Asian Journal of Informational Technology 5 (10): 1117-1125, 2006
© Medwell Online, 2006

Controlled Dan/Petri Nets for Modeling Multiple and Simultaneous Control of Discrete Event Systems

Eleazar Jiménez Serrano, Keijiro Araki and Shigeru Kusakabe
Department of Computer Sciences and Electrical Engineering,
Kyushu University, Hakozaki 6-10-1, Higashi-ku, Fukuoka 812-8551, Japan

Abstract: For the purpose of modeling Discrete Event Systems (DES), a Petri net-based modeling framework called Dan/Petri Net (D/PN) is presented, working only with subnets and having a different enabling rule based on an arcs-counting function and non-boolean guards called dans. And Controlled D/PN is one extension developed to explain the modeling of multiple and simultaneous control (MSC) in DES through the fundamental MSC subnet and its notation convention, called valid MSC subnet $\Sigma^{c(q)}$ and its derivative subnet $\Sigma_n^{a(q)}$. At the end of this paper we present the usage of Controlled D/PN in the MSC design for an D/PN model through MSC subnets.

Key words: Petri nets, simulation, discrete event systems, control systems

INTRODUCTION

Petri Net (PN) has proved for many years to be not just a classic tool, but a very useful in the modeling[1,2] and control[3,4] of distributed dynamic systems which present concurrency and non-determinism.

This study tackles the modeling of Multiple and Simultaneous Control (MSC) of Discrete Event Systems (DES). These are systems where, from a set of conditions B, there exist a symmetric combinatorial series of causal dependencies (called events) between one (initial) condition $b_i \in B$ with a subset of final conditions $B_f \subset B$. And these events can be controlled with a subset of controllers which can be activated individually, associatively and collectively, called MSC, representing individual and multiple controls.

For these, we have created Dan/Petri Net (D/PN) which can model the state-transition structure of DES like an ordinary PN despite the novelty of working with single-transition subnets and having a different enabling rule based on two new elements: an arcs-counting function for each transition and non-boolean guards called dans. Later, from the results of Krogh's Controlled Petri nets (CtlPN's) and feedback logic for marked graphs[5] and Ichikawa and Hiraishi's decision-free Petri net with external input and output places[6], we identify external input ports (called controllers from now on) which could simultaneously provide individual and multiple controls in a system (MSC) and to model this MSC-logic, we defined a D/PN-based modeling framework called Controlled Dan/Petri net (Controlled D/PN), which uses a different type of places called controllers restricted to exist only in MSC subnets (Jiménez, Araki and Kusakabe 2006). The relationship between PN, D/PN and Controlled D/PN can be more easily understandable in the Fig. 1.

The first and main different between our control modeling framework and the previous two is that, one controlled transition can have one or more controllers and one controller can control one or more controlled transitions, which is the fundament for MSC subnets. And the way to model MSC subnets are based on the amount of controlled transitions obtained for each possible combinatorial relation of controllers. These relations are a cluster of siphons in an MSC subnet, where

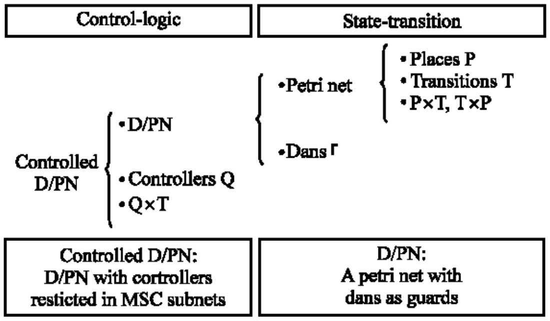

Fig. 1: Diagram of D/PN and Controlled D/PN

Corresponding Author: Eleazar Jiménez Serrano, Department of Computer Sciences and Electrical Engineering, Kyushu University, Hakozaki 6-10-1, Higashi-ku, Fukuoka 812-8551, Japan

we can see one siphon when all the controllers form a collective subset, siphons when the controllers form associate subsets and siphons when the controllers form individual subsets.

The fundamental MSC subnet contains a complete combinatorial arrangement of controlled transitions and allow just one controlled transition to be enabled through the enabling rule and it is called a valid MSC subnet $\Sigma^{c(q)}$ (or just VMSCS $\Sigma^{c(q)}$). And when this subnet lacks of all the transitions controlled individually by one controller and lack of all the transitions controlled associatively by more than one controller up to a specific associative control, it is a derivative subnet called the VMSCS $\Sigma_n^{a(q)}$.

Without loss of generality, the simplest VMSCS $\Sigma^{c(q)}$ is the net $\Sigma^{c(2)}$ containing just two controllers q_1 and q_2 which can change the state of the DES Fig. 2c.

For controlled D/PN, from the distributive perspective, the state-transition is preserved through D/PN and so it is non-determinism between all types of subnets (single-transition subnets and MSC subnets). This lead us to use the words concurrency and simultaneity by separate, limiting the usage of one of them under the Controlled D/PN domains and explaining them later.

This study is constructed as follows: section two describes D/PN and controlled D/PN; section three explains the VMSCS $\Sigma^{c(q)}$ and the derivative VMSCS $\Sigma_n^{a(q)}$; section four explain the properties of the VMSCS $\Sigma^{c(q)}$ and conditions to preserve the properties in the derivative VMSCS $\Sigma_n^{a(q)}$; section five explains a simple design of MSC for a bounded D/PN model based on the algorithm to design MSC for marking control in[6]. In the last section are the conclusions and further research.

D/PN AND CONTROLLED D/PN

A D/PN is a PN-based modeling framework formally introduced in[7] and used to model DES. It is a single-arc, safe and self-loop free PN extension.

A Petri net is a tuple M = (P, T, I, O) which is a finite bipartite directed graph where:

- P is a finite non-empty set of conditions called places,
- T is a finite non-empty set of transitions where $P \cap T = \varnothing$,
- $I \subseteq (P \times T)$ is the set of directed arcs connecting places to transitions and
- $O \subseteq (T \times P)$ is the set of directed arcs connecting transitions to places.

Places are graphically represented by circles, transitions by rectangles and all directed arcs by arrows. Let $t \in T$. If $I(p, t) \neq \varnothing$ then we call •t the pre-conditions of t and they constitute the preset of input places of t. If $O(t, p) \neq \varnothing$ then we call t• the post-conditions of t and they constitute the postset of output places of t.

