HYBRID STOCHASTIC PETRI NETS WITH MATRIX ATTRIBUTES FOR MODELING OF DISCRETE-CONTINUOUS PROCESS

Emil Guţuleac, dr. hab, prof. univ., Sergiu Zaporojan, dr., conf. univ., Ion Gîrleanu, drd., Viorel Cărbune, drd.
Technical University of Moldova

INTRODUCTION

Hybrid systems (*HS*) are a class of systems which incorporates discrete-continuous process, such that the discrete dynamics and continuous dynamics are intertwined with each other. They arise in numerous important applications in CAD, real-time computing, computer networks, safety analysis, robotics and automation, flexible manufacturing systems, transport systems, fault tolerant control systems, mechatronics process control, biological systems, fluid systems, etc., and have recently been at the center of intense research activity in the computer systems and networks, control theory, computer-aided verification and artificial intelligence communities. Thus, *HS* have received increasing attention in the last few years, due to the ubiquitous trend of employing digital controllers in traditionally analogous environments, for example, manufacturing systems. For various applications and modeling of *HS* we refer to [1].

Discrete-continuous modelling and simulation is concerned with the description, analysis and performance evaluation of the dynamic behaviour of *HS* [3, 6, 7]. This approach is a research area that becomes more and more interesting and is due to the fact that most systems of real world applications are not purely discrete nor purely continuous and often both parts influence each other.

In the past several years, methodologies have been developed to model *HS* with stochastic behavior, to analyse their dynamic properties and asses their performance specifications [4, 5].

The generalized stochastic Petri nets (*GSPN*) provide a convenient and concise formalism for describing the discrete event dynamics of *HS* (computer systems, manufacturing systems, communication systems, biological systems, etc) [3, 7]. However, the underlying state space of *GSPN* models tends to be extremely large in practical modeling applications, often forcing us to seek approximate solution methods [4]. An alternative modeling paradigm for the purpose of analysis and simulation of *HS* is based on stochastic fluid models (*SFM*). The *SFM* paradigm allows the aggregation of multiple events into a single event associated with a “significant change” in the system dynamics. This offers the possibility of integrating, in a natural way, continuous and discrete dynamics in a single model.

Among the most *SFM* popular formalisms that are used for modelling of *HS*, there are the timed hybrid Petri nets (*THPN*) [4], fluid stochastic Petri nets (*FSPN*) [2, 6, 8] and hybrid stochastic Petri nets (*HSPN*) [5]. In such models the some places may hold a discrete number of tokens while others contain a continuous quantity represented by real quantities. However, for real *HS* visually modelling and simulation, it is possible that some attributes of these systems should take specific multiple different values; that cannot be easily described in *HSPN* or *FSPN* since their modelling will significantly increase graphical complexity of the system model. For example, in order to evaluate the performance measures of some hybrid systems processes for a specific simulation task considering thousands of services with different values, a high number of places, transitions and arcs will be needed in *HSPN* model in order to be able to obtain desired load value for each specified time interval. This brings a considerable higher structural complexity of this type model, so it is difficult to analyse such a complex structure, for example the amount of states introduces a complexity in global computing and because of that we have a longer simulation time. However, it should also to enhance this formalism in order to be able to fully represent, more concise and flexible describe *HS* systems with complex discrete-continuous stochastic process.

In order to address such issues, we introduce the model definition, behavior rules and the graphical representation for a new kind of *HSPN* formalism with matrix attributes, called bellow as *HSMN*, similarly as they were used in [6] through introducing *database arcs* with matrix weight, that makes possible the use of real data in the simulation process, assuring the validity of the obtained results.. This extension allows the modeling of high complexity systems without the danger of having a very graphically complicated *HSPN* model that is too difficult to represent and hard to understand. In the same context, we consider some examples to graphically represent *HSMN* and them unfolding with *HSPN* models that whose behaviours are equivalents.

An important advantage of proposed approach is the fact that *HSPN* model representation is very concise and flexible, because majority of its attributes are parameterized and can take various marking-dependent values.

1. HSPN WITH MATRIX ATTRIBUTES

1.1. Formal definition

Let the IN_+ and IR are the sets of non-negative natural and real numbers, respectively.

The definition of a *HSPN* with matrix attributes, called *HSMN*, is derived from [3, 5, 6, 8] and it inherits most of the features of *GSPN*, *THPN* and *FSPN*. In a *HSMN* net the matrix attributes of objects (arcs, place capacities, transition guard and priority functions, transition firing rates, etc.) type *z*, depending on current network state *s*, are defined by a set of matrix $\boldsymbol{A}^z = [a^z_{i\,j}(s)]_{k\times n}$, $\boldsymbol{A}^z \in \mathrm{A}$. The value of elements $a^z_{i\,j}(s)$ are constants, variable or functions of specified type, eventually they can be depending on current network state *s* of a *HSMN*.

