2011 50th IEEE Conference on Decision and Control and
European Control Conference (CDC-ECC)
Orlando, FL, USA, December 12-15, 2011

# Colored Hybrid Petri-nets for modeling material handling systems

Francesco Basile, Pasquale Chiacchio, Jolanda Coppola

*Abstract*— **Material handling systems are usually modeled as discrete event systems. However, when the size of these systems grows, the overall system performances depend also on continuous time phenomena. We present a hybrid model based on a new Petri net formalism that merges the concepts of Hybrid Petri Nets and Colored Petri Nets to obtain compact models for these systems. An example is discussed in detail to motivate the introduction of a new formalism.**

## I. INTRODUCTION

Literature about Hybrid Petri Nets (HPNs) is wide: a their complete presentation is given in [1]; in [2] it is shown how HPNs can be used to describe a general hybrid system having jumps in the state space and switches in its dynamic.

Differential Petri Nets (DPNs) are introduced the first time in [3]; in these nets the marking of a differential place may be negative as well as the weights of arcs to or from a differential place. In [1], it has been shown how behavior of DPNs can be obtained using HPNs whose transitions firing speeds is a function of the net marking, and for this reason they are called Modified HPNs [1]. Then, it is not a limitation the use of no-negative markings and weights, as we do in this paper.

To model systems having first-order continuous behavior, which can be studied by linear algebraic tools, Balduzzi et al. introduce the First-Order Hybrid Petri Nets (FOHPNs) [4]. In FOHPNs continuous transition firing speeds are constant values, chosen by a control agent in a fixed range. When an event occurs, the net state changes, and a controller can decide to vary speed values, while between two event occurrences the firing speeds remain constant. In this paper firing speed values are not chosen in a fixed set but they are function of the marking of the net.

High-level HPNs are proposed in [5] and [6] to obtain compact hybrid models. Vectors of real numbers are used to represent the ordinary differential equations describing the continuous evolution of systems and structural individual tokens (colors) are used in the discrete part of the nets. In this paper, we use colors in both discrete and continuous part of the net, to enhance the compactness of the models.

In this paper Colored Modified Hybrid Petri Nets (CMHPNs) are introduced where no-negative marking and weights are used, firing speeds are not constant but they are linear functions of the continuous transitions input places marking of the net and no elements that extend expressive power of the net (e.g. inhibitor and test arcs as in [7]) are used. A similar approach is presented in [8]: a new kind of High-level Petri Nets, called Predicate Transition Nets (PTNs) is used to model batch systems, having several states, each one described by a particular set of equations. Every place of the PTN corresponds to a state of the system: if the place is marked then system is in the corresponding state. Its evolution is regulated by the set of equations associated to the enabled transitions acting on the marked place. A transition is enabled as long as the enabled function associated to it is verified. Values of state variables are reported in the marking of the place and they are call ”attributes” of the marking. Then, the net structure is used just to model the change in the differential equations of the system, while its evolution is described by the data structure associated to the places of the net. A colored version of PTNs is presented in [6]. The main difference between the formalism proposed in this paper and PTNs is that here the net structure completely describes the changes in the system differential equations of the hybrid system whose state is represented by the marking of the places, as well as the state evolution linked to the transition firing speeds.

Material handling systems are used to transfer something between two points along a path by a vehicle in any automated industrial plant as well as in automated warehouses. They are usually modeled by discrete event systems [9], [10]. In this framework each activity is modeled by a fixed time duration. When the spatial extension of these systems grows, their continuous time behaviors cannot be neglected. Indeed, a more precise information about the position/state of the vehicles becomes relevant. As for example, using discrete event system formalism like Petri Nets, a path is represented by a number of places. Such places model the presence of a vehicle in a certain zone. The exact position in the zone is unknown. A better precision requires many places. On the other hand, a continuous time system allows to represent the exact position but the mode changing in dynamic of vehicles (acceleration, deceleration or constant velocity) as well as the stop and go state of the vehicles (e.g. a vehicle stops when it reaches a certain position) would not be easily modeled. The simple case study of this preliminary work shows that CMHPNs are a promising tool to model complex material handling systems. In [11] an application of CMHPNs to real automated warehouse systems is discussed. Moreover, CMHPNs are a general hybrid system tool and then they can be used to obtain compact models of general hybrid systems. A control oriented simulation tool has been also developed for CMHPNs, see [12] for further details on simulation algorithm.

Francesco Basile, Pasquale Chiacchio and Jolanda Coppola are with Dip. Ing. Elettronica e Ing. Informatica, Università di Salerno, Italy.

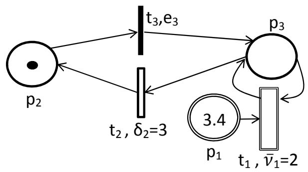



Fig. 1. A basic HPN.