Let $p \in P$. If $O(t, p) \neq \varnothing$ then we call •p the pre-events of p and they constitute the preset of input transitions of p. If $I(p, t) \neq \varnothing$ then we call p• as the post-events of p and they constitute the postset of output transition of p.

A non-empty subset of places $S \subseteq P$ is called a *Siphon* if $\bullet S \subseteq S\bullet$ and has the property that once it is empty of tokens, it will remain empty.

And a non-empty subset of places $S \subseteq P$ is called a Trap if $S\bullet \subseteq \bullet S$ and has the property that if it has one token, it will continue having at least one token.

We suppose that the reader is familiar with the PN terminology, therefore most of the PN characterizations are assumed for D/PN, except the ones which are explained.

A D/PN is a tuple N = (M, **Γ**), where:

- M is a PN,
- **Γ** is a finite non-empty set of alphanumerical guards called dans,

The set $T \in M$ is redefined as a finite non-empty set of single-transition subnets, (or sometimes just transitions)□ and dans are graphically represented by subnet identifiers in the transitions.

The set of dans **Γ** defines the subnets. For D/PN, a subnet is composed by one transition with one unique dan, having a subset of input and output places.

The subnet-belonging function g maps T to **Γ** where $\forall t \in T$, $g(t) \in \mathbf{\Gamma}$, i.e., no transition has the dan null.

For an arbitrary transition $t \in T$, we define the amount of all directed arcs of the set I going to t as the function c (t), mapping c (t) to {1, 2,. ..}.

Let m be a function of set P called marking, mapping P to {0, 1}, where m_0 is the initial marking. We say that a place p is marked iff m (p) = 1. The finite set of all possible markings (i.e., the state space) for a D/PN is denoted by M. A D/PN with initial marking will be denoted by (N, m_0). For a D/PN N, suppose there is one arbitrary subnet with one transition $t \in T$ and has assigned one dan $\lambda \in \mathbf{\Gamma}$. We call t* an enabled transition in a marking ***m*** when it covers its input places but not its output places and has the highest value when evaluating c (t*) against all other transitions with the same dan; i.e. $\forall t \in T : g(t) = \lambda, \exists\_t^* : c(t^*) > c(t) \wedge [\bullet t^* \subseteq m] \wedge [t^*\bullet \cap m = \varnothing]$.

The set of all enabled transitions in a marking m is defined as T* (m) or just T*.

The basic characterization of D/PN is just to add one unique dan to each transition and yet the behavior of the D/PN prevail the same as an ordinary PN when modeling the state-transition of a DES. Out of D/PN and for the case when more than one transitions have the same dan, these transitions, their input and output places and arcs belong to one subnet of single-transitions subnets (or just called subnet of transitions) and only exist in another modeling framework called D^+/PN, which is implicitly assumed in all this paper for modeling Controlled D/PN and explained next.

Controlled D/PN: A Controlled D/PN is a D/PN with extended modeling capabilities intended to model MSC. From Ichikawa and Hiraishi's results on the control of DES, some of their characterizations are used for our Controlled D/PN, except the ones explained here.

A Controlled D/PN is a tuple G = (N, Q, D), where N is a D/PN defined previously, Q is a finite set of places called *controllers* which are graphically represented by circles with thicker lines (to distinguish from regular places) and D ⊆ (Q×T) is the set of directed arcs connecting controllers to transitions of N and they are represented by arrows.

Let t ∈ T. If D (q, t) ≠ ∅ then we call •t the controllers of t and they constitute the preset of controllers of t.

Let q ∈ Q. If D (q, t) ≠ ∅ then we call q• as the control of q and they constitute the postset of controlled transitions of q.

For Controlled D/PN, the set of transitions T can be seen as two subsets, one of uncontrolled transitions $T_u = \{t \in T \mid \forall q \in Q, (q, t) \notin D\}$ where each transition have one unique dan and are called uncontrolled single-transition subnets (or just called single-transition subnets) Fig. 2a and the other subset of controlled transitions $T_c = \{t \in T \mid (q, t) \in D\}$ where each transition have one unique dan and are called controlled single-transition subnets (or just controlled transition) Fig. 2b. When more than one controlled transition has the same dan, they belong together with their pre and post-conditions and arcs to a subnet of controlled single-transition subnets (or just subnet of controlled transitions) Fig. 2c. Every controlled single-transition subnet and every subnet of controlled single-transition subnets are called MSC subnets.

Let u be a function of the set Q called control, mapping Q to {0, 1}, where u_0 is the initial control. We say that a controller q is active iff u (q) = 1. The finite set of admissible controls for a Controlled D/PN is denoted by U. A Controlled D/PN with initial marking and control will be denoted by (G, m_0, u_0). The marking of a controller q in the net is not necessarily always defined in the initial condition. Tokens can be deposited consecutively in Q at any discrete step, making the marking of one controller an individual control and an associative control or collective control if tokens are simultaneously put in more than one controller or all controllers.

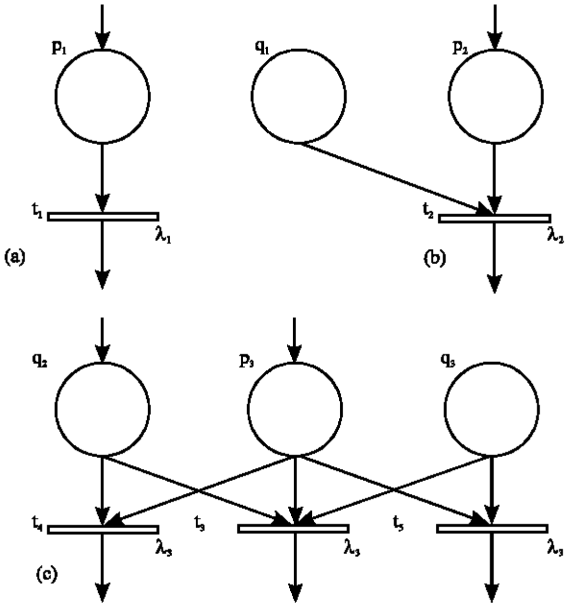

Fig. 2: The three basic types of subnets: a) subnet of (uncontrolled) single-transition; b) VMSCS $\Sigma^{c(1)}$ with a controlled single-transition; and c) VMSCS $\Sigma^{c(2)}$ of controlled transitions

The time is not specified in this modeling framework and discrete step is defined as the instant where a state-transition occurs.