The dimension $k\times n$ and the localisation of current element $a^z_{i\,j}(s)$ of matrix $\boldsymbol{A}^z$ is specified by a discrete control place set $P^z_A \subset P_D$. For example, for specification $\boldsymbol{A}^z$ and current computing of its element it should be a control place p_l set $P^z_A = \{p_l, p_v\}$. So, the current number of tokens $i = m_l = M(p_l)$ and $j = m_v = M(p_v)$ of control places p_l and p_v respectively shows the element's position in the $\boldsymbol{A}^z$ matrix, and its values needs to be imported and taken in consideration when executing and analysing the model. Moreover, the capacity of control place $p_l \in P^z_A$ and place $p_v \in P^z_A$ should be specified to $K^p(p_l) = k$ and $K^p(p_v) = n$, respectively.

Formally, a *HSMN* is specified as a 14-tuple $H\Gamma$ = < *P*, *T*, *Pre*, *Post*, *Test*, *Inh*, K_p , K_b, *G*, *Pri*, M_0, Λ, *W*, *V* >, where:

- *P* is the finite set of places consisting of a set of discrete places P_D and a set of continuous places P_C , $P=P_D\cup P_C$, $P_D \cap P_C = \varnothing$. A discrete place p_i is drawn with a single circle and can contain a number of tokens, $m_i = M(p_i) \in IN_+$, non-negative integer values. A continuous place (buffer) b_k is drawn with two concentric circles and can contain a real number of fluid $x_k = x(b_k) \in IR$. The marking (the *state*) s=(M, $\boldsymbol{x}$) of the $H\Gamma$ is given by pair of vector-columns, M and $\boldsymbol{x}$, describing the contents value of each type place, (M, $\boldsymbol{x}$) $\in \hat{S} = IN_+^{|P_D|} \times IR^{|P_C|}$, respectively. We call $\hat{S}$ the "potential state space", as opposed to the "actual state space" $S \subseteq \hat{S}$, the set of marking actually reachable during the evolution of the $H\Gamma$. The current marking $s = (M,\ \boldsymbol{x}) \in S$ evolve in time, which we indicate by τ , so, formally, it is a stochastic process $\{(M(\tau), \boldsymbol{x}(\tau),\ \tau \ge 0)$.
- *T* is a finite set of transitions, $T \cap P = \varnothing$, that is partitioned into a set T_D of discrete timed transitions and a set T_C of continuous timed transitions, that $T = T_D \cup T_C$, $T_D \cap T_C = \varnothing$. A continuous timed transition $u_k \in T_C$ is drawn as an empty rectangle. The set of discrete transitions T_D is partitioned into $T_D = T_0 \cup T_\tau$, $T_0 \cap T_\tau = \varnothing$ so that: T_τ is a set of timed discrete transitions and T_0 is a set of immediate discrete transitions.
- *Pre*, *Test* and $Inh\colon P\times T\times \hat{S}\times IN_+^{|P_A|} \to Bag(P)$ respectively are a backward flow, test and inhibition functions incidence mappings. $Bag(P)$ is a discrete or continuous multiset over *P*. The forward flow function incidence mappings in the multisets of *P* is a $Post: T\times P\times \hat{S}\times IN_+^{|P_A|} \to Bag(P)$ describe the set of arcs *A* with the marking-dependent cardinality, connecting transitions with places and vice-versa.
- $K_p : P_D \times IN_+^{|P_D|} \to IN_+ \cup \{\infty\}$ describe the capacity bound K_{p_k} on each discrete place $p_i \in P_D$, $0 \le K^{\min}_{p_i} \le M(p_i) \le K^{\max}_{p_i} < +\infty$, which can contain an *integer* number of *tokens*, respectively. By default, the $K^{\min}_{p_i} = 0$ and $K^{\max}_{p_i}$ is set to infinity.
- The $K_b : P_C \times IR^{|P_C|} \times IN_+^{|P_A|} \to IR$ describe the fluid bound on each continuous place $b_k \in P_C$, such that $-\infty < x^{\min}_k \le x(b_k) < x^{\max}_k < +\infty$, where the $x^{\min}_k$ describe the lower fluid bound and x_i^{max} upper bounds of of b_k. By default the $\forall\ x^{\max}_k$ is set to infinity, and it no effect. An implicit lower bound of continuous place is 0.
- $G : T\times \hat{S}\times IN_+^{|P_A|} \to \{True,\ False\}$ describe the marking-dependent *guard function* of each transition. For $t_j \in T$ a guard function $g_j(s)$ will be

evaluated in each marking s, and if it is *true* (the default value is *true*), the transition may be enabled, denoted $t_j \in T(s)$, otherwise t_j is disabled.