## II. Background on Hybrid Petri Nets

A hybrid system is defined like a system consisting of a mixture of a continuous time system and a discrete event system (DES), having each one an own state space. These two systems are not independent but they influence each other. For the continuous time system, influence of DES results in abrupt changes in the dynamic and can occur either as switches in the vector field or as jumps in the state. Reversely, the continuous evolution influences the DES one by generating events that affect the discrete states [2].

A continuous system can be described by differential equations: $\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}(t), \boldsymbol{u}(t), t), \quad \boldsymbol{x}(0) = \boldsymbol{x}_0,\ \boldsymbol{y}(t) = \boldsymbol{g}(\boldsymbol{x}(t), \boldsymbol{u}(t), t)$ where $\boldsymbol{x} \in \mathbb{R}^n$ is the state vector, $\boldsymbol{u} \in \mathbb{R}^m$ is the input vector and $\boldsymbol{y} \in \mathbb{R}^r$ is the output vector. In particular if the interest is focused on the class of hybrid systems having autonomous commutations, i.e. systems for which changes in the dynamic occur if an analytical boundary condition about the instantaneous state value is reached, the equation $\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}(t), \boldsymbol{u}(t))$ with:

$$\boldsymbol{f} = \begin{cases} \boldsymbol{f}_1(\boldsymbol{x}(t), \boldsymbol{u}(t)) & for \quad \boldsymbol{h}(\boldsymbol{x}(t)) \leq 0 \\ \boldsymbol{f}_2(\boldsymbol{x}(t), \boldsymbol{u}(t)) & for \quad \boldsymbol{h}(\boldsymbol{x}(t)) > 0 \end{cases} \tag{1}$$

can be used, where it has been supposed the system can switch only between two possible dynamics ($\boldsymbol{f}_1$ and $\boldsymbol{f}_2$) and $\boldsymbol{h}$ is the boundary condition. For systems having linear, time-invariant, continuous part, like the ones we treat in this paper, each dynamic in (1) can be written as:

$$\boldsymbol{f}_i(\boldsymbol{x}(t), \boldsymbol{u}(t)) = \boldsymbol{A}_i \cdot \boldsymbol{x}(t) + \boldsymbol{B}_i \cdot \boldsymbol{u}(t) \tag{2}$$

where $\boldsymbol{A}_i$ is a constant n-order square matrix and $\boldsymbol{B}_i$ is a (n×m)-order matrix.

To model the hybrid systems behavior Hybrid Petri Nets (HPNs) can be used [2], [5], [1], [13].

In more general hybrid systems, switching between different dynamics is caused not only by the boundary conditions but also by external input events; in this case, the HPNs used to model the system behavior are said *synchronized*, as those used in this article. A HPN can be view as the combination of a "discrete" PN, [14], and a "continuous" PN.

In formal way, a HPN is a 7-tuple $\{P, T, \mathbf{Pre}, \mathbf{Post}, h, \boldsymbol{\delta}, \boldsymbol{\nu}\}$ such that: $P = P^D \bigcup P^C$, with $P^D \bigcap P^C = \emptyset$, where $P^D$ ($P^C$) is the set of $w_d$ discrete ( $w_c$ continuous) places, drawn like one (two) line circles; $T = T^D \bigcup T^C$, with $T^D \bigcap T^C = \emptyset$, where $T^D$ is the set of $n_d$ discrete transitions, which can be both immediate (drawn like black bars) and timed (drawn like white bars) and $T^C$ is the set of $n_c$ continuous transitions, drawn as a two lines boxes; $\mathbf{Pre} : P \times T \to \mathbb{R}^+$ is the pre-incidence matrix; $\mathbf{Post} : P \times T \to \mathbb{R}^+$ is the post-incidence matrix; $h : P \bigcup T \to \{D, C\}$, called "hybrid function", indicates for every node whether it is a discrete node (sets $P^D$ and $T^D$) or a continuous one (sets $P^C$ and $T^C$); $\boldsymbol{\delta} : T^D \to (\mathbb{R}^+)^{n_d}$ is the firing delay vector, whose element $\delta_i$ is the firing delay associated to each discrete transition $t_i^D$: if $\delta_i = 0$ then the transition $t_i^D$ is immediate, else if $\delta_i > 0$ then $t_i^D$ is timed. Function $\boldsymbol{\nu} : T^c \to (\mathbb{R}^+)^{n_c}$ is the firing speed vector. Note that in case of discrete nodes, $\mathbf{Pre}$ and $\mathbf{Post}$ assume integer positive values. The incidence matrix of the net is defined as $\boldsymbol{C} = \mathbf{Post} - \mathbf{Pre}$ and it can be written as the block matrix: $\boldsymbol{C} = \left(\begin{array}{c|c} \boldsymbol{C}_{CC} & \boldsymbol{C}_{CD} \\ \hline \boldsymbol{C}_{DC} & \boldsymbol{C}_{DD} \end{array}\right)$ where $\boldsymbol{C}_{CC}$ is the block regarding connections between continuous nodes, $\boldsymbol{C}_{DD}$ is the block regarding connections between discrete nodes, $\boldsymbol{C}_{CD}$ is the block regarding connections between continuous places and discrete transitions and $\boldsymbol{C}_{DC}$ is the block regarding connections between discrete places and continuous transitions.