Controllers have capacity of one token and can hold it during just one discrete step. Each token in a controller represent an instantaneous control impulse in a system with a dependent existence based on the condition of its MSC module. If in the state of the system the condition of the MSC module does not hold, a control can not exist. Therefore, in order for a token to exist in a controller and before it is consumed by a controlled transition, a token must exist in the input place of the same MSC subnet. More than one controller can be assigned to one transition of the same MSC subnet and one controller can be assigned to different transitions of the same MSC subnet.

For a Controlled D/PN G, suppose there is one arbitrary MSC subnet with one dan λ ∈ Γ assigned to all its controlled transitions $t_c \in T_c$. We call t_c* an enabled transition in a marking m and control u, when the control u covers its controllers, the marking m covers its input places but not its output places and has the highest value when evaluating c (t_c*) against other controlled transitions with the same dan and covered under the same marking m and control u in the MSC subnet. Formally: $\forall t_c \in T_c : g(t_c) = \lambda, \exists\, t_c^* : c(t_c^*) > c(t_c) \wedge [\bullet t_c^* \subseteq m] \wedge [t_c^*\bullet \cap m = \varnothing] \wedge\_[\bullet t_c^* \subseteq u]$.

For controlled D/PN, the function c (t) in the enabling rule considers only the directed arcs from the set D since the relations in the set I are implicitly assumed through the marking m. And for the underlying D/PN, all uncontrolled transitions in T_u are control active by nature and follow the enabling rule of D/PN.

For a Controlled D/PN G containing uncontrolled and controlled transitions at a given marking m and control u, the set of all enabled transitions (uncontrolled t_u and controlled t_c) is defined as T_c*(m, u) or just T_c*.

Uncontrolled transitions are single-transition subnets of D/PN and they are left to fire when they have been enabled. Controlled transitions exists in MSC subnets of Controlled D/PN and fire when they are fireable (i.e., there is not just a marking m, but a control u). The result will be the same as an ordinary PN, to eliminate the marking m (and control u) and to create a new making m'.

From between single-transitions subnets and controlled single-transition subnet, more than one transition is allowed to fire at a discrete step when they are concurrently fireable. And from within subnets of controlled transitions, more than one transition could fire in a subnet, but when only one transition can fire, this subnet is called a Valid MSC Subnet (VMSCS). A controlled single-transition subnet is also a VMSCS. The VMSCS will be explained in details in the next section.

Given a marking and control (m, u) of G, the firing of all the enabled transitions T_c*(m, u) in G, which eliminates the marking m and the control u and creates the new marking m' is defined as (m, u)[T_c*>m'. Its effect is to eliminate the token in m and the control u and to put new tokens in m'.

Now, from the definitions of state and transition provided in the common literature of PN, we have limited the usage of the word *simultaneous* under Controlled D/PN domains, saying that ⌈a (distributed) state is a set of conditions holding simultaneously⌋v; for the previous definitions, m' is said to be marked simultaneously. And so it is for controllers, saying that ⌈putting tokens in more than one controller at the same discrete step is said to be a simultaneous marking⌋. Now we have respected the usual meaning of the word concurrency, saying that ⌈a (distributed) transition is a set of events occurring concurrently⌋, it is two or more transitions which are causally independent from firing at the same time. Therefore, from the previous definition, the enabled transitions of T_c*(m, u) are said to fire concurrently.

THE VALID MSC SUBNET $\Sigma^{c(q)}$ AND THE DERIVATIVE SUBNET $\Sigma_n^{a(q)}$

For controlled D/PN, an MSC subnet is composed of: a subset of controllers Q, one unique input place $p_i \in P$, one arbitrary dan $\lambda \in \Gamma$, a subset of controlled transitions where $\forall\, t_c \in T_c : g(t_c) = \lambda$, a subset of output places $P_f \subset P$ and a subset of directed arcs I, O and D.

An MSC subnet is called a VMSCS iff under contact-free conditions in the subnet (i.e. there are no tokens in the output places of the subnet) and throughout the enabling rule, there exists just one unique enabled transition t_c* in the MSC subnet for a marking m and control u. Controlled D/PN must work only with VMSCS's in order to congruently model the control-logic of any MSC module.

A controlled D/PN is called a net $\Sigma^{c(q)}$ iff it contains just one VMSCS with $\Sigma^{c(q)}$ symmetry. A net $\Sigma^{c(q)}$ is composed of: a set of controllers Q, one unique input place $p_i \in P$, one arbitrary dan $\lambda \in \Gamma$, a set of controlled transitions where $\forall\, t_c \in T_c : g(t_c) = \lambda$, a subset of output places $P_f \subseteq P$ and the sets of directed arcs I, O and D. The $\Sigma^{c(q)}$ symmetry refers to the possible amount of controlled transitions T_c in the net $\Sigma^{c(q)}$, which follow the symmetric series of 1, 3, 7, 15,. .. and is represented in the formula (1), described from the exhaustive or complete (collectively, associatively and individually) combinatorial amount of controllers Q (only when it is > 2). Lets assign k as the cardinality of set Q and define the amount of controlled transitions T_c in $\Sigma^{c(q)}$ by:

$$|T_c| = 1 + \sum_{i=2 \to k-1} ({}^{k}|_{i}) + k. \tag{1}$$

where $({}^{a}|_{b}) = a!/(a-b)!b!$.

A controlled D/PN is called a net $\Sigma_n^{a(q)}$ iff it contains just one VMSCS $\Sigma^{c(q)}$ which lacks of the transitions controlled individually (by one controller) and some transitions controlled associatively (by more than one controller, except the collective association) up to an n^{th} trimmed control. When net $\Sigma^{c(q)}$ was trimmed from all the transitions controlled by one or two controllers, we say it is a net $\Sigma_2^{a(q)}$. When it was trimmed from all the transitions controlled by three or less controllers, we say it is a net $\Sigma_3^{a(q)}$, etc. Therefore the net $\Sigma_n^{a(q)}$ is an absolutely-trimmed VMSCS $\Sigma^{c(q)}$ with $\Sigma^{c(q)}$ symmetry trimmed at n^{th}-associative control (for n > 1). The possible amount of controlled transitions T_c in $\Sigma_n^{a(q)}$ trimmed at n^{th}-associative control follows the series represented in the formula (2) below.

