DESCRIPTIVE COMPOSITIONAL HSPN MODELING OF COMPUTER SYSTEMS

Emilian Gutuleac

Technical University of Moldova, Department of Computer Science, 168 Bd. Stefan cel Mare, MD-2004 ,Chisinau, Republic of Moldova, E-mail: egutuleac@mail.utm.md

Abstract: In this paper we define a set of composition operations and descriptive based expressions to construction of composite Hybrid Stochastic Petri Nets (HSPN) for performance discrete-continuous modeling of computer systems. We consider the enhancements of our approach for a performance modeling of multiprocessor system.

Keywords: Descriptive expressions, discrete-continuous modeling, hybrid stochastic Petri nets, performance evaluation.

1. INTRODUCTION

Performance evaluation is concerned with the description, analysis and optimization of the dynamic behavior of computer systems. Such systems, which are described by both discrete and continuous variables, are usually referred to as hybrid systems and are very complex to be modeled and analyzed.

The hybrid systems are characterized by having non-trivial interactions between the discrete and continuous parts of the model; in this case a modeling environment is desirable which allows a unified specification of the complete system to be studied. In this area more main formalisms have been discussed in the literature, namely Hybrid Petri Nets (HPN) (Alla, 1998), Fluid Stochastic Petri Nets (FSPN) (Horton, 1996) and *Stochastic HPN* (HSPN) described by second order partial differential equations (*PDEs*) in the continuous variables representing the fluid levels (Gutuleac, 2002).

Practical methodologies in engineering and computer science take a structural approach, designing systems from smaller subsystems and components, which can be combined and reused.

In (Gutuleac, 2004) is presented a method for the compositional construction of this type models. The approach of a *Descriptive Expressions* (*DE*), presented in this paper for compositional construction of HPN and HSPN models, is intended to solve certain problems of compositionality in its formalisms without loosing the benefits of visual concurrent semantics. The rest of this paper is structured will be the following. In Section II we briefly introduce second order HSPN formalism. We then proceed, in Section III, to present a descriptive compositional approach for the construction of HSPN based models, covering both functional and stochastic behavior. We define a set of compositional operations and descriptive expressions on HSPN based models. This incorporate place and transition fusion. We give a number of examples to illustrate this synthesis of HPN and HSPN models. In Section IV we present one example of descriptive compositional HSPN models applications for performance modeling of pipe-line multiprocessor systems. In the final Section we present the conclusions of this work and forthcoming research efforts.

2. COMPOSITE LABELED HPN

In this section, we define a variant of HPN called composite labeled HPN. Let L be a set of labels $L = L_P \cup L_T$, $L_P \cap L_T = \varnothing$. Each place p_i labeled $l(p_i) \in L_P$ has a local state and transition t_j has action labeled as $l(t_j) \in L_T$.

Definition 1. A *composite labeled* HPN structure is a 11-tuple HG = < *P*, *T*, *Pre*, *Post*, *Test*, *Inh* , K_p , K_b, *G*, *Pri*, *l* >, where: *P* is the finite set of places partitioned into a set of discrete places P_D, and a set of continuous places P_C , $P=P_D \cup P_C$, $P_D \cap P_C = \varnothing$.

The discrete places may contain a natural number of tokens, while the marking of a continuous place is a non-negative real number (fluid level). In the graphical representation, a discrete place is drawn as a single circle while a continuous place is drawn with two concentric circles; T is a finite set of transitions, $T \cap P = \varnothing$, that can be partitioned into a set T_D of discrete transitions and a set T_C of continuous transition, $T = T_D \cup T_C$, $T_D \cap T_C = \varnothing$. A discrete transition $t_j \in T_D$ is drawn as a black bar and continuous transition $t_i \in T_C$ is drawn as a rectangle; *Pre*, *Test* and $Inh\!:\!P \times T \rightarrow Bag(P)$ respectively is a forward flow, test and inhibition functions. $Bag(P)$ is a multiset over P. The $Post\!:\!T \times P \rightarrow Bag(P)$ is a backward flow function in the multi-sets of P, where defined the set of arcs A and describes the marking-dependent cardinality of arcs connecting transitions and places. The set of arcs A is partitioned into five subsets: A_d, A_s, A_h, A_c and A_t.