HPN marking is a function $\boldsymbol{m} = \{\boldsymbol{m}^C, \boldsymbol{m}^D\}$, with $\boldsymbol{m}^C : P^C \to \mathbb{R}^+$, $\boldsymbol{m}^D : P^D \to \mathbb{N}$ that assigns to each discrete place a nonnegative integer number of tokens and to each continuous place a real number. The notation $\boldsymbol{m}(\tau_k)$ is used to denote the value of the marking of the net at the instant $\tau_k$. The marking of a place $p$ at a time $\tau_k$ is denoted by $m_p(\tau_k)$. The symbols ${}^\bullet p$ (${}^\bullet t$) and $p^\bullet$ ($t^\bullet$) are used for the *pre-set* and *post-set* of a place $p \in P$ (transition $t \in T$), respectively, e.g. ${}^\bullet t = \{p \in P \ |\mathbf{Pre}(p, t) > 0\}$.

A discrete transition $t^D$ is enabled at time $\tau_k$ if $m_p(\tau_k) \geq \mathbf{Pre}(p, t^D), \quad \forall p \in \ {}^\bullet t^D$. A transition $t^D$ can be either autonomous or synchronized to a logical expression that is function of a control event $g$ (associated to the occurrence of an external event) and/or of an internal condition $e$. Both $g$ and $e$ are boolean functions $g, e : T^D \to \{0, 1\}$. A discrete transition $t^D$ can fire if it is enabled and the associated logical expression is true. As for example, in a system formed by two masses traveling along a path, an internal condition is the reaching of a threshold distance that makes masses decelerate; an external control event for the same system is an asynchronous stop command arriving from an external controller; a logical expression can be the logic function OR between $g$ and $e$, e.g. $(g + e)$.

The continuous transition $t^C \in T^C$ is enabled at time $\tau_k$ if i) $m^D_{p^D}(\tau_k) \geq \mathbf{Pre}(p^D, t^C), \quad \forall p^D \in \ {}^\bullet t^C$ and ii) $m^C_{p^C}(\tau_k) \geq 0 \ \forall p^C \in \ {}^\bullet t^C$. At each continuous transition $t_i^C$ is associated the instantaneous firing speed (in the following also called simply firing speed) $\nu_i$: if $t_i^C$ is disabled $\nu_i = 0$; when $t_i^C$ is enabled $\nu_i$ is equal to the maximal firing speed $\bar{\nu}_i$, indicated near the transition. Notation $\boldsymbol{\nu}$ indicates the firing speed vector. The firing of continuous transitions cannot change the marking of discrete places, consequently $\boldsymbol{C}_{DC}(p^D, t^C) = 0, \ \forall p^D \in P^D$, thus $\boldsymbol{C}_{DC} = \boldsymbol{0}$. The evolution of the net can be described by its fundamental equation (written in a way pointing out the continuous part and the discrete part):

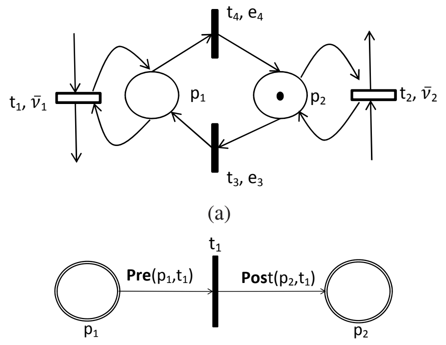



Fig. 2. Interaction between discrete and continuous nodes in HPNs: switch (a); jump (b).

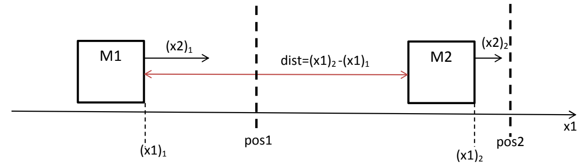



Fig. 3. System of the example of section III-A.