They describe the collectively and associatively (but not individually) combinatorial amount of controllers (only for Q > 2) trimmed at the associative control n. Lets assign k as the cardinality of set Q and x as the trimmed control n and define the amount of controlled transitions T_c in $\Sigma_n^{a(q)}$ by:

$$|T_c| = 1 + \sum_{j=x \to k-1} (^{k}|_{j}). \qquad (2)$$

Individual, associative and collective control: MSC means the existence of three types of controls: individual, associative and collective. We talk about individual (or single) control when one controller controls a transition and defined with n = 1. Associative control when two or more controllers control a transition, defined with n = 2 for two controllers, n = 3 for three controllers, etc. And collective control when all the controllers control a transition and defined with n = k.

For the combinations of $(^{k}|_{k})$ controllers represented by the number one in the formula (1) and (2), we have the subset $\Sigma^{(k|k)} = \{q_1, q_2, .. ., q_k\}$. We call the collective control n = k of the VMSCS as the subset $C \subset T_c$ containing $(^{k}|_{k})$ (i.e. one) controlled transition $t_{c1|k} \in C$. The union $\{ p_i \} \cup \Sigma^{(k|k)} \in \bullet t_{c1|k}$. And it contains a subset of k directed arcs $D_c \subset D$ where each arc connects every controller in $\Sigma^{(k|k)}$ with $t_{c1|k}$; and a subset with one directed arc $I_c \subset I$ where the arc connects p_i with $t_{c1|k}$.

When the VMSCS $\Sigma^{c(q)}$ is trimmed at the collective control (when n = k), it is a VMSCS $\Sigma_n^{a(q)}$ called $\Sigma_2^{a(2)}$ with one controlled single-transition and 2 controllers Fig. 3, $\Sigma_3^{a(3)}$ for 3 controllers Fig. 4c, $\Sigma_4^{a(4)}$ for 4 controllers Fig. 5d), etc.

For the combinations of $(^{k}|_{k-1})$ controllers, we have the subset $Q^{(k|k-1)} = \{(q_1, q_2, ..., q_{k-1}), (q_1, q_2, .. ., q_{k-2}, q_k), (q_1, q_2, .. ., q_{k-3}, q_{k-1}, q_k), .. ., (q_2, q_3, .. ., q_{k-2}, q_{k-1}, q_k)\}$. We call the associative control n = k-1 of the VMSCS as the subset $A_{k-1} \subset T_c$ containing $(^{k}|_{k-1})$ controlled transitions $t_{c1|k-1}, t_{c2|k-1}, \ldots, t_{c(k|k-1)|k-1} \in A_{k-1}$. The union of $\{ p_i \}$ with every controller of the first element of the subset $Q^{(k/k-1)} \in \bullet t_{c1|k-1}$; the union of $\{ p_i \}$ with every controller of the second element of the subset $Q^{(k/k-1)} \in \bullet t_{c2|k-1}$;. ..; and the union of $\{ p_i \}$ with every controller of the l ast element of the subset $Q^{(k/k-1)} \in \bullet t_{c(k|k-1)|k-1}$. And it contains a subset of $(k-1)*(^{k}|_{k-1})$ directed arcs $D_{k-1} \subset D$ where each arc connects every controller of every element in $Q^{(k|k-1)}$ with its corresponding controlled transition in A_{k-1} and a subset of $(^{k}|_{k-1})$ directed arcs $I_{k-1} \subset I$ where each arc connects p_i with every controlled transition in A_{k-1}.

When the VMSCS $Ó^{c(q)}$ is trimmed at the associative control n = k-1, it is a VMSCS $\Sigma_n^{a(q)}$ called $\Sigma_2^{a(3)}$ with 4 controlled transitions and 3 controllers Fig. 4b, $\Sigma_3^{a(4)}$ with 5 controlled transitions and 4 controllers Fig. 5c, $\Sigma_4^{a(5)}$ with 6 controlled transitions and 5 controllers, etc.

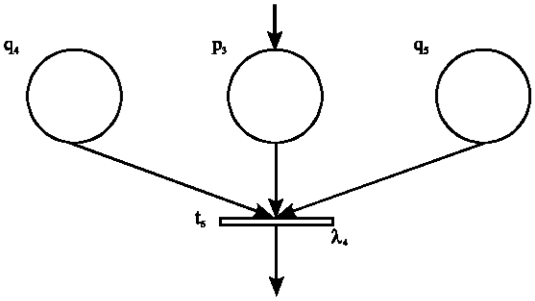

Fig. 3: The controlled D/PN $\Sigma_2^{a(2)}$. The postset of output places can be arranged in any association

For the combinations of $(^{k}|_{k-2})$ controllers, we have the subset $Q^{(k/k-2)} = \{(q_1, q_2, ..., q_{k-2}), (q_1, q_2, .. ., q_{k-3}, q_{k-1}), (q_1, q_2, .. ., q_{k-3}, q_k), .. ., (q_3, .. ., q_{k-1}, q_k)\}$. We call the associative control n = k-2 of the VMSCS as the subset $A_{k-2} \subset T_c$ containing $(^{k}|_{k-2})$ controlled transitions $t_{c1|k-2}, t_{c2|k-2}, \ldots, t_{c(k|k-2)|k-2} \in A_{k-2}$. The union of $\{p_i\}$ with every controller of the first element of the subset $Q^{(k|k-2)} \in \bullet t_{c1|k-2}$; the union of $\{p_i\}$ with every controller of the second element of the subset $Q^{(k|k-2)} \in \bullet t_{c2|k-2}$;. .. and the union of $\{p_i\}$ with every controller of the last element of the subset $Q^{(k|k-2)} \in \bullet t_{c(k|k-2)|k-2}$. And it contains a subset of $(k-2)*(^{k}|_{k-2})$ directed arcs $D_{k-2} \subset D$ where each arc connects every controller of every element in $Q^{(k|k-2)}$ with its corresponding controlled transition in A_{k-2} and a subset of $(^{k}|_{k-2})$ directed arcs $I_{k-2} \subset I$ where each arc connects p_i with every controlled transition in A_{k-2}.

When the VMSCS $\Sigma^{c(q)}$ is trimmed at the associative control n = k-2, it is a VMSCS $\Sigma_n^{a(q)}$ called $\Sigma_2^{a(4)}$ with 11 controlled transitions and 4 controllers Fig. 5b, $\Sigma_3^{a(5)}$ with 16 controlled transitions and 5 controllers, $\Sigma_4^{a(6)}$ with 22 controlled transitions and 6 controllers, etc.