The subset A_d contains the *discrete normal arcs* which can be seen as a function A_d: $((P_D \times T_D) \cup (T_D \times P_D)) \rightarrow IN_+$ and *continuous set arcs* A_s: $((P_C \times T_D) \cup \{T_D \times P_C)) \rightarrow IR_+$. The arcs A_d and A_s, are drawn as single arrows. The subset A_h contains the discrete *inhibitory* arcs A_h: $(P_D \times T) \rightarrow IN_+$ or continuous *inhibitory* arcs $(P_C \times T) \rightarrow IR_+$. These arcs are drawn with a small circle at the end. The subset A_c defines the continuous *flow* arcs A_c: $((P_C \times T_C) \cup (T_C \times P_C)) \rightarrow IR_+$, these arcs are drawn as double arrows to suggest a pipe. A *test* A_t input arc is directed from a place of any kind to a transition of any kind, A_t: $(P_D \times T) \rightarrow IN_+$ or $(P_C \times T) \rightarrow IR_+$ and are drawn as dotted single arrows. It does not consume the content of the source place. The arc of net is drawn if the cardinality is not identically zero and non-zero cardinality of arcs is labeled next to the arc with a default value being 1. The IN_+ and IR_+ is respectively the set of discrete and real nonnegative numbers; $K_p : P_D \rightarrow IN_+$ is the capacity of discrete places, and by default it is ∞, i.e. being infinite value. The function $K_b : P_C \rightarrow IR_+ \cup \{\infty\}$ describes the fluid lower bound x_i^{min} and upper bounds x_i^{max} on each continuous place. This x_i^{max} by default it is ∞, i.e. being infinite value and bound has no effect when it is set to infinity. Each continuous place has an implicit lower bound at level is 0; The *guard function* $G: T \times Bag(P) \rightarrow \{True, False\}$ is defined for each transition. For $t \in T$ a guard function $g(t, M)$ is a Boolean function that will be evaluated in each marking M, and if it evaluates to *true*, the transition may be enabled, otherwise t is disabled (the default value is *true*); *Pri*: $T_D \rightarrow IN_+$ defines the priority functions for the firing of each transition. The enabling of a transition with higher priority disables all the lower priority transitions; $l: P \cup T \rightarrow L$, is a labeling function that assigns a label to a place and a transition. In this way that maps place (or transition) name into condition (or action) names that $l(p_i) = l(p_n) = b$ but $p_i \neq p_n$ (or $l(t_j) = l(t_k) = \alpha$ but $t_j \neq t_k$). ■

Figure 1a and Figure 1b summarizes the all possible ways of placing arcs in a *HN* net for discrete transition and continuous transition, respectively.

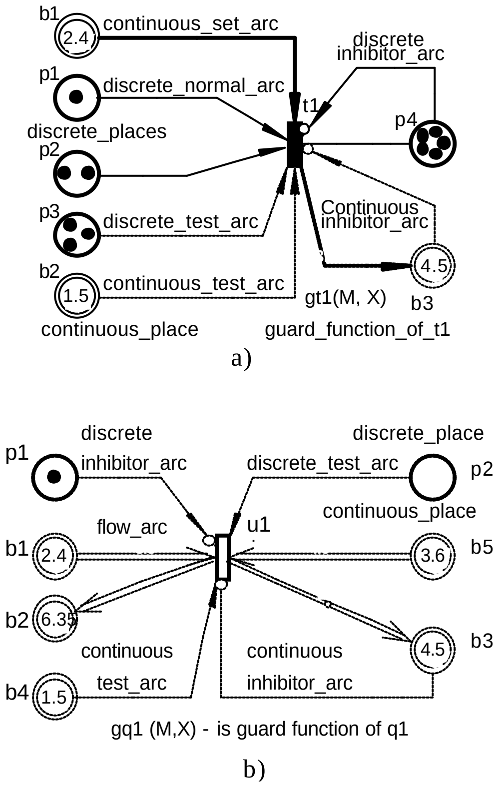

Fig. 1. All the possible ways of placing arcs in a net.

Definition 2. A *marked* composite labeled HPN net is a pair $HN = \langle HG, M_0 \rangle$, where HG is a composite labeled HPN structure (see Def. 1) and $M_0 = (\boldsymbol{m_0}, \boldsymbol{x}_0)$ is the initial marking of the net.