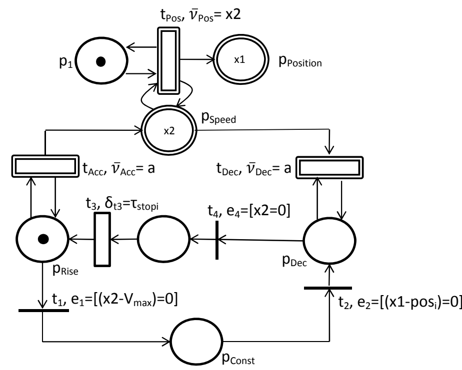



Fig. 4. A MHPN model of mass $i$ moving along a path.

$$\begin{bmatrix} \boldsymbol{m}^C(\tau_k) \\ \boldsymbol{m}^D(\tau_k) \end{bmatrix} = \begin{bmatrix} \boldsymbol{m}^C(\tau_{k-1}) \\ \boldsymbol{m}^D(\tau_{k-1}) \end{bmatrix} + \\ + \begin{bmatrix} \boldsymbol{C}_{CC} & \boldsymbol{C}_{CD} \\ \boldsymbol{0} & \boldsymbol{C}_{DD} \end{bmatrix} \left( \begin{bmatrix} \boldsymbol{0} \\ \boldsymbol{\sigma}(\tau_k) - \boldsymbol{\sigma}(\tau_{k-1}) \end{bmatrix} + \int_{\tau_{k-1}}^{\tau_k} \begin{bmatrix} \boldsymbol{\nu} \\ 0 \end{bmatrix} \right) \tag{3}$$

where $\boldsymbol{\sigma}(\tau_k) \in \mathbb{N}^n$ is the discrete firing vector whose component $\sigma_{t_i^D}(\tau_k)$ represents the number of times the discrete transition $t_i^D$ is fired up to the current time $\tau_k$.

A basic HPN is shown in Fig. 1, having:

- $P^C = \{p_1\}$, $P^D = \{p_2, p_3\}$;
- $T^C = \{t_1\}$, $T^D = \{t_2, t_3\}$ where $t_3$ is an immediate discrete transition, synchronized to the internal condition $e_3$ and $t_2$ is a discrete timed transition;
- $\delta = \{\delta_2\}$;
- $\boldsymbol{C} = \left( \begin{array}{c|cc} -1 & 0 & 0 \\ \hline 0 & 1 & -1 \\ 0 & -1 & 1 \end{array} \right)$.

Interactions between discrete and continuous nodes are shown in Fig. 2; a *switch* of dynamic, due to the occurring of a discrete event, is reported in Fig. 2a): the firing of $t1$ and $t2$ depends on the change in discrete places marking after the firing of the discrete transitions $t3$ and $t4$. Discrete transitions firing is synchronized to two generic internal conditions, $e_3$ and $e_4$. A *jump* in the continuous place marking is shown in Fig. 2b): when $t1$ fires, a quantity $\mathbf{Pre}(p1, t1)$ is taken from $m^C_{p_1}$ and the quantity $\mathbf{Post}(p2, t1)$ is added to $m^C_{p_2}$, producing a discontinuity in the marking trend.

For basic HPNs, the maximal firing speed of continuous transitions is a constant value, but powerful modifications have been proposed where continuous transition maximal firing speed is a function of the transition input places marking, of the input vector and of the time:

$$\nu_t(\tau) = f(\boldsymbol{m}(\tau), \boldsymbol{u}(\tau), \tau) \tag{4}$$

These kind of HPNs are called Modified HPNs (MHPNs).

## III. Colored Modified Hybrid Petri Nets

In this paper we use a MHPN where (4) is a linear function. In this way, systems having several dynamics as those described by the equations (2) can be modeled. Moreover, to compact the state representation, a structured individual token is used, as proposed in [5] and [6]. In addition, for the whole net, colors are used to define a more synthetic model of the systems. This new kind of net is named Colored MHPN (CMHPN).

### A. Example

Consider two unitary masses moving, without friction, along a path with a uniformly accelerated linear motion. Each mass state is described by position $x1$ and speed $x2$, related each other by the following equations:

$$\begin{pmatrix} \dot{x1} \\ \dot{x2} \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} x1 \\ x2 \end{pmatrix} + \begin{pmatrix} 0 \\ 1 \end{pmatrix} a \tag{5}$$

where $a$ is the constant acceleration. Assume the masses can accelerate until $(V_{max} - x2) = 0$, and then, they continue to move with constant speed, so (5) becomes:

$$\begin{pmatrix} \dot{x1} \\ \dot{x2} \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} x1 \\ V_{max} \end{pmatrix} \tag{6}$$

To avoid collisions, the masses regulate their speed in the manner that distance between them is equal or greater than a fixed threshold. Moreover, each mass has to start to decelerate if its position $x1$ is equal to a certain value $pos_i$. It decelerates until its speed $x2 = 0$ and then it stays for a time $\tau_{stopi}$, after that it starts to accelerate again only if the distance with the next mass is still greater than the threshold.