And by following the same procedure for the combinations of $(^{k}|_{k-3}), (^{k}|_{k-4}), .. ., (^{k}|_{2})$ controllers, we have the subsets $Q^{(k|k-3)}, Q^{(k|k-4)}, .. ., Q^{(k|2)}$. We call the associative control n = k-3 of the VMSCS as the subset $A_{k-3} \subset T_c$ containing $(^{k}|_{k-3})$ controlled transitions $t_{c1|k-3}, t_{c2|k-3}, \ldots, t_{c(k|k-3)|k-3} \in A_{k-3}$; associative control n = k-4 of the VMSCS as the subset $A_{k-4} \subset T_c$ containing $(^{k}|_{k-4})$ controlled transitions $t_{c1|k-4}, t_{c2|k-4}, \ldots, t_{c(k|k-4)|k-4} \in A_{k-4}$;. ..; associative control n = 2 of the VMSCS as the subset $A_2 \subset T_c$ containing $(^{k}|_{2})$ controlled transitions $t_{c1|2}, t_{c2|2}, \ldots, t_{c(k|2)|2} \in A_2$.

The union of $\{p_i\}$ with every controller of the first element of the subset $Q^{(k/k-3)} \in \bullet\, t_{c1|k-3}$; the union of $\{p_i\}$ with every controller of the second element of the subset $Q^{(k/k-3)} \in \bullet t_{c2|k-3}$;. .. and the union of $\{p_i\}$ with every controller of the last element of the subset $Q^{(k/k-3)} \in \bullet\, t_{c(k|k-3)|k-3}$. And it contains a subset of $(k-3)*(^{k}|_{k-3})$ directed arcs $D_{k-3} \subset D$ where each arc connects every controller of every element

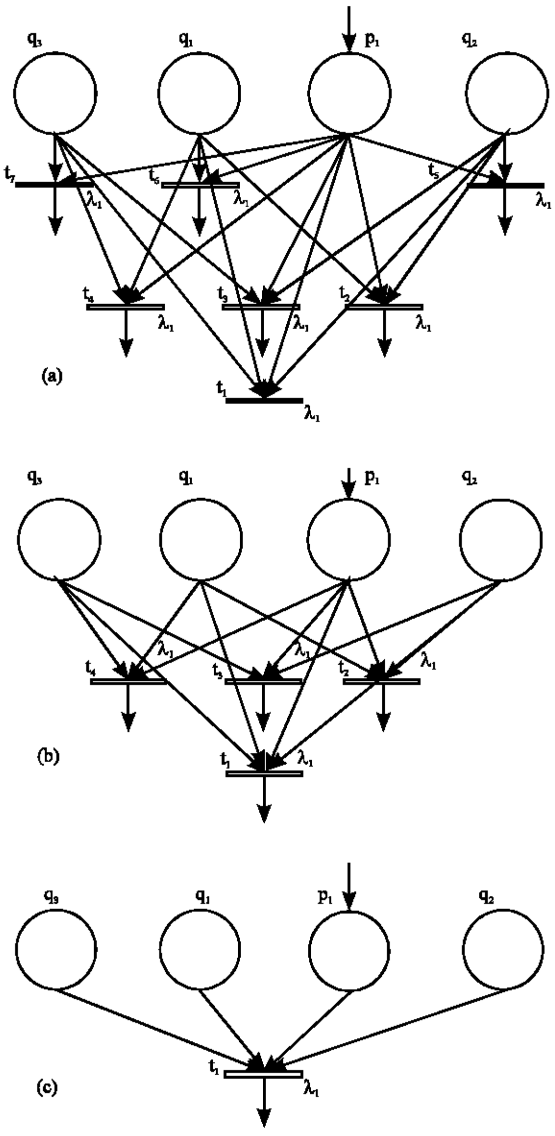

Fig. 4: The controlled D/PN $\Sigma^{c(3)}$ (a), $\Sigma_2{}^{a(3)}$ (b) and $\Sigma_3{}^{a(3)}$ (c). The Postset of Output Places Can Be Arranged in Any Association

in $Q^{(k|k-3)}$ with its corresponding controlled transition in A_{k-3} and a subset of $(^k|_{k-3})$ directed arcs $I_{k-3} \subset I$ where each arc connects p_i with every controlled transition in A_{k-3}.

When the VMSCS $Ó^{c(q)}$ is trimmed at the associative control n = k-3, it is a VMSCS $\Sigma_n{}^{a(q)}$ called $\Sigma_2{}^{a(5)}$ with 26 controlled transitions and 5 controllers, $\Sigma_3{}^{a(6)}$ with 42 controlled transitions and 6 controllers, $\Sigma_4{}^{a(7)}$ with 64 controlled transitions and 7 controllers, etc.

The union of $\{p_i\}$ with every controller of the first element of the subset $Q^{(k/k-4)} \in \bullet t_{c1|k-4}$; the union of $\{p_i\}$ with every controller of the second element of the subset $Q^{(k/k-4)} \in \bullet t_{c2|k-4}$;. .. and the union of $\{p_i\}$ with every controller of the last element of the subset $Q^{(k/k-4)} \in \bullet t_{c(k|k-4)|k-4}$. And it contains a subset of (k-4)*$(^k|_{k-4})$ directed arcs $D_{k-4} \subset D$ where each arc connects every controller of every element in $Q^{(k|k-4)}$ with its corresponding controlled transition in A_{k-4}; and a subset of $(^k|_{k-4})$ directed arcs $I_{k-4} \subset I$ where each arc connects p_i with every controlled transition in A_{k-4}.

When the VMSCS $\Sigma^{c(q)}$ is trimmed at the associative control n = k-4, it is a VMSCS $\Sigma_n{}^{a(q)}$ called $\Sigma_2{}^{a(6)}$ with 57 controlled transitions and 6 controllers, $\Sigma_3{}^{a(7)}$ with 99 controlled transitions and 7 controllers, $\Sigma_4{}^{a(8)}$ with 163 controlled transitions and 8 controllers, etc.