The current marking of net is $M = (\boldsymbol{m}, \boldsymbol{x})$, where $\boldsymbol{m}: P_D \rightarrow IN_+$ and $\boldsymbol{x}: P_C \rightarrow IR_+$. The vector-column $\boldsymbol{m} = (m_i\, p_i,\ m_i \geq 0, \forall\, p_i \in P_D)$ with $m_i\, p_i$ is the number m_i of tokens in discrete place, and it is represented by the black dots. The $\boldsymbol{x} = (x_k b_k,\ x_k \geq x_k^{\min}, \forall\, b_k \in P_C)$ is vector-column, where $x_k b_k$ is the fluid level x_k in continuous place b_k, and it is the real number. ■

Definition 3. A composite labeled *HSPN* net as a 3-tuple, $SHN = \langle HN, q, V \rangle$, where:

HN is a marked composite HPN, (see def. 2), where the set of discrete transitions T_D is partitioned into $T_D = T_\tau \cup T_0$, $T_\tau \cap T_0 = 0$ so that T_t is a set of timed transitions (exponentially distributed firing) and T_0 is a set of immediate transitions (fire in zero time once they are enabled). The $Pri(T_0)>Pri(T_t)$. A timed transition $t_j \in T_t$ is drawn as a black rectangle and has a firing rate is associated to it. An immediate transitions $t_j \in T_i$ is drawn with a black thin bar and has a constant zero firing time. Let $T(M)$ denote the set of enabled transitions in current $M = (\boldsymbol{m}, \boldsymbol{x})$; $q: T_D \times Bag(P) \to IR_+$ is the firing speed function associated with the discrete transitions T_D so that: a) thus a timed transitions $t \in T_\tau(M)$ is enabled in current (tangible) marking M, it fires with rate $\theta(t, M) = \lambda(t, M)$. Note once again, we do allow the firing rate to be dependent on fluid levels; b) thus a immediate transitions $t \in T_0(M)$ is enabled in current (a vanishing) marking M, it fires with probability:

$$q(t, M) = \theta(t, M) / \sum_{t' \in T_0(M)} \theta(t', M);$$

$V: T_C \times Bag(P) \to IR_+$ is the marking-dependent fluid rate function of timed continuous transitions T_C. Thus rates appear as labels next to the continuous timed transitions. If $t_j \in T_C$ is enabled in tangible marking M it fires with fluid rate $V_j(M)$, which is a normally distributed random variable, that is specified by the distributions expectation $v(t,M)$ and the squared coefficients $Kv(t,M)$ of variance $\sigma^2(t,M)=v(t,M) \cdot Kv(t,M)$. ■

Enabling and Firing of Transitions. Given a $t_j \in T$, we denote $^\bullet t_j = \{p_i \in P : \mathrm{Pre}(p_i, t_j) > 0\}$ the input and $t_j^\bullet = \{p_i \in P : Post(p_i, t_j) > 0\}$ output set places of t_j, and with $^\circ t_j = \{p_i \in P : Inh(p_i, t_j) > 0\}$ the inhibition and $^* t_j = \{p_i \in P : Test(p_i, t_j) > 0\}$ the test set places of transition t_j, respectively. Also, in following we denote by $u_k \in T_C$ the continuous transitions and by $b_k \in P_C$ the continuous places can be distinct it between discrete transitions and continuous places, respectively. Let $T(M)$ the set o enabled transitions in current marking M. We say that a discrete transition $t_j \in T_D(M)$ is enabled in current marking M if the following logic expression (enabling condition $ec_d(t_j)$) is verified:

$$ec_d(t_j) = (\bigwedge_{\forall p_i \in {}^\bullet t_j} (m_i \geq \mathrm{Pr}\, e(p_i, t_j)) \;\&$$
$$(\bigwedge_{\forall p_k \in {}^\circ t_j} (m_k < Inh(p_k, t_j)) \;\&$$
$$(\bigwedge_{\forall p_l \in {}^* t_j} (m_l \geq Test(p_l, t_j)) \;\&$$
$$(\bigwedge_{\forall p_n \in t_j^\bullet} ((K_p - m_n) \geq Post(p_n, t_j)) \;\&$$
$$(\bigwedge_{\forall b_i \in {}^\bullet t_j} (x_i \geq \mathrm{Pr}\, e(b_i, t_j)) \;\& \; (\bigwedge_{\forall b_k \in {}^\circ t_j} (x_k <$$
$$Inh(b_k, t_j)) \;\& \; (\bigwedge_{\forall b_l \in {}^* t_j} (x_l \geq Test(b_l, t_j)) \;\&$$
$$(\bigwedge_{\forall b_n \in t_j^\bullet} ((K_b - x_n) \geq Post(x_n, t_j)) \;\& \; g(t_j, M) \;.$$