To model each mass behavior we can use the modified HPN shown in Fig. 4: marking of the place $p_{Positon}$ represents the actual mass position, while marking of the place $p_{Speed}$ represents its actual speed. When the mass accelerates (decelerates), the speed value is incremented (decremented) by the transition $t_{Acc}$ ($t_{Dec}$) with a firing speed just equal to the input value $a$. Position depends on the firing speed

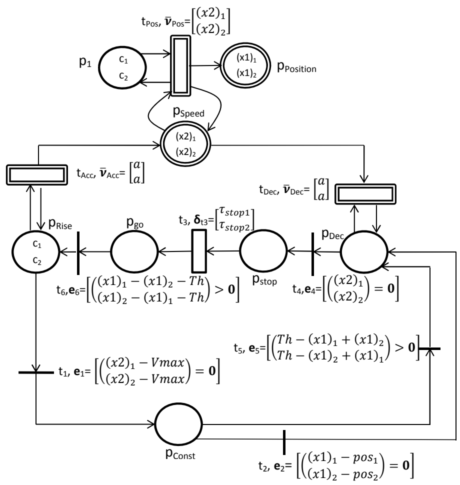



Fig. 5. The use of colors in a MHPN model of two masses moving along a path.

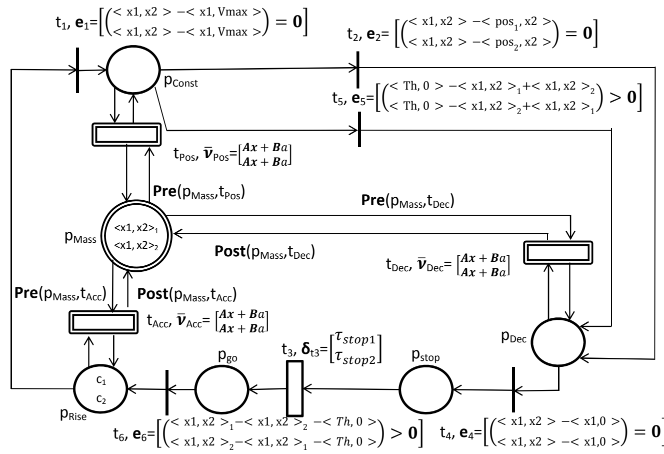



Fig. 6. A CMHPN model of two masses moving along a path.

of transition $t_{Pos}$, which is equal to $p_{Speed}$ marking. Note that when $m_{p_{Speed}} = 0$, $\nu_{Pos} = 0$ and, consequently, even if $tPos$ is still enabled, it does not change $m_{p_{Position}}$.

The whole system is modeled reproducing the net shown in Fig. 4 for each mass. To have a more compact representation we introduce colors, presented in [15], in the HPN model. We associate a different color to each mass, so the system can be modeled with just one net that evolves w.r.t. two colors. For the sake of clarity, in Fig. 5 the marking of a discrete place w.r.t. the color $i$ is indicated as $c_i$; the marking of the continuous places is indicated as $(x1)_i$ or $(x2)_i$ in the manner that its meaning is still obvious to the reader. Note that now firing speeds (both instantaneous and maximal), firing delays and logical expressions are column vectors, of dimension equal to the colors number. The $i$-th element of firing speeds (firing delays or logical expressions) vector associated to a continuous (discrete) transition is the firing speed (firing delay or logic expression) associated to the transition, w.r.t. the $i$-th color.

Moreover, two new discrete synchronized transitions, $t_5$ and $t_6$, have been added respect to the single mass model; these transitions manage the mass speed when the distance between the masses violates the threshold. Since the model represents the behavior of both the masses, thresholds violation can be managed as an internal condition associated to $t_5$ and $t_6$; with the uncolored model of Fig. 4 threshold violation can be detected only using an external controller that looking at the state of the two masses properly manages their speeds.

Finally, using a structured continuous marking, a more compact representation of the masses state can be obtained (see Fig. 6). The two state variables are collected in a vector which is the marking of the new place $p_{Mass}$, obtained by the fusion of $p_{Position}$ and $p_{Speed}$. There is a vector for each place color and we use the notation $< x1, x2 >_i = < \boldsymbol{x} >_i$ to indicate the structured marking w.r.t. the color $i$. Marking of discrete places is still represented by $c_i$.