And we follow same procedure until reaching the union of $\{p_i\}$ with the two controllers of the first element of the subset $Q^{(k/2)} \in \bullet t_{c1|2}$; the union of $\{p_i\}$ with the two controllers of the second element of the subset $Q^{(k/2)} \in \bullet t_{c2|2}$;. .. and the union of $\{p_i\}$ with the two controllers of the last element of the subset $Q^{(k/2)} \in \bullet t_{c(k|2)|2}$. And it contains a subset of (2)*$(^k|_2)$ directed arcs $D_2 \subset D$ where each arc connects the two controller of every element in $Q^{(k|2)}$ with its corresponding controlled transition in A_2 and a subset of $(^k|_2)$ directed arcs $I_2 \subset I$ where each arc connects p_i with every controlled transition in A_2.

When the VMSCS $\Sigma^{c(q)}$ is trimmed at the associative control n = 2, it is a VMSCS $\Sigma_n{}^{a(q)}$ called $\Sigma_2{}^{a(2)}$ with 1 controlled transition and 2 controllers (since $\Sigma^{c(2)}$ has 2 associative controls, it can also be said to be trimmed at n = k, Fig. 3), $\Sigma_2{}^{a(3)}$ with 4 controlled transition and 3 controllers (since $\Sigma^{c(3)}$ has 3 associative controls, it can also be said to be trimmed at n = k-1 Fig. 4b), $\Sigma_2{}^{c(4)}$ with 11 controlled transitions and 4 controllers (since $\Sigma^{c(4)}$ has 4 associative controls, it can also be said to be trimmed at n = k-2, Fig. 5b), $\Sigma_2{}^{a(5)}$ with 26 controlled transition and 5 controllers (since $\Sigma^{c(5)}$ has 5 associative controls, it can also be said to be trimmed at n = k-3), etc. The subset of associative control A_2 is the lowest associative control where the VMSCS $\Sigma^{c(q)}$ can be trimmed and they are all the derivative subnets $\Sigma_2{}^{a(q)}$.

Finally, for the combinations of $(^k|_1)$ controllers represented by k just in the three previous formulas, we have the subset $Q^{(k|1)} = \{(q_1), (q_2). .., (q_k)\}$. We call the individual (or single) control of the VMSCS as the subset $S \subset T_c$ containing $(^k|_1)$ (i.e., k) controlled transitions $t_{c1|1}, t_{c2|1}, \ldots, t_{c(k|1)|1} \in S$. The union of {p } with the first element of the set $Q^{(k/1)} \in \bullet t_{c1|1}$; the union of $\{p_i\}$ with the second element of the set $Q^{(k/1)} \in \bullet t_{c2|1}$;. ..; and the union of $\{p_i\}$ with the last element of the set $Q^{(k/1)} \in \bullet t_{c(k|1)|1}$. And it contains a subset of k directed arcs $D_s \subset D$ where each arc connects every element in $Q^{(k|1)}$ with its corresponding controlled transition in S; and a subset of k directed arcs $I_1 \subset I$ where each arc connects p_i with every controlled transition in S.

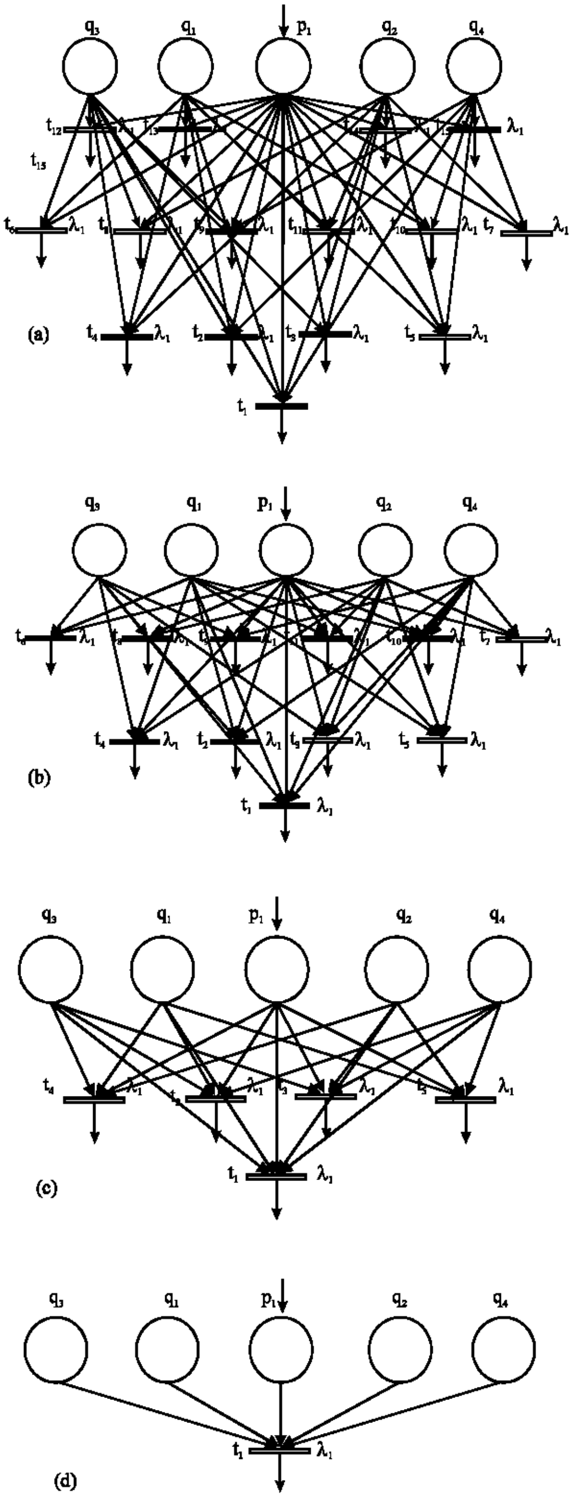

Fig. 5: The Controlled D/PN $\Sigma^{c(4)}$ (a), $\Sigma_2^{a(4)}$ (b), $\Sigma_3^{a(4)}$ (c) and $\Sigma_4^{a(4)}$ (d). The postset of output places can be arranged in any association

The VMSCS $\Sigma^{c(q)}$ can not be trimmed at the last (individual or single) control because it will lead to a redundant notation, therefore is excluded from the formula (2). The VMSCS $\Sigma_n^{a(q)}$ called $\Sigma_1^{a(1)}$ is the VMSCS $\Sigma^{c(1)}$ with one controlled single-transition Fig. 2b. The VMSCS $\Sigma_1^{a(2)}$ is the subnet $\Sigma^{c(2)}$ with three controlled transitions Fig. 2c; $\Sigma_1^{a(3)}$ is $\Sigma^{c(3)}$ with seven controlled transitions Fig. 4a, $\Sigma_1^{a(4)}$ is $\Sigma^{c(4)}$ with fifteen controlled transitions Fig. 5a, etc.