The transition $t_j \in T_D(M)$ fire if no other transition $t_k \in T_D(M)$ with higher priority has enabled. Also, we say that a continuous transition $u_j \in T_C(M)$ is enabled and continuously fire in current marking M if the following logic expression (the enabling condition $ec_c(u_j)$) is verified:

$$ec_c(u_j) = (\bigwedge_{\forall b_i \in {}^\bullet u_j} (x_i > 0) \;\& \; (\bigwedge_{\forall p_k \in {}^\circ u_j} (m_k <$$
$$Inh(p_k, u_j)) \;\& \; (\bigwedge_{\forall p_l \in {}^* u_j} (m_l \geq Test(p_l, u_j)) \;\&$$
$$(\bigwedge_{\forall b_k \in {}^\circ u_j} (x_k < Inh(b_k, u_j)) \;\& \; g(t_j, M) \;\&$$
$$(\bigwedge_{\forall b_l \in {}^* u_j} (x_l \geq Test(b_l, u_j)) \;\&$$
$$(\bigwedge_{\forall b_n \in u_j^\bullet} ((K_b - x_n) \geq V_j \cdot Post(x_n, u_j))$$

and no transition with higher priority has concession. If an immediate discrete transition has concession in marking $M = (\boldsymbol{m}, \boldsymbol{x})$, it is enabled and the marking is vanishing. Otherwise, the marking is tangible and any timed discrete transition with concession is enabled in it.

An immediate discrete transition t_j enabled in marking $M = (\boldsymbol{m}, \boldsymbol{x})$ yields a new vanishing marking $M' = (\boldsymbol{m}', \boldsymbol{x}$. We can write $(\boldsymbol{m}, \boldsymbol{x})$ $[t_j > (\boldsymbol{m}', \boldsymbol{x})$. If the marking $M = (\boldsymbol{m}, \boldsymbol{x})$ is tangible, fluid could continuously flow through the flow arcs A_c of enabled continuous transitions into or out of fluid places. As a consequence of this, a transition t_c is *enabled* at M iff for every $p_c \in {}^\bullet t_c$, $\boldsymbol{x}(p_c)>0$, *and its enabling degree* is:

$$enab(t_c, M) = \min_{p_c \in {}^\bullet t} \{\boldsymbol{x}(p_c)/Pre(p_c, t_c)\}.$$

In the *HN*, the potential rate $b_i(M)$ of change of fluid level in place $p_i \in P_C$ in marking current M is given by: $b_i(M) = \sum_{t_k \in T(M)} [V_{k,i}(M) - V_{i,k}(M)]$, where, for any given $t_k \in T_C$, $V_{k,i}(M)$ is an input fluid rate of fluid place $p_i \in P_C$ and $V_{i,k}(M)$ is an output fluid rate of this place. We allow the firing rates and the enabling functions of the timed discrete transitions, the firing speeds and enabling functions of the timed continuous transitions, and arc cardinalities to be dependent on the current state of the *HSN*, as defined by the current marking $M(t)$.

The reachability graph of *HSN* are isomorphic to hybrid continuous-time Markov chains (*HCMC*) and this describe the dynamics of an *HSN* can be represented by a system of partial differential equations (PDE) in a probability density of state. In (Gutuleac, 2002) is derived thus type of equations which describe the steady-state behavior of the underlying *HCMC* of a *HSN*.

In the following, if not are mentioned apart, we assume that $Pri(t_j) = 0$ and $g(t_j,M) = 1$ and therefore we can omit $Pri(t_j)$ and $g(t_j,M)$ in the formal definitions of *DE* element of compositional operations and its HSN translations.

3. DESCRIPTIVE EXPRESSIONS OF HPN

Due to the space restrictions we will only give a brief overview to this topic and refer the reader to (Gutuleac, 2004) and the references therein.

In following for abuse of notation, labels and name of transitions/ places are the same.

We introduce the concept of a basic descriptive element (*bDE*) for a basic HPN (*bHN*) as following: $bDE = |_{t_j}^{a_j} m_i^0 p_i\ [W_i^+, W_i^-]\,|_{t_k}^{a_k}$. The translation of this *bHN* is shown in figure 2a, where respectively $t_j = {}^\bullet p_i$ is input transition (action type a_j) and $t_k = p_i^\bullet$ is the output transition (action type a_k) of place $p_i \in P$ with $m_i^0 \in \{m_0(p_i), x_0(b_i)\}$ initial marking, and respectively the flow type relation functions $W_i^+ = \mathrm{Pre}(t_j, p_i)$ and $W_i^- = Post(p_i, t_k)$, which return the multiplicity of input and output arcs of the discrete place $p_i \in P_D$ or the continuous place $p_i = b_i \in P_C$, respectively.