Maximal firing speeds are still column vectors of two elements (one for each color of the net), but now, the $i$-th element is exactly $\boldsymbol{A} \cdot \boldsymbol{x} + \boldsymbol{B} \cdot a$, with $\begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} = \boldsymbol{A}$, $\begin{pmatrix} 0 \\ 1 \end{pmatrix} = \boldsymbol{B}$ and $\begin{pmatrix} x1 \\ x2 \end{pmatrix} = \boldsymbol{x}$; as for example, let $\tau_{k-1}$ be the previous observation instant of the net and $\tau_k$ be the current time, while transition $t_{Acc}$ is firing, the value of the structured marking elements, at the time $\tau_k$, will be

$$\begin{cases} x1(\tau_k) & = & x1(\tau_{k-1}) + \int_{\tau_{k-1}}^{\tau_k} (x2(\tau))d\tau \\ x2(\tau_k) & = & x2(\tau_{k-1}) + \int_{\tau_{k-1}}^{\tau_k} (a)d\tau \end{cases} \tag{7}$$

where with $xi(\tau_{k-1})$ we indicate the value of $xi$ at the instant $\tau_{k-1}$. Finally, note that the use of the structured marking requires the introduction of new arcs connecting continuous transitions with the place $p_{Mass}$ to modify separately masses position and speed, during the different dynamics. The separation of the effects is obtained using proper weights. For the sake of brevity only the expression of $\mathbf{Post}(p_{Mass}, t_{Acc})$ and $\mathbf{Pre}(p_{Mass}, t_{Acc})$ are reported:

$$\mathbf{Post}(p_{Mass}, t_{Dec}) = \begin{bmatrix} \boldsymbol{\pi}_1 & 0 \\ 0 & \boldsymbol{\pi}_1 \end{bmatrix},$$

$$\mathbf{Pre}(p_{Mass}, t_{Dec}) = \begin{bmatrix} \boldsymbol{\pi}_2 & 0 \\ 0 & \boldsymbol{\pi}_2 \end{bmatrix}$$

with $\boldsymbol{\pi}_1 = \begin{bmatrix} 1 & 0 \\ 0 & 2 \end{bmatrix}, \boldsymbol{\pi}_2 = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}$.

### *B. Formal definition*

A CMHPN is a four-tuple $\{\mathcal{N}, Cl, Co, \boldsymbol{\nu}\}$ where $\mathcal{N}$ is a HPN; $Cl$ is the set of colors. $Co$: $P \cup T \longrightarrow Cl$ is a color function that associates to each element in $P \cup T$ a set of colors; $\forall t_i^C \in T^C$, $\boldsymbol{\nu}$ is the mapping $Co(t_i^C) \rightarrow \mathbb{R}^+$ that associates an instantaneous firing speed to each continuous transition $t_i^c$. $\forall p_i \in P, Co(p_i) = \{a_{i,1}, a_{i,2}, ..., a_{i,u_i}\} \subseteq Cl$ is the set of possible colors of tokens in $p_i$, and $u_i$ is their number. $\forall t_j \in T, Co(t_j) = \{b_{j,1}, b_{j,2}, ..., b_{j,v_j}\} \subseteq Cl$ is the set of possible occurrence colors of $t_j$ and $v_j$ is their number. $\forall p_i^D \in P^D$, the marking $\boldsymbol{m}_{p_i^D}$ is defined as the mapping $Co(p_i) \rightarrow \mathbb{N}$ that associates to each possible token

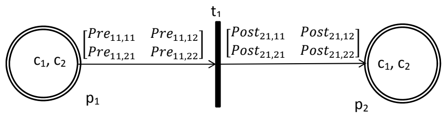



Fig. 7. Jumps in colored MHPNs.

color in $p_i^D$ a non-negative integer representing the number of tokens of that color that is contained in place $p_i^D$. For the sake of simplicity, a discrete marking w.r.t. color $r$ is indicated as $c_r$.

For each continuous place $p_i^C \in P^C$, the structured marking $\boldsymbol{m}_{p_i^C}$ is defined as the mapping $Co(p_i) \rightarrow (\mathbb{R}^+)^{(q)}$, thus, at each place $p_i^C \in P^C$, w.r.t. the color $r$, a vector of $q$ non-negative real numbers, $< x_1 \ldots x_q >_r$, is associated. The $q$ values of marking are called "attributes" and they completely describe the state of the system. For the sake of simplicity, a marking, w.r.t. the color $r$, with just one attribute is indicated as $c_r$.

For each continuous transition $t_i^C \in T^C$, $\boldsymbol{\nu}(t_i^C) = \boldsymbol{\nu}_i = \{\nu_{i,1}, \nu_{i,2}, \ldots, \nu_{i,v_c}\}$ is the vector of firing speeds of the continuous transition $t_i^C$. Its $r$-th element, $\nu_{i,r}$, is the firing speed of $t_i^C$ w.r.t. the color $r$ and, when $t_i^C$ is enabled, it is a linear function of the marking of the net. Similarly, $\forall t_i^D \in T^D$, $\boldsymbol{\delta}_i = (\delta_{i,1}, \ldots, \delta_{i,v_c})^T$ is the column vector of the discrete transition $t_i^D$ firing delays. The $r$-th element $\delta_{i,r}$ is the firing delay associated to the color $r$.