Now, out of the formula (1) and for the cases where there exist two controllers (i.e., when k = 2) for the VMSCS $\Sigma^{c(q)}$, the number of transitions is calculated with $({}^{2}|_{1}) + ({}^{2}|_{2}) = 3$, containing four directed arcs $d \in D$ and three directed arcs $i \in I$. And for the case where k = 1 follows easily that it is a subnet with a controlled single-transition with one directed arc d and one directed arc i.

SIMULATING ZERO-TOKEN BEHAVIOR WITH THE VMSCS

Controlled D/PN does not detect zero-token in the places, but its enabling rule, under contact-free conditions, allow any VMSCS's to behave just like a PN with inhibitor arcs, making any VMSCS a partially Turing-machine.

In the modeling using PN, inhibitor arcs are the Turing-power modeling extension which allow this zero-testing; however, when it comes to model the fundamental VMSCS $\Sigma^{c(q)}$, the more controllers q exist in it, the higher the amount of necessary controlled transitions t_c as the result of all possible combinations between them and so it is for the amount of necessary arcs and specially inhibitor arcs. But for Controlled D/PN, the amount of arcs is less than in PN since it does not need inhibitor arcs.

For example, the net $\Sigma^{c(3)}$ shown in the Fig. 4a can be modeled with the PN of the Fig. 6. which have inhibitor arcs.

A visual comparison of the two models let us see it is more visually-friendly the model of Fig 4-a than the one in Fig. 6. A more formal comparison is the one for the amount of arcs between the VMSCS $\Sigma^{c(q)}$ with Controlled D/PN and its corresponding PN with inhibitor arcs is shown in the Fig. 7. where the exponential growth of the amount of arcs in the models using PN is faster compared with the ones using Controlled D/PN.

The enabling rule of Controlled D/PN satisfy our modeling necessities regarding the zero-testing in VMSCS's while preserving non-determinism between any other subnets. Each combinatorial amount of controllers in the VMSCS $\Sigma^{c(q)}$ is a subset of pre-conditions and they define a natural MSC-logic for all its controlled transitions based on the amount of incoming arcs.

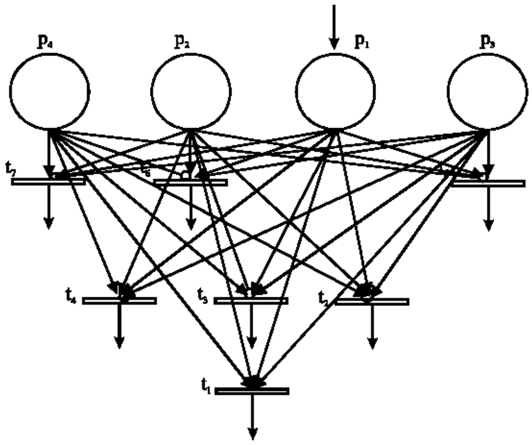

Fig. 6: A PN with inhibitor arcs, with same behavior of the VMSCS $\Sigma^{c(3)}$. The postset of output places can be arranged in any association

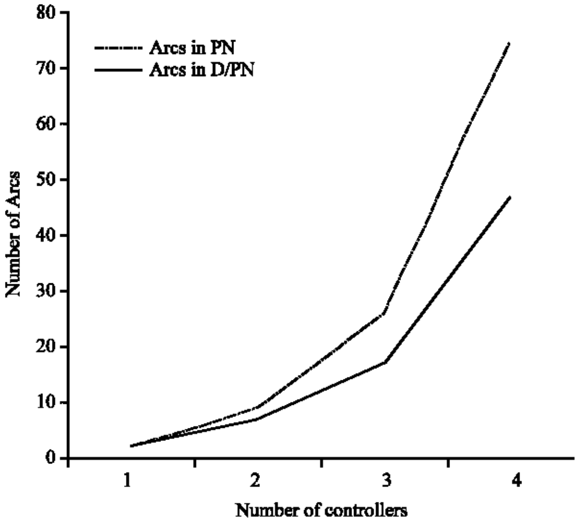

Fig. 7: Comparison between the amount of arcs for PN and D/PN with the behavior of the VMSCS $\Sigma^{c(q)}$

In the VMSCS $\Sigma^{c(q)}$, the modeler is not fully defining the transition that should fire, but instead, the enabling rule creates an enabling order overlooking the transitions according with their amount of incoming arcs and the selection of the transition to fire is determined just by the marking and control.

The VMSCS $\Sigma^{c(q)}$ is just a completely exhaustive combinatorial structure which allows the congruent MSC-logic when working with the disregarding zero-testing enabling rule of Controlled D/PN, behaving like a decision-free Petri net (df-net)[6] under contact-free conditions[7].

And the derivative VMSCS $\Sigma_n^{a(q)}$ with its incomplete structure is also a df-net as long as its trimming preserve same MSC-logic.

Property: The trimming of $\Sigma_n^{a(q)}$ preserve the MSC-logic of the VMSCS $\Sigma^{c(q)}$ when it is a progressive trimming which starts at the lowest control, i.e., n = 1.

From the results in[7] in the VMSCS $\Sigma^{c(q)}$, through the enabling rule we identify a cluster of df-nets in the form of siphons. The enabling starts from the highest control (the collective control, when n = k) and continue to lower controls (associative controls), until reaching the lowest control (individual control, when n = 1). Therefore any trimming which starts from the individual control and without skipping any control sequence in the trimming of higher controls, preserves the MSC-logic of the VMSCS $\Sigma^{c(q)}$, making the derivative VMSCS $\Sigma_n^{a(q)}$ still a cluster of df-nets in the form of siphons.

Property: The derivative VMSCS $\Sigma_n^{a(q)}$ is partially equivalent to a Turing machine.

A df-net has the power of a Turing machine and has the ability to detect zero-token in places. The net $\Sigma_n^{a(q)}$ is a clusters of df-nets and despite Controlled D/PN does not detect zero-token, the enabling rule allows this derivative VMSCS to simulate the same zero-token behavior under contact-free conditions, making them behave like Turing machine.