The derivative elements of *bDE* are for $p_i^\bullet = \varnothing,\ W_i^- = 0$ is $|_{t_j}^{a_j}\ m_i^0 p_i\ [W_i]$ with final place p_i of t_j and for ${}^\bullet p_i = \varnothing,\ W_i^+ = 0$ is $m_i^0 p_i\ W_i\,|_{t_k}^{a_k}$ with entry place p_i of t_k (see figure 2b). If the initial marking m_i^0 of place p_i is a zero tokens (or fluid level) we can omit $m_i^0 = 0$ in *bDE*. By default, if the type of action α is not mentioned this to match the name of a transition *t*. From a *bDE* we can build more complex *DE* of PN components by using composition operations.

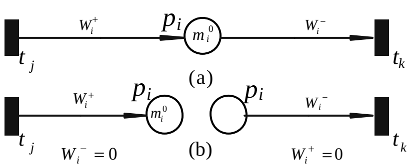

Fig. 2. Translation in *bHN* (a) of *bDE* and (b) its derivatives.

Also by default, if $W_i^+ = W_i^- = 1$, we present *bDE* and it derivatives as following:

$$|_{t_j}^{a_j}\ m_i^0 p_i\,|_{t_k}^{a_k},\quad |_{t_j}^{a_j}\ m_i^0 p_i \quad \text{or} \quad m_i^0 p_i\,|_{t_k}^{a_k}.$$

Definition 4. A descriptive expression (*DE*) of a labeled composite HN is either *bDE* or a composition of *DE* a HN: $DE ::= bDE \mid DE * DE \mid \circ DE$, where $*$ represents any binary composition operation and $\circ$ any unary operation. ■

Descriptive Compositional Operations. In the following by default the labels of HN and HN are encoded in the name of the transitions and places. The composition operations are reflected at the level of the DE components of HN models by fusion of places, fusion of transitions with same type and same name (label) or sharing of as subnets.

Place-Sequential Operation. This binary operation, denoted by the " | " *sequential operator*, determines the logic of a interaction between two local states p_i (pre-condition) and p_k (post-condition) by t_j action that are in precedence and succeeding (causality-consequence) relation relative of this action. Sequential operator is the *basic mechanism* to build *DEs* of HNs models.

The sequential operation is an *associative*, *reflexive* and *transitive* property, but is *not commutative* operation. The $DE1 = m_i^0 p_i\ [W_i]\,|_{t_j}^{a_j} m_k^0 p_k\ [W_k] \neq m_k^0 p_k\ [W_k]\,|_{t_j}^{a_j} m_i^0 p_i\ [W_i]$ means the fact that the specified conditions (local state) associated with place- symbol p_i are fulfilled always happens before then the occurrence of the conditions associated with place-symbol p_k by means of the action t_j. The translation of $DE1$ in $HN1$ is shows in figure 3. Also, the PN modeling of the *iteration* operation is obtained by the fusion of head (entry) place with the tail (final) place that are the same name (*closing* operation) in *DE* which describes this net. The self-loop of $HN2$ described by an:

$$DE2 = m_i^0 p_i\ [W_i]\,|_{t_j}^{a_j} m_k^0 p_k\ [W_k] = m_i^0 \tilde{p}_i\ [W_i]\,|_{t_j}^{a_j},$$

it is the *test operator* "~", i.e. represent the *test arc*.

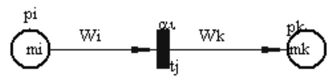

Fig. 3. Translation of $DE1$ in $HN1$.

Inhibition Operation. This unary operation is represented by *inhibitory operator* " ¯ " (place-symbol with overbar) and it $DE3 = m_i^0 \overline{p}_i\ [W_i]\,|_{t_j}^{a_j}$ describe the inhibitor arc in *HN* models with a weight function (arc multiplicity) $W_i = Inh(p_i, t_j)$.

Synchronization Operation. This binary operation is represented by the "•" or "∧" *join* operator describe the rendezvous synchronization (by transition t_j) of a two or more conditions represented respectively by symbol-place $p_i \in {}^\bullet t_j$, $i = \overline{1,n}$, i.e. it indicate that all preceding conditions of occurrence actions must have been completed. Also, this operation is a commutative, associative and reflexive property.