$\mathbf{Pre}(p_i, t_j)$ is a mapping $\mathbf{Pre}(p_i, t_j) : Co(t_j) \rightarrow \mathbb{R}^+(Co(p_i))$, for $i = 1, \ldots, m$ and $j = 1, \ldots, n$. At the same way we define $\mathbf{Post}(p_i, t_j)$ as the mapping $\mathbf{Post}(p_i, t_j) : Co(t_j) \rightarrow \mathbb{R}^+(Co(p_i))$, for $i = 1, \ldots, m$ and $j = 1, \ldots, n$. $\mathbf{Pre}(\mathbf{Post})(p_i, t_j)$ is a matrix of dimensions $u_i \times v_j$; the element $Pre(p_i, t_j)(r, s) = Pre_{ij,rs}$ $(Post(p_i, t_j)(r, s) = Post_{ij,rs})$ is the weight of the arc connecting $p_i$ $(t_i)$ w.r.t. the color $r$ (color $s$) to $t_j$ $(p_i)$, w.r.t. the color $s$ (color $r$). The nature of the element depend on the kind of nodes it connects, e.g. weighs of arcs connecting transitions to discrete places are non-negative integer numbers, while weighs of arcs connecting transitions to continuous places are row vectors of non-negative real numbers, with dimension equal to the attributes token number. When all the elements of the matrix are equal to 1, weights are not reported near the arcs. The incidence matrix $\boldsymbol{C}$ can be written as shown in section II). Elements of $\boldsymbol{C}$ are the matrices $\boldsymbol{C}(p_i, t_j) = \mathbf{Post}(p_i, t_j) - \mathbf{Pre}(p_i, t_j)$ with dimension $u_i \times v_j$.

An example of jump in CMHPN is shown in Fig. 7: for the sake of simplicity it has been shown only the jump with marking having just one attribute but also jumps with $q > 1$ are possible.

## IV. Case Study

Material handling systems, in general, consists of vehicles that are able to move something along a path, from a source to a destination point, avoiding collisions. Starting from this consideration, in this section to show why CMHPNs are a promising tool to model material handling systems, a simple example is discussed.

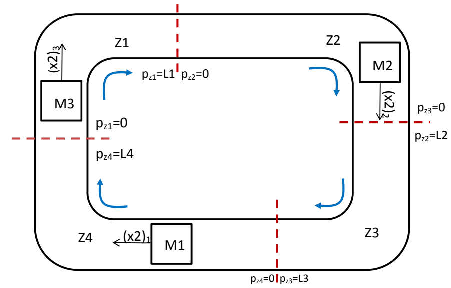



Fig. 8. Three masses moving along a circuit.

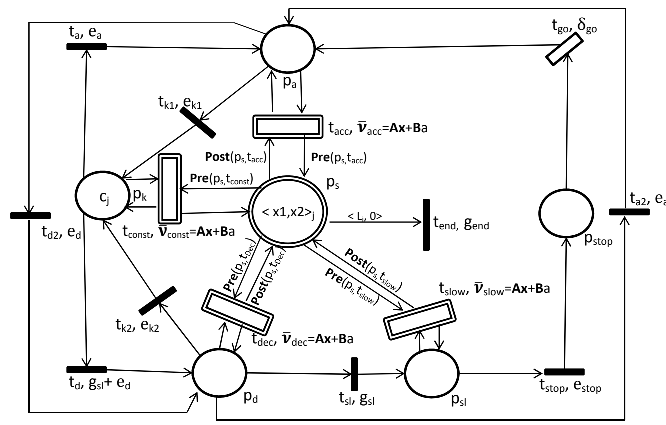



Fig. 9. Colored HPN modeling 3 masses moving along a circuit.

Consider 3 unitary masses, named $M_1$, $M_2$ and $M_3$, moving along a ring as shown in Fig. 8, without friction, with initially speed $V_{max}$, that can be varied with constant acceleration, $a$. The ring is divided in four zones, each one with the own length $L_i$. Each mass $M_j$ moves along the path until an external controller orders it to slow down so that $M_j$ can stop at a desired position $x_{d_i}$. Here the mass stays for a time $\delta_{stop}$ after that it starts to move again. Note that masses do not know neither where nor when they will have to stop. Moreover, a fixed distance between two consecutive masses has to be maintained. The net modeling this system is shown in Fig. 9. The colored structured continuous place $p_s$ represents the state (position $x1$ and speed $x2$) of each mass $M_j$. Evolution of the masses state is due to the firing of the colored continuous structured transitions $t_{acc}, t_{dec}$ and $t_{slow}, t_{const}$ that are enabled if masses are accelerating, decelerating or moving with constant speed, respectively; discrete immediate transitions model the mass dynamic switches: for the sake of clarity, in Fig. 9, the expressions of the internal conditions are not reported, but they can be read in the table I. Mass stops are managed by the external controller by means of the external event $g_{sl}$: if controller decides the mass $M_j$ has to stop at $x_{d_i}$, when the mass arrives next to a particular position (called *slow down* point and given, from time to time, according to $M_j$ speed), it sets $g_{sl} = 1$. Consequently a change in the $M_j$ dynamic occurs and the mass starts to decelerate up to arrive at $x_{d_i}$ with $x2 = 0$. This triggers the firing of the immediate transition $t_{stop}$. The timed transition