- $Pri: T \times \hat{S} \times IN_+^{|P_A|} \to IN_+$ defines the dynamic priority function for the firing of each transition. The firing of a transition with higher priority potentially disables all the transitions with the lower priority. By default, the $Pri(T_0) > Pri(T_\tau)$.

Figure 1 summarizes the representation of all the $H\Gamma$ graphical primitives.

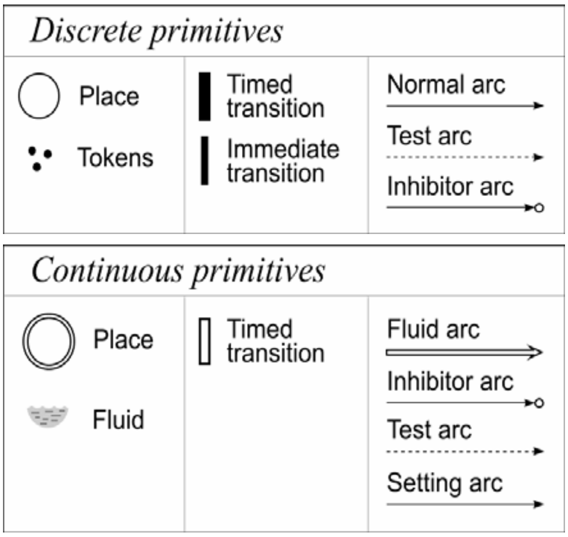

Figure 1. All the graphical primitive of the $H\Gamma$.

- The current marking (*state*) $s=(M, \boldsymbol{x}) \in S$ value of a $H\Gamma$ net depends on the kind of place. The $m_i = M(p_i)$ describes the number of tokens in discrete place p_i, and it is represented by black dots. The $x_k = \boldsymbol{x}(b_k)$ describes the fluid level in continuous place b_k and it is a real number, also allowed to take negative real value. The initial marking of net is $s_0 = (M_0, \boldsymbol{x}_0)$. Graphically, the initial marking is represented by writing the value of m_i^0, or x_k^0, inside the corresponding place. If the number m_i^0 is small it is common to drawn m_i^0 tokens inside the place p_i, represented by dots. A missing value indicates zero.

- A timed discrete transition $t \in T_\tau$ is drawn as a black rectangle and has an exponentially distributed firing time which marking - dependent firing rate $\Lambda : T_\tau \times \hat{S} \times IN_+^{|P^A|} \to IR_+$.

- $W : T_i \times Bag(P) \times IN_+^{|P^A|} \to IR_+$ is the weight function of immediate discrete transitions $t_k \in T_0$, and this type of transition is drawn with a black thin bar and has a zero firing time.

- $V : T_C \times \hat{S} \times IN_+^{|P^A|} \to IR_+$ is the marking dependent fluid rate function of timed continuous transitions T_c. These rates appear as labels next to the continuous timed transitions. If $u_j \in T_c$ is enabled in *tangible* marking M it fires with rate $V_j(M)$, that continuously change the fluid level of continuous place P_C. ■

Figure 2 summarizes the all possible ways placing of arcs in a $H\Gamma$ net for discrete transition and continuous transition with the discrete places and continuous places, respectively.

Given a transition $t_j \in T$, we denote by ${}^\bullet t_j$ and $t_j^\bullet$ the directed preset places and postset places and by ${}^\circ t_j$ and ${}^* t_j$ the *inhibition* set places and *test* set places connected respectively with transition t_j.

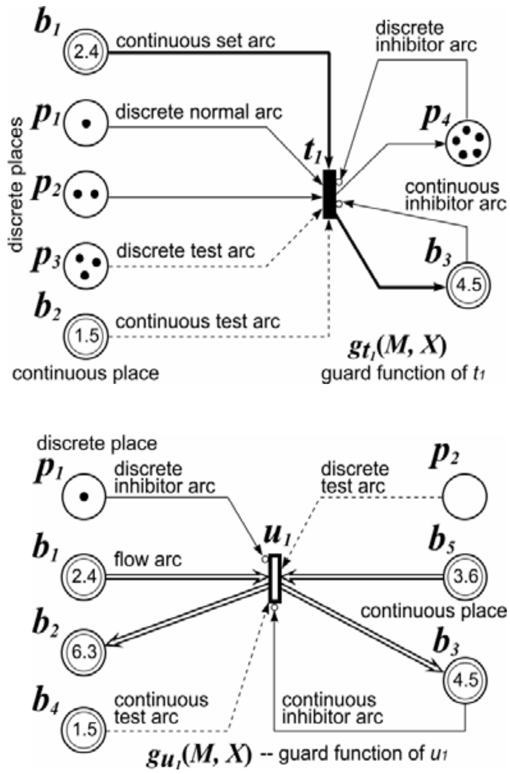

Figure 2. All kinds of arcs and their possible ways for placing in a $H\Gamma$.