Split Operation: This binary operation represented by the "◊" or "(◊)" *split* or *fork* operators and it describe, determine the causal relations between activity t_j and its post-conditions: after completion of

the preceding action of t_j concomitantly several other post-condition can take occurs in parallel ("message sending"). Property of split operation is a commutative, associative and reflexive.
Thus the HN shows in figure 1a and figure 1b are described by the DE's A and B, respectively:

$$A = A1|_{t_1}\ (p_4 \Diamond b_3)\ ;\ B = B1|_{u_1}\ (6.35 b_2 \Diamond b_3)\ \text{with}$$

$$A1 = (2.4b_1 \cdot 1p_1 \cdot 2p_2 \cdot 3\tilde{p}_3 \cdot 1.5\tilde{b}_2 \cdot 5\overline{p}_4 \cdot 4.5\overline{b}_3)\,|_{t_1}\ \text{and}$$

$$B1 = (2.4b_2 \cdot 1\overline{p}_1 \cdot \tilde{p}_2 \cdot 3.6b_5 \cdot \tilde{b}_4 \cdot 4.5\overline{b}_3)\,|_{u_1}$$

Competing Parallelism Operation. This compositional binary operation is represented by the "$\vee$" competing parallelism *operator*, and it can be applied over two HN_A with $DE_A = A$ and HN_B with $DE_B = B$ or internally into resulting HN_R with $DE_R = R$, between the places of a single HN_R which the symbol-places with the same name are fused, respectively.
We can then represent the resulting DE_R= $R = A \vee B$ as a set of ordered pairs of places with the same name to be fused, with the first element belonging to A the second to B. The fused places will inherit the arcs of the place in A and B. Also, this compositional binary operation is a *commutative*, *associative* and *reflexive* property.
Precedence Relations between the Operations. We introduce the following *precedence relation* between the compositional operations in the *DE*: a) the evaluation of operations in *DE* are applied left-to-right; b) an unary operation binds stronger than a binary one; c) the "•"operation is superior to"/ " and "◊", in turn, its are superior the "∨" operation. Further details on definitions, enabling and firing rules, and evolution for discrete part of *HN* can be found in (Gutuleac, 2004) as they require a great deal of space.

4. DISCRETE- CONTINUOUS MODELING OF PIPE-LINE MULTIPROCESSOR SYSTEM

In order to illustrate *HSPNs* we give a performance modeling example. Consider a pipe-line multiprocessor system consisting of three processors elements PE_j, j=1,2,3. Each element PE_j can be in two local states $a_j \in \{0, 1\}$. In the active state $a_j = 1$, the element PE_j with speed V_j will in continuous mode decrease the level x_k of buffer b_k, k=4-j and in the same time it will in continuous mode increase the level x_j of buffer b_j, j=1, 2, 3. In passive state $a_j = 0$ it will not change them any more. The time sojourn of each element PE_j in the respective states $a_j = 1$ or $a_j = 0$ are negative exponentially distributed random variable with rate l_j or m_j.
Hence the behavioral discrete and continuous behavior process of a PE_j can represented by the descriptive expression DE_D and DE_C respectively.
The superposed discrete-continuous behavior process of system is represented by the following descriptive expression DE_{sys} of HSN model:

$$DE_{sys} = DE_D \vee DE_C\,,\ \ DE_D = \mathop{\vee}_{i=1}^{3}(p_i\,|_{t_i}^{l_i}\ p_{3+i}\,|_{t_{3+i}}^{m_i}\ p_i)\,,$$

$$DE_C = (p_1 \cdot b_3)\,|_{u_1}^{V_1}\ b_1 \vee (p_2 \cdot b_1)\,|_{u_2}^{V_2}\ b_2 \vee (p_3 \cdot b_2)\,|_{u_3}^{V_3}\ b_3\,.$$

Figure 4 show the translation of DE_{sys} in HSN_{sys}.

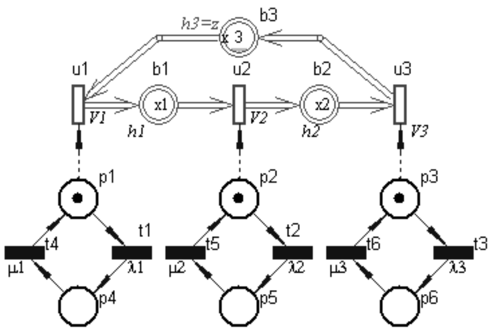

Fig. 4. Translation of DE_{sys} in HSN_{sys}.

