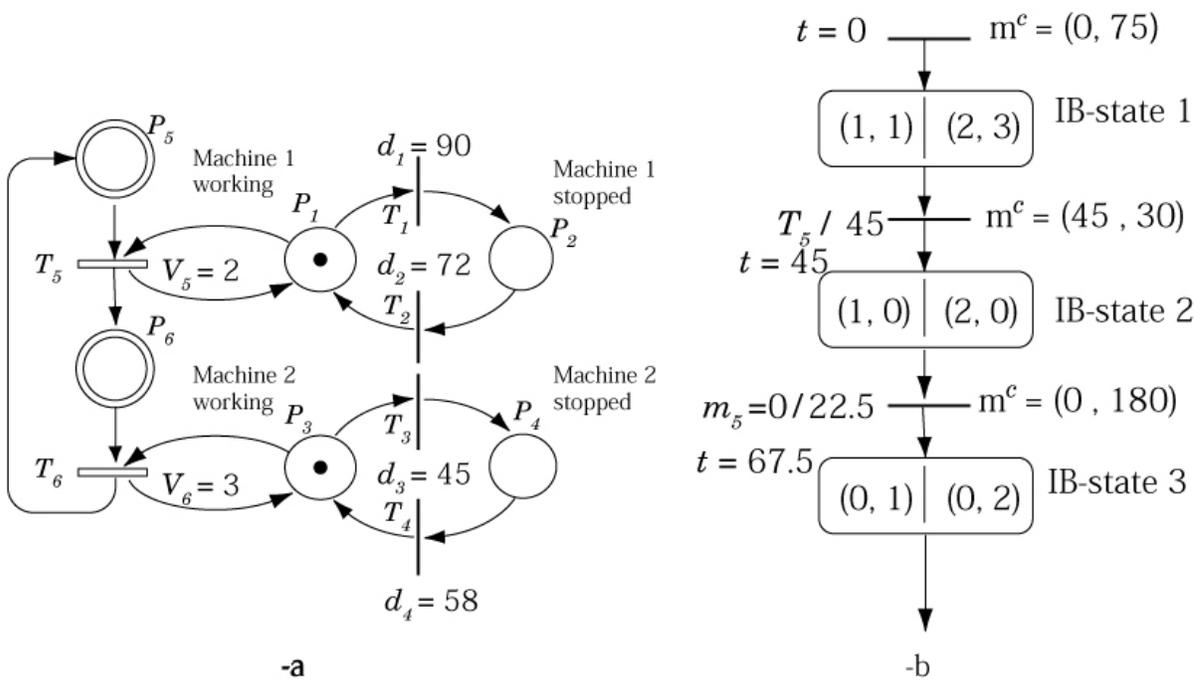


Figure 6: a- Hybrid PN model of the production system; b- Part of the evolution graph.

The dynamic behavior of a hybrid PN is more complex to study since it combines two dynamics. It can also be described by an evolution graph. Figure 6.b shows the beginning part of the evolution graph of the hybrid PN in Figure 6.a. This graph is made of IB-states and transitions among them. An IB-state is such that the marking of the discrete part and the instantaneous speed vector of the continuous part remain constant as long as the system is in the same IB-state. For example, IB-state1 in Figure 6-b corresponds to the behavior from the initial state: the marking of the discrete part is  $m_1 = m_2 = 1$ ; and the instantaneous speed vector is  $v_1 = 2, v_2 = 3$ . *i.e.*, the two machines are working at their maximal speeds. The marking of the continuous part which is written  $m_5 = 0$  and  $m_6 = 75$ , corresponds to the continuous marking when the

IB-state is reached (initial state in our example). The continuous marking evolves continuously and linearly as long as the system is in the same IB-state. For each continuous place, the balance of the marking is defined as the algebraic sum of instantaneous speeds of the transitions feeding the place (i.e., input transitions, with a positive sign), and of the transitions emptying the place (i.e., output transitions, with a negative sign). Hence, this balance, denoted  $B_i$  for place  $P_i$ , corresponds to the time derivative of  $m_i$ .

For the continuous places in Figure.6.a:

$$dm_5/dt = (3 - 2) = 1, \text{ and } dm_6/dt = (2 - 3) = -1 \quad (5)$$

Accordingly, the marking  $m_6$  decreases and will become 0 at time  $t = 75$ . However the discrete markings enable transitions  $T_1$  and  $T_3$ . The first firing occurrence is the one of transition  $T_3$  at time  $t = 45$ . This firing corresponds to from IB-state 1 to IB-state 2 in Figure 6.b. Only the continuous transition  $T_5$  remains enabled and its instantaneous firing speed is  $v_5 = 2$ . Then the next event is the marking of place  $v_5$  which becomes null at time  $t = 67,5$  ( $45 + 22,5$ ) and so on. We have highlighted here the main events which pilot the dynamic of the hybrid PN: the firing of a discrete transition and the marking of a continuous place becoming null. Complete simulations algorithms can be found in the references.