1.2. The HSMN dynamics

The dynamics of the *HSMN* combines both time-driven and event-driven dynamics. We define *macro-events* the events that occur when [3]:

The evolutions of *HSMN* in current marking $s=(M, x)$ are determined by the following rules:

1. Localization of the elements $a_{i,j}^z(s) \in \boldsymbol{A}^z$ for $i = M(p_l)$ and $j = M(p_v)$, $p_l, p_v \in P_A^z$ of type z;

2. Computing the current value of $a^{z}_{i,j}(s)$, obtaining the respective constant values. If $i = M(p_l) = 0$ and/or $j = M(p_v) = 0$ then the cardinality value of a respective attribute type z is given by default;

3. For theses values obtain the enabling set of transitions $T(s) = T_D(s) \cup T_C(s), T_D(s) \cap T_C(s) = \varnothing$;

4. Firing of selected transition $t \in T(s)$ and change the current state: $s[t > s'$.

Enabling and Firing rules. As already described [??], two types firing of enabled transition, called *discrete firing* and *continuous firing*, govern the state evolution of the net.

Let $T(s)$ be the set of enabled transitions in current state $s=(M, x) \in S$.

We say that a *discrete transition* $t_j \in T_D(s)$ is enabled in current state s if the following logic (Boolean) expression (enabling condition $ec_D(t_j)$ of t_j is verified:

$$ec_D(t_j) = (\underset{\forall p_i \in {}^{\bullet}t_j}{\wedge}(m_i \geq Pre(p_i, t_j)) \;\&$$
$$(\underset{\forall p_k \in {}^{\circ}t_j}{\wedge}(m_k < Inh(p_k, t_j)) \;\&$$
$$(\underset{\forall p_l \in {}^{*}t_j}{\wedge}(m_l \geq Test(p_l, t_j)) \;\&$$
$$(\underset{\forall p_n \in t_j^{\bullet}}{\wedge}((K_p - m_n) \geq Post(p_n, t_j)) \;\& \;(\underset{\forall b_i \in {}^{\bullet}t_j}{\wedge}(x_i \geq Pre$$
$$(b_i, t_j)) \;\& \;(\underset{\forall b_k \in {}^{\circ}t_j}{\wedge}(x_k < Inh(b_k, t_j)) \;\&$$
$$(\underset{\forall b_l \in {}^{*}t_j}{\wedge}(x_l \geq Test(b_l, t_j)) \;\&$$
$$(\underset{\forall b_n \in t_j^{\bullet}}{\wedge}((K_b - x_n) \geq Post(x_n, t_j)) \;\& \; g_j(s).$$

The transition $t_j \in T_d(s)$ may fire if no other transition $t_k \in T_d(s)$ with higher priority is enabled, and yielding:

$$M' = M + C(\cdot, t_j), \text{ there}$$

$$C(p, t_j) = Post(p, t_j) - Pre(p, t_j), \; \forall p \in P_D .$$

The stochastic evolution of the *HSMN* in *tangible* marking is governed by a race [2, 3]: the timed discrete transition t with the shortest firing time is the one chosen to fire next. If an immediate discrete transition is enabled in current marking s, it is *vanishing*. Otherwise, the marking is tangible and any timed discrete transition is enabled in it [3, 5]. If several enabled immediate transitions $t_j, t_k \in T_D(s)$ are scheduled to fire at the same time in *vanishing* marking s, the transitions t_k, with the respective weights w_k , fire with probability:

$$q(t_k, s) = w(t_k, s) / \sum_{t_j \in T_0(M)} w(t_j, s) .$$

Also, we say that a continuous transition $u_j \in T_C(s)$ is enabled and continuously fires in current marking s if the following logic expression (the enabling condition $ec_C(u_j)$) is verified:

$$ec_C(u_j) = (\underset{\forall b_i \in {}^{\bullet}u_j}{\wedge}(x_i > 0) \;\& \;(\underset{\forall p_k \in {}^{\circ}u_j}{\wedge}(m_k <$$
$$Inh(p_k, u_j)) \;\& \;(\underset{\forall p_l \in {}^{*}u_j}{\wedge}(m_l \geq Test(p_l, u_j)) \;\&$$
$$(\underset{\forall b_k \in {}^{\circ}u_j}{\wedge}(x_k < Inh(b_k, u_j)) \;\& \; g_j(s) \;\&$$
$$(\underset{\forall b_l \in {}^{*}u_j}{\wedge}(x_l \geq Test(b_l, u_j)) \;\&$$
$$(\underset{\forall b_n \in u_j^{\bullet}}{\wedge}((K_{b_n} - x_n) \geq V_j \cdot Post(x_n, u_j)) ,$$

and no other transitions with higher priority are enabled in current state.

If the state s is tangible, fluid flow could continuously through the flow arcs of enabled continuous transitions into or out of continuous places. As a consequence, if transition t_c is *enabled* in current state it *enabling degree*, for every $b \in {}^{\bullet}u$ and $x(b)>0$, is:

$$Enab(u, s) = \min_{b \in {}^{\bullet}u} \{x(b)/Pre(u, b)\}.$$

Given two time instants τ and τ', the evolution of the fluid level in buffer $b_i \in P_C$ is given as:

$$x(b_i, \tau) = x(b_i, \tau') + \vartheta(b_i, u, \tau, \tau'), \text{ there}$$

$$\vartheta(b_i, u, \tau, \tau') := \sum\nolimits_{u_j \in {}^{\bullet}b_i} Post(b_i, u_j) \cdot \int_{\tau'}^{\tau} \upsilon_{u_j}(\theta) d\theta -$$
$$\sum\nolimits_{u_k \in b_i^{\bullet}} Pre(b_i, u_j) \cdot \int_{\tau'}^{\tau} \upsilon_{u_k}(\theta) d\theta , \upsilon_{u_j}$$

and υ_{u_k} denote the firing speeds of u_j and u_k at time θ respectively.

Upon firing, the discrete (continuous) transition removes a specified number (quantity) of tokens (fluid) for each discrete (fluid) input place, and deposits a specified number (quantity) of tokens (fluid) for each discrete (fluid) output place. The levels of fluid places can change the enabling/disabling of transitions.

2. HSMN EXAMPLES

In the following, we illustrate the power and flexibility representation of proposed *HSMN* formalism with a few examples.

We allow the firing rates and the enabling functions of the timed discrete transitions, the enabling functions and firing speeds of the timed continuous transitions, and arc cardinalities to be dependent on the current states of the $H\Gamma$.

Graphically, a matrix attribute of *HSMN* will be presented in a way that it will contain the matrix name in square brackets. So, for example, a direct arc matrix cardinality [2, 3, 5], denoted by $\xrightarrow{[A]}$, can take values that are contained in a specified matrix **A**. To well understand the meaning of this model type, an example of *HSMN*1 is presented in figure 3 with the following initial state $s_0 = (M_0, x_0)$ with:

$$M_0 = (2,\ 3,\ 0,\ 1,\ 4) = (2p_1 3p_2 p_4 4p_5),$$

$$x_0 = (17.4, 3.25, 4.18) = (17.4b_1, 3.25b_2, 4.18b_3).$$

The mean matrix cardinality values of a discrete arc (t_1, p_3), setting continuous arcs $\{(b_1, t_1)$, $(t_1, b_3)\}$ and fluid arc (u_1, b_2) in *HSMN*1, controlled by $P^A = \{p_1, p_2\}$, are given by following specified matrices:

$$\mathbf{A1} = \begin{bmatrix} 3 & 2m_1 + m_3 & 1 & 1 + m_3 \\ m_4 & 7 & m_2 + 2m_5 & 3 + m_2 \\ 1 & 4 & 8 & m_2 + 4m_3 \end{bmatrix},$$

$$\mathbf{A2} = \mathbf{A3} = \begin{bmatrix} 3.0 & 2x_3 & 3.65 & 0.85 + x_3 \\ 1.75 & 2.35 & 3.25 + x_3 & 3.27 \\ 1.25 & 1.40 & x_3 & 2.15 + m_4 \end{bmatrix},$$

$$\mathbf{A4} = \begin{bmatrix} 0.25 & x_3 & 3.65 & 0.85 \\ 0.75 & 2.35 & 0.32 + x_3 & 3.27 \\ 1.25 & 1.40 & x_2 & 2.15 + x_3 \end{bmatrix},$$

where $m_i = M(p_i), i = 1, 2, 3, 4, 5$ is the number of tokens in discrete place p_i in current state, and $x_k = x(b_k), k = 1, 2, 3$ is the quantity of a fluid level in buffer b_k.