The blocking effect of PE_j in $a_j = 1$ is represented by capacity $K_{b_j} = h_j$ of buffer b_j if this is full.
In the next, to we will note x_1=x, x_2=y and x_3=z.
The net HSN_{sys}. is bounded, live and reinitialized again because it has four P-invariants that cover all places: $m(p_j) + m(p_{j+3})$=1, j=1,2,3 for discrete places and $x + y + z = h$ for continuous places.
For initial marking $m(p_j)$=1, x_0=y_0=0, z_0= h_3=h_1+h_2 then the current state of *HSPN1* can be described by 7-tuple $(a_1 a_2 a_3,\ xy; ß_x, ß_y)$, where $ß_x$ and $ß_y$ are respective dynamic balances of buffers b_1 and b_2.
The analytical analysis of underlying *HCMC* of this HSN_{sys} model in general case is very difficult. For this analysis to be must use the special tool.
Here we give a simplified case for $l_2 = l_3$=0, where the elements PE_2 and PE_3 always will be in active state $a_2 = a_3 = 1$ and in this way, the element PE_2 (respective PE_3) with the speed V_2 (respective V_3), will transfer the content of buffer b_1 (respective b_2) in buffer b_2 (respective b_3).

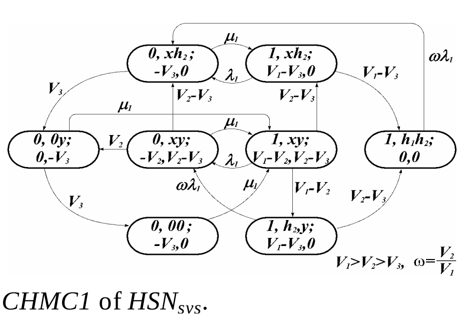

Fig. 5. *CHMC1* of HSN_{sys}.

The behavior of HSN_{sys} depends of rapport between speeds V_j. For $V_1 > V_2 > V_3$ the chain *CHMC1*, with the respective internal and boundary states, in considerate case, is represented in figure 5, where the discrete marking is $m_i \in \{0, 1\}$ because just the element PE_i can be or in passive or in active state.
Let $f_i(x, y)$ denote the steady-state fluid density of *CHMC1* in current marking (m_i, xy), i=0,1. For each

internal state (m_i, xy; v_x, v_y), $0<x<h_1$ and $0<y<h_2$ of the chain *CHMC1* the $f_i(x, y)$ obeys the following system of partial differential equations (*PDE*):

$$-V_2 \cdot \frac{\partial f_0(x,y)}{\partial x} + (V_2 - V_3)\cdot \frac{\partial f_0(x,y)}{\partial y} +$$

$$m_1 f_0(x,y) = l_1 f_0(x,y); \qquad (1)$$

$$(V_1 - V_2)\cdot \frac{\partial f_1(x,y)}{\partial x} + (V_2 - V_3)\cdot \frac{\partial f_1(x,y)}{\partial y} +$$

$$l_1 f_1(x,y) = m_1 f_0(x,y) \cdot$$

For level $z = \hat{x} + \hat{y}$ the total values of two buffers b_1 and b_2 using the same approach, with the stationary condition $m_1/V_3 < l_1/(V_1 - V_3)$, we obtain:

$$j'_0(z) = m_1 \cdot p'_0(0)e^{r_1 z}/V_2 \,,$$

$$j'_1(z) = V_3 \cdot j'_0(z)/(V_1 - V_3)\,,$$

$$r_1 = m_1/V_3 - l_1/(V_1 - V_3)$$

$$z = m_1 V_1 (V_1 - V_3)/((l_1 + m_1)(l_1 V_3 - m_1(V_1 - V_3))).$$

Because we know some filled and emptied numeric behaviour characteristics of buffer b_1 for $0 < h_i < ¥$, $i = 1,2,3$ we can look the solution of PDE equations in following multiplicative form:

$$f_0(x,y) = j_0(x)\cdot y_0(y),$$

$$f_1(x,y) = j_1(x)\cdot y_1(y) = A\cdot f_0(x,y), \text{ where}$$

$$j_0(x) = C\cdot e^{g_1 x}, \quad y_0(y) = e^{g_2 x}, \qquad (2)$$

$$j_1(x) = a_1 j_0(x), \quad y_1(y) = a_2 y_0(y),$$

$$g_1 = r_1 m_1/V_2 - l_1/(V_1 - V_2),\ a_2 = A/a_1\,,$$

$$a_1 = V_3/(V_2 - V_3)\,.$$

The value C is a constant obtained from the normalization condition, but g_1 and A are obtained like solution of characteristic equation of system *PDE* (1): $A^2 + b\cdot A - r = 0$, where $r = l_1/m_1$,

$$b = 1 + V_1/V_2 - (r(2V_1 - V_2)/(V_1 - V_2)).$$

From this characteristic equation and from condition that density probability always is a positive value we obtain only one positive solution, $A>0$. Using what we determine $g_1 = (Am_1 + gV_2 - l)/(V_2 - V_3)$ which give the second stationary condition $Am_1 > l - gV_2$.