MSC DESIGN

Controlled D/PN was firstly created to explain MSC and its modeling using VMSCS's. However, the ultimate purpose is to design MSC using VMSCS's. In this section we present a simple control design for marking control using Controlled D/PN, which is an extension to the control design algorithm in[6], where necessary and sufficient conditions for reachability were obtained for a trap-containing PN (tc-net) Ë, if any fundamental circuit is a trap and:

- x is the minimal non-negative solution of the sate equation $m_f = m_o + Bx$.
- All siphons in Λx are marked at m_o.

Also sufficient conditions for reachability were obtained for trap-containing circuit PN (tcc-net) Ø, if any circuit in the net contains a trap and:

- x is the minimal non-negative solution of the sate equation $m_f = m_o + Bx$.
- All siphons in Øx contain marked trap at m_o.

Therefore, to define the MSC (the set of controllers and controls) that reaches a target marking m_f, our algorithm should be limited for the type of nets containing these conditions.

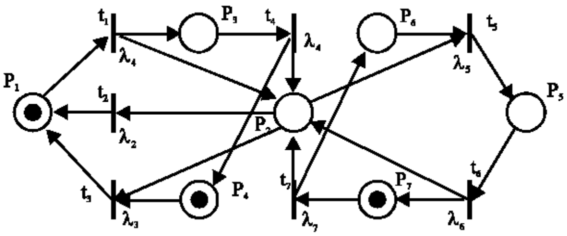

Fig. 8: A D/PN Model with Initial Marking in p_1, p_4 and p_7 and Enabled transitions t_1 and t_7

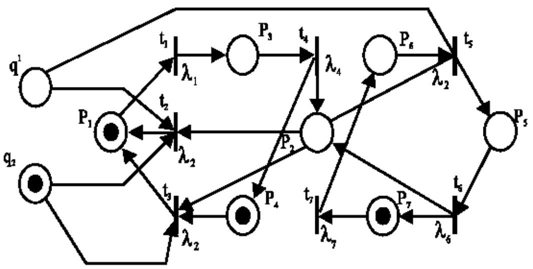

Fig. 9: A Controlled D/PN Model with Initial Marking in p_1, p_4 and p_7 and Marked Controller q_2

It will be for the bounded D/PN model presented in Fig. 8, where the initial condition is already known.

From the initial state m_0 = (1 0 0 1 0 0 1) we want to design a MSC to obtain a behavior that reaches the marking m_f = (0 1 1 1 0 1 0) in our model. The minimal non-negative solution of the sate matrix $m_f = m_0 + Bx$ is the vector x = (2 0 1 1 0 0 1) with a firing sequence satisfying the conditions stipulated before.

One difference in our algorithm with the one in[6] is when assigning the controllers. Instead of adding one controller to each fireable transition at m_f, first we identify from among those transitions the ones with the same input place p_i and associate them as subnets of controlled transitions by assigning one unique dan to each subnet. We count the number of controlled transitions in each subnet and from the symmetric series of $\Sigma^{c(q)}$ and $\Sigma_n^{a(q)}$, search for the number of transitions and select the one with the minimal number of controllers (for this example, our judgment is based in using as less as possible) and assign those controllers according with the selected VMSCS.

For this example we select the VMSCS $\Sigma^{c(2)}$, where two controllers can design this MSC for three controlled transitions. We connect the controllers q_1 and q_2 to the transitions t_1, t_2 and t_3 and fix the same dan for them in order to identify this VMSCS. Then we can obtain the Controlled D/PN model of the Fig. 9.

For the control, we deposit the corresponding token in the controllers as in[6], at each consecutively and corresponding discrete step to achieve our control. First we deposit one token at controller q_2 and let the model to fire freely. Uncontrolled transitions t_1 and t_7 fire, resulting in the marking (0 1 1 1 0 1 0). Then the control is activated, enabling the controlled transition t_3 to fire and resulting in the marking (1 0 1 0 0 1 0). Finally the transitions t_4 and t_1 fire consecutively, reaching the desired marking (0 1 1 1 0 1 0).

CONCLUSION

Controlled D/PN claims to reduce the amount of necessary controllers in the modeling and design of MSC using VMSCS's. It claims to provide the determinist congruency of a Turing machine for these VMSCS's. And finally, to create a more visually-friendly nets that could model the VMSCS's, by reducing the amount of necessary arcs for not using inhibitor arcs.

The two VMSCS's $\Sigma^{c(q)}$ and $\Sigma_n^{a(q)}$ can be seen as two types of scalable symmetries for VMSCS's, belonging to a systematized subnet structures or siphons notation for Controlled D/PN which model the proper behavior of MSC modules.

This research will continue in defining other families of derivative VMSCS's in order to extend the modeling scope for control with MSC subnets.

REFERENCES

1. Peterson, J.L., 1981. Petri Net Theory and the Modeling of Systems. Prentice-Hall, Englewood Cliffs, N.J.
2. Murata, T., 1989. Petri Nets: Properties, Analysis and Applications. In Proc.of the IEEE. IEEE, Piscataway N.J., 77: 541-580.
3. Halloway, L.E., B.H. Krogh and A. Giua, 1997. A survey of Petri net methods for controlled discrete event systems. In Discrete Event Dynamic Systems: Theory and Application. Kluwer Academic Publishers, Boston, 7: 151-190.
4. Moody, J.O. and P.J. Antsaklis, 1998. Supervisory Control of Discrete Event Systems Using Petri Nets. Kluwer Academic Publishers, Boston/Dordrecht London.
5. Krogh, B.H., 1987. Controlled Petri Nets and Maximal Permissive Feedback Logic. In Proceedings 25th Allerton Conference. University of Illinois, Urbana, (Sept).
6. Ichikawa, A. and K. Hiraishi, 1988. Analysis and Control of Discrete Event Systems Represented by Petri Nets. In Proceedings of the IIASA Conference on Discrete Event Systems (Sopron, Hungary, Aug. Springer-Verlag, LNCIS, 103: 115-134.
7. Jiménez, E., K. Araki and S. Kusakabe, 2006. Derivative Subnets of the Valid MSC Subnet $\Sigma^{c(q)}$ of Controlled Dan/Petri Nets. To appear in Proc. of the 2nd European Modeling Simulation Symposium-EMSS2006. (Barcelona, Spain).