Control place p_1 has the specified capacity $K^p(p_1) = k = 3$, but the capacity of place p_2 is $K^p(p_2) = n = 4$.

For *HSMN*1 in figure 3a the selected $a^z_{i,j}$ element position in matrix $\mathbf{A}^z$, the value must to be imported, is realized by information about the current token number contained in control place set $\{p_1, p_2\}$, that specify the index row $i = m_1$ and index column $j = m_2$. So, for $i = m_1^0 = 2$ and $j = m_2^0 = 3$ we obtain: the cardinality value of arc (t_1, p_3) is equal to $a^1_{2,3} = m_2 + 2m_5 = 10$, of arcs (b_1, t_1) and (t_1, b_3) its are equal to $a^2_{2,3} = a^3_{2,3} = 3.25 + x_3 = 7.43$, respectively, but for arc (u_1, b_2) it is equal to $a^4_{2,3} = 0.32 + x_3 = 4.5$. For these corresponding current values of arc cardinality the enabled set of transitions is $T(s_0) = \{t_1, u_1\}$.

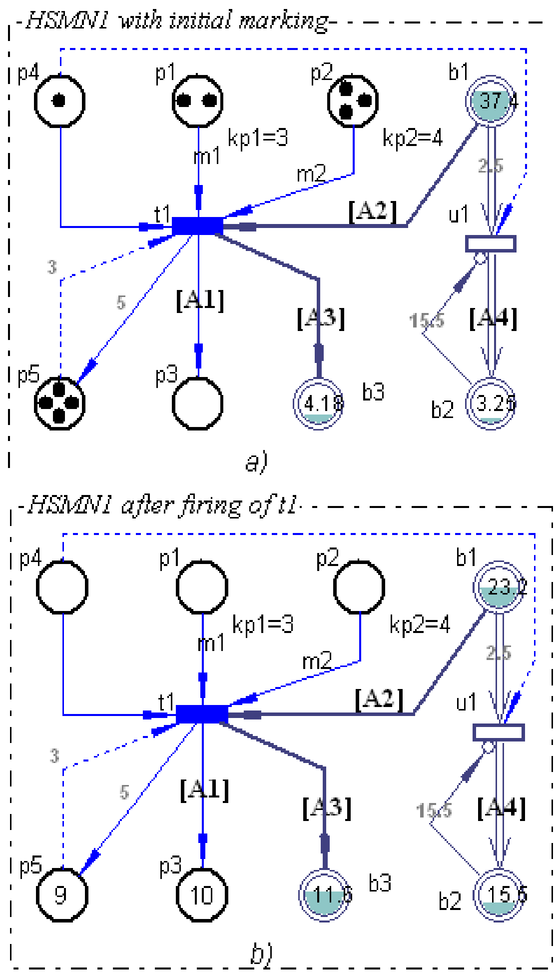

Figure 3. A *HSMN*1 with matrix arc cardinality: a) initial state; b) final state.

Let the firing speed of continuous transition u_1 is υ_1=1 and firing rate of timed transition t_1 is $\lambda_1 = 0.25$, that it mean firing delay is $\tau_{t1} = 4$ t.u.

Because the cardinality of $Inh(b_2, u_1)$=15.50, then the transition u_1 continuously fire only during $\tau_{u_1} = (15.50 - x_2)/a^4_{2,3} = 2.72 < \tau_{t1} = 4$ t.u., so it is disabled and the fluid level x_1' of buffer b_2 become: $x_1' = x_1 - (2.5 \cdot \tau_{u_1} + a^2_{2,3}) = 23.17$.

Figure 3b presents the state of HSMN1 network after t_1 fires, from where we can observe that place p_3 has a number of tokens equal to 9, because the element $a_{3,4} = m_2 + 4m_3 = 8$ was selected. As a result the firing of t_1, we yield the new state $s' = (M', x')$ with:

$$M' = M_0 + a^1_{2,3} = (10p_3, 9p_5),$$

$$x'_2 = x_2 + a^2_{2,3} \cdot \tau_{u_1} = 15.50,\ x'_3 = x_3 + a^3_{2,3} = 11.61.$$

In a similar way all *HSMN* nets attributes can be parameterized using matrix cardinality, for example, see the following attributes [3]:

- Guard function of transitions $\boldsymbol{G}^{z1}$,
  $\boldsymbol{G}^{z1} = [g^{z1}_{i,j}(s)]_{k\times n}$, $z1 \in \{t, u\}$;
- Place capacity $\mathrm{K}^{z2} = [k^{z2}_{i,j}(s)]_{l\times n}$, $z2 \in \{p, b\}$;
- Firing rate $\boldsymbol{R}^{z3} = [r^{z3}_{i,j}(s)]_{k\times n}$, $z3 \in \{\lambda, w, \upsilon\}$;
- Priority firing of transitions T
  $\Gamma^{z4} = [\gamma^{z4}_{i,j}(s)]_{k\times n}$, $z4 = Pri$;

The most important benefit we get from using *HSMN* when describing and verifying discrete-continuous processes of hybrid systems is that the structure of these models' is very concise and is very flexible when modifying it parameters in real time, because most of its attributes are parameterized. This permits assigning in a current state controlled mode, alternative values of *HSMN* attributes. Also, using proposed approach matrix attributes, real time data can be easily imported in simulating model, and in this way we can ensure results correctness.

Moreover, the modelling the same system through the other kind of Petri net, for example *GSPN* or *THPN*, to drive the structure of this models it is necessity for each matrix attribute to use additional $2k \times n$ arcs and $k \times n$ additional transitions (respective places), and this means that the model's complexity grows significantly. To illustrate the usage advantages of *described approach*, we present a simple example of *HSMN*2 shown in figure 4a, that a matrix cardinality of direct arc (t_5, p_6) is parameterized by matrix **A**:

$$\mathbf{A} = \begin{bmatrix} 5 + m_1 & 3 & 2 \\ 4 & 2 + m_2 & 1 + 2m_2 \end{bmatrix}.$$

In this model control places are $P^{A5} = \{p_l, p_v\}$ with $K^p(p_2) = 2$ and $K^p(p_3) = 3$ respectively. Selection of respective row $i = m_2$ and column $j = m_3$ of element $a^5_{i,j}(M)$, $i = 1, 2$; $j = 1, 2, 3$ from matrix **A** is done in a dynamic way by current marking of places p_2 and p_3, respectively.

The unfolding of the model *HSMN*2 trough *GSPN*2 attributes approach, that whose behaviours are equivalents, is shown in figure 4b.

In order to show it in such mode, we need to substitute in *HSMN*2 attributes as bellow:

1) The transition t_5 is substitute by $k \times n = 6$ transitions $t_{5,l}$, $l = 1, \ldots, k \times n$, that whose guard functions are respectively:

$$g_{5,1}(M) = (m_2 = 2) \,\&\, (m_3 = 3),$$
$$g_{5,2}(M) = (m_2 = 2) \,\&\, (m_3 = 2),$$
$$g_{5,3}(M) = (m_2 = 2) \,\&\, (m_3 = 1),$$
$$g_{5,4}(M) = (m_2 = 1) \,\&\, (m_3 = 2),$$
$$g_{5,5}(M) = (m_2 = 1) \,\&\, (m_3 = 3),$$
$$g_{5,6}(M) = (m_2 = 1) \,\&\, (m_3 = 1);$$

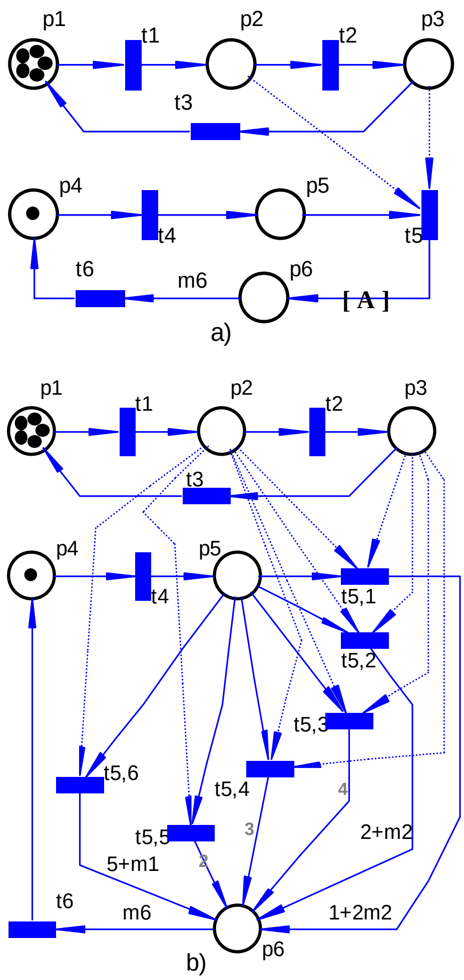

Figure 4. Behavioural equivalent Petri net: *a*) model *HSMN*2 and *b*) model *GSPN*2.