#### 4 CONCLUSION

In this paper, we have presented the basic ideas describing the continuous and hybrid models. This has been done mainly in an intuitive way in order to make accessible these modeling tools to the simulation community. A complete presentation with case studies can be found in the book (David and Alla, 2010).

Continuous systems together with their modeling, analysis and control have long provided research with subject matters. Modeling, analysis and control of discrete systems have undergone major developments in recent decades. In recent years the need has emerged to consider systems which are partially continuous and partially discrete, in other words hybrids systems. To model such systems we could use certain modeling tools normally used for discrete systems. Petri nets are known to be powerful tools for modeling and analysis of discrete systems. The continuous and hybrid Petri nets allow modeling and analysis of continuous and hybrid systems on the same conceptual basis.

#### REFERENCES

- Ajmone-Marsan, M., G. Balbo, A. Bobbio, G. Chiola, G. Conte, and A. Cumati. 1989. "The Effect of Execution Policies on the Semantics and Analysis of Stochastic Petri Nets for the Performance of Multiprocessor Systems", *IEEE Trans. on Software Engineering*, vol. 15, no. 7, pp. 832–846.
- Balduzzi, F., A. Giua, and G. Menga. 1998. "Hybrid Stochastic Petri Nets: Firing Speed Computation and FMS Modelling", *WODES '98*, Gagliari (IT).
- Berthomieu, B., and M. Diaz. 1991. "Modeling and Verification of Time Dependent Systems Using Time Petri Nets", *IEEE Trans. on Software Engineering*, vol. 17, no. 3, pp. 259–273.
- Ciardo, G., D. Nicol, and K. S. Trivedi. 1997. "Discrete-event Simulation of Fluid Stochastic Petri Nets", *Petri Nets & Performance Models (PNPM '97)*, Saint Malo (FR), pp. 217–225.
- David, R., and H. Alla. 2010. *Continuous and Hybrid Petri Nets*. 2<sup>nd</sup> edition. Springer, Heidelberg, Germany: <http://www.springer.com/engineering/robotics/book/978-3-642-10668-2>.
- Demongodin, I., and N. T. Koussoulas. 1998. "Differential Petri Nets: Representing Continuous Systems in a Discrete Event World", *IEEE Transactions on Automatic Control*. Vol. 38, no. 4, pp. 573–579.
- Di Febbraro, A., and N. Sacco. 2003 "Experimental Validation of a Hybrid Petri-Net Based Model of Urban Transportation Networks", *IFAC Conf. on Analysis and Design of Hybrid Systems (ADHS 03)*, Saint-Malo (FR), pp. 111–116.
- Doi, A., S. Fujita, H. Matsuno, M. Nagasaki, and S. Miyano. 2004. "Constructing Biological Pathway Models with Hybrid Functional Petri Nets", *In Silico Biology* 4.

- Petri, C. A., 1962. *Communication with Automata*, Supplement 1 to Technical Report RADC-TR-65-337, N.Y., 1965. Translation by C.F. Greene of *Kommunikation mit Automaten*, PhD Dissertation, Univ. of Bonn.
- Ramchandani, C., 1973, *Analysis of Asynchronous Concurrent Systems by Timed Petri Nets*. Ph.D., MIT (US).
- Recalde, L., S. Haddad, and M. Silva, 2007. "Continuous Petri Nets: Expressive Power and Decidability Issues", in Proc. of the 5th Int. Symp. on Automated Technology for Verification and Analysis, Tokyo, vol. 4762 of LNCS, Springer, pp. 302-377.
- Saito, A., M. Nagasaki, A. Doi, K. Ueno and Satoru Miyano, 2006. "Cell Fate Simulation Model of Gustatory Neurons with MicroRNAs Double-Negative Feedback Loop by Hybrid Functional Petri Net with Extension", *Genome Informatics* 17(1): 100–111.
- Silva, M., and L. Recalde, "Continuization of Timed Petri Nets: From Performance Evaluation to Observation and Control", Proc. of 26th Int. Conf. on Application and Theory of Petri and Other Models of Concurrency, Springer-Verlag, 2005.

**HASSANE ALLA** is Professor at the University Joseph Fourier of Grenoble. His research is mainly concerned with tools derived from Petri nets used for the performance evaluation and for the control synthesis of discrete event systems. He is author or co-author of one hundred and twenty publications. One of its main publications is a book on Continuous and Hybrid Petri nets which has been published in English and in French. His email address is: [hassane.alla@gipsa-lab.fr](mailto:hassane.alla@gipsa-lab.fr)

**LATÉFA GHOMRI** is an assistant professor in the electrical engineering department of the University of Tlemcen (Algeria). She is member of the manufacturing engineering laboratory of Tlemcen. Her research interests are modeling, simulation and control of dynamic systems. His email address is: [ghomrilatefa@yahoo.fr](mailto:ghomrilatefa@yahoo.fr)