To write the boundary equation directly from graph of chain *CHMC1* we introduce the notation: $p_i(0)$ or $p_i(h_1)$, thich are probabilities of boundary states of buffer b_1 for x=0 or x=h_1, but $Q(0)$ or $Q(h_2)$ of buffer b_2 for y=0 or y=h_2, respectivelly.

For each state with $w = V_2/V_1$ we can write the steady-state boundary equations:

$$wl_1 \times p_1(h_1) = V_3 \times j_0(h_1);\ wl_1 \times p_1(h_1) = (V_2 - V_3)\times j_1(h_1);$$

$$wl_1 \times Q_1(h_2) = (V_2 - V_3)\times j_1(h_2);\ m_1 \times p_0(0) = (V_1 - V_2)\times j_1(0);$$

$$m_1 \times Q_0(0) = V_3 \times y_1(0);\ m_1 \times Q_0(0)\times y_0(h_2) = V_3 \times j_0(0) Q_0(h_2).$$

The solution of this equation system is given by:

$$p_0(0) = C\cdot a_1 \cdot (V_1 - V_2)/m_1,$$

$$Q_0(0) = V_3/m_1;\ p_1(h_1) = c\cdot V_3/wl_1 \cdot e^{g_1 h_1}\,,$$

$$Q_0(h_2) = (V_1 - V_2)/(V_2 - V_3)\cdot e^{g_2 h_2}\,,$$

$$Q_1(h_2) = (V_2 - V_3)a_2/(wl_1)\cdot e^{g_2 h_2}\,.$$

Taking in consideration thus relations we obtain the average levels $\hat{x}$ and $\hat{y}$ in buffers b_1 and b_2:

$$\hat{x} = C\cdot\left[(V_3 h_1/m_1 l_1 + (1+a_1)(g_1 h_1 - 1)/g_1^2)e^{g_1 h_1} + (1+a_1)/g_1^2\right],$$

$$\hat{y} = C\cdot\left[(h_2((V_1 - V_2)/(V_2 - V_3) + (V_2 - V_3)a_2/ml_1) + D\right],$$

where $D = (1+a_2)(g_2 h_2 - 1)/g_2^2)e^{g_2 h_2} + (1+a_2)/g_2^2$.

The time redundancy of system is a: $t_x = \hat{x}/V_2$ and $t_y = \hat{y}/V_3$. The some considerations hold for the throughput of system.

CONCLUSIONS

In this paper we have defined a set of descriptive composition operation and descriptive expressions for the creation of HSPN models from the behavioral state based discrete-continuous process of system components. The descriptive compositional approach can preserve the functional structure of the model and support several type of communication between components. In this way we support the performance modeling of distributed and parallel systems where both synchronous and asynchronous communication is required. To illustrate the use of this approach, we apply them to a performance modeling a pipe-line multiprocessor system.

The translation rules of descriptive expressions into HSPN model summarized and illustrated in this paper suggest that it may be possible to develop a tool to combining the visualization of small components by mean of HSPN and the composition mechanisms of this approach.

REFERENCES

Alla A., H. David (1998). *Continuous and hybrid Petri nets*, Journal of systems Circuits and Computers, 8(1) pages 159-188.

Gutuleac E. (2004). *Descriptive Compositional Construction of GSPN Models for Performance Evaluation of Computer Systems.* In: Proc.of 8-th International Symposium on Automatic Control and Computer Science, SACCS2004, Iasi, România, pp. 43-50.

Gutuleac E., V. Gîsca, S. Zaporojan, N. Objelean (2002). *Hybrid stochastic Petri nets for the discrete-continuous modeling and performance evaluation of computers systems.* In: Proc. of the 6-th Intern. Conf. on DAS2002, 23-25 May 2002, Suceava, România, pp. 279-286.

Horton G., V. Kulkarni, D. Nicol, K. Trivedi (1996) *Fluid Stochastic Petri Nets: Theory, Applications and Solution.* ICASE Technical Report 96-5.

Wolter K., Hommel G. (1997) *Hybrid Modelling with Second Order Fluid Stochastic Petri Nets.* In Proc. Workshop on Parallel and Distr. Real-Time Systems, Geneva, IEEE-CS, pp. 239-243.