2) The direct matrix arc (t_5, p_6) with matrix cardinality **A** is substitute by $k \times n$ direct arcs $(t_{5,l}, p_6)$, $l = 1, \ldots, 6$ with weight's value from respective **A** matrix's elements;

3) To connect place p_5 with each introduced transition $t_{5,l}$ through arcs $(p_5, t_{5,l})$, $l = 1,...,6$;

4) Place p_2 (respectively p_3) is connected with each introduced transition $t_{5,l}$ through test arcs $(p_2, t_{5,l})$ (respectively $(p_3, t_{5,l})$).

In addition, it is to mention, that the real hybrid system is modelled by a *HSMN* approach, were contain multiple matrix attributes $\mathbf{A}^z$ with different sizes, the behavioural equivalents resulting *HSPN* model is still too complex to be of practical use for conveying the system behaviour visually. For example, the equivalents resulting *HSPN*1 model of *HSMN*1 shown in figure 3, may contain potentially at least $3N_1$ graphical elements (transitions, places and arcs), there:

$$N_1 = (\sum_j^4 (k_j \cdot n_j)) = 3^4 \cdot 4^4 = 20736 \cdot$$

Proposed framework is generic and can be applied to a numerous system types with discrete-continuous process. Additionally, with minor changes and additions, described approach can be generalised for studying domains with similar characteristics. Presented analysis shows that *HSMN*, which were defined and studied in this paper, can be used as a much promising instrument for modelling and evaluating of hybrid system performance indicators.

This work is supported by National Institutional Applied Reserche Project under grants 15.817.02.28A, Republic of Moldova.

3. CONCLUSIONS

In this paper, a new framework *HSMN* was introduced, as a derivative of *GSPN* and *HSPN*. Modelling and performance evaluation of stochastic discrete-continuous process is illustrated.

The *HSMN* approach is very efficient for representing, modelling, verifying and analysing of hybrid system performance, because *HSMN* use has the following advantages: 1) there are additional visualisation features for modelling and simulating procedures, that permits to create a string environment for validation and evaluation; 2) it is possible to visualise in the same model attribute's dynamic change; 3) the real data can be easily imported in simulation process, assuring correctness and validity of obtained results.

The applicability of this approach is illustrated through a few examples of *HSMN* models with different matrix attributes. Moreover, this approach with rather few modifications and additions may be further generalized to study a reconfigurable hybrid system from areas with similar enhanced characteristics.

We aim to elaborate and develop a software product for visual simulation and analysis of *HSMN* models that describe the evolution of hybrid systems with discrete-continuous process.

Bibliography

1. **Bujorianu M.L., Lygeros J.** *Towards modelling of general stochastic hybrid systems. Stochastic Hybrid Systems: Theory and Safety Critical Applications. LNCIS, vol. 337, Springer, Heidelberg, pp. 3–30, 2006.*

2. **Ciardo G., Nicol D.M., Trivedi K.S.** *Discrete-event simulation of fluid stochastic Petri nets, IEEE Transactions on Software Engineering, 2, (25), pp. 207–217, 1999.*

3. **Chiola G., Ajmone- Marsan M., Balbo G., Conte G.** *Generalized stochastic Petri nets: A definition at the net level and its implications. IEEE Transactions on Software Engineering, 19(2), pp. 89-107, 1993.*

4. **David R., Alla H.** *Discrete, Continuous, and Hybrid Petri Nets. Springer-Verlag, Berlin, 2010. - 568 p., DOI 10.1007/978-3-642 -10669-9.*

5. **Guţuleac E.** *Descriptive compositional HSPN modeling of computer systems. Annals of the Craiova University, România, vol. 3 (30), no.2, pp.82-87, 2006.*

6. **Katsigiannis Y. A., Georgilakis P. S. Tsinarakis G. J.** *Introducing a coloured fluid stochastic Petri net-based methodology for reliability and performance evaluation of small isolated power systems including wind turbines. IET Renewable Power Generation, Vol. 2, No. 2, pp. 75–88, 2008.*

7. **Herajy M., Schwarick M., Heiner M**. *Hybrid Petri Nets for modelling the eukaryotic cell cycle. ToPNoC VIII, pp. 123–141, 2013. Doi: 10.1007/ 978-3-642- 40465-8/7.*

8. **Horton G., Kulkarni V. G., Nicol D. M., Trivedi, K. S.** *Fluid Stochastic Petri Nets: Theory, Application, and Solution Techniques. European Journal of Operations Research, 105 (1), pp. 184-201, 1998.*

Recommended for publication: 26.05.2016.