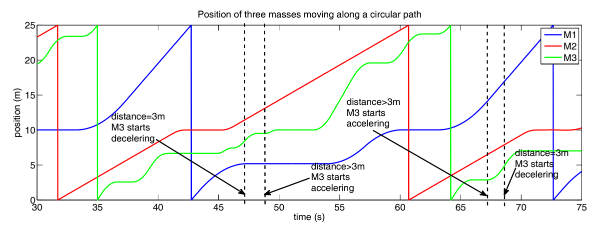
TABLE I

INTERNAL CONDITIONS AND THEIR MEANING.

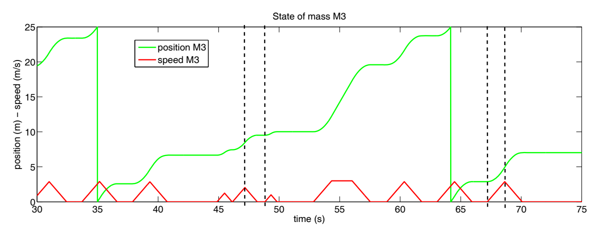
| Internal Condition | Expression of the $j$-th element | Meaning |
|---|---|---|
| $\boldsymbol{e}_{k1}$ | $\left[ (< x_1, x_2 >_j - < x_1, V_{max} >_j) = \mathbf{0} \vee (< x_1, x_2 >_j - < x_1, V_{in} >_j) = \mathbf{0} \right]$ | mass speed is equal to $V_{max}$ OR it is equal to the desired value |
| $\boldsymbol{e}_{k2}$ | $\left[(< x_1, x_2 >_j - < x_1, x_2 >_k - < Th, 0 >) \geq \mathbf{0} \wedge ((< x_1, x_2 >_j - < x_1, 0 >_j) = \mathbf{0} \vee (< x_1, x_2 >_j - < x_1, V_{in} >_j)) = \mathbf{0} \right]$ | distance between mass j and k is $\geq$ than the threshold $Th$ AND (mass speed is $= \mathbf{0}$ OR mass speed is equal to a desired value) |
| $\boldsymbol{e}_a$ | $\left[ ((< x_1, V_{max} >_j - < x_1, x_2 >_j) \gneq \mathbf{0} \vee (< x_1, V_{in} >_j - < x_1, x_2 >_j) \gneq \mathbf{0}) \wedge ((< x_1, x_2 >_j - < x_1, x_2 >_k - < Th, 0 >) \gneq \mathbf{0}) \right]$ | (speed value is lower than $V_{max}$ OR it is lower than the desired value) AND distance between mass j and k is equal to or greater than the threshold $Th$ |
| $\boldsymbol{e}_d$ | $\left[ (< x_1, x_2 >_j - < x_1, 0 >_j) \gneq \mathbf{0} \wedge ((< x_1, x_2 >_j - < x_1, V_{in} >_j) \gneq \mathbf{0} \vee (< Th, 0 > - < x_1, x_2 >_j + < x_1, x_2 >_k) \gneq \mathbf{0}) \right]$ | (speed value is greater than 0) AND (speed is greater than the desired value OR the distance with the next mass is lower than the threshold $Th$) |
| $\boldsymbol{e}_{stop}$ | $\left[ (< x_1, x_2 >_j - < x_1, 0 >_j) = \mathbf{0} \right]$ | mass speed is $= 0$. |

$t_{go}$, with firing delay $\delta_{go}$, models the steady of the mass at $x_{d_i}$. Finally, the entering of the masses into a new zone is synchronized on an external control event: when a mass arrives at the end of $Z_i$, $g_{end}$ is set 1 by the controller and the immediate transition $t_{end}$ fires; consequently, a jump in the state mass occurs and position value changes from $x1 = L_i$ to $x1 = 0$.

Results of the case study simulation are reported in Fig. 10, where the switch of dynamic, due to the violation of the threshold distance between $M3$ and $M2$ is shown: when the distance goes under $3m$, $M_3$ starts decelerating; speed begins to rise again only when the distance between the two masses is up of $3m$. Vertical lines in position trend correspond to the jumps in the state at the begin of a new zone. Note that the model complexity is independent from the length of each zone. In a classical discrete event approach a very large number of places would be needed to describe the position of the masses with a reasonable precision. In a classical continuous time approach the dynamic changes would not be easily represented.




(a)




(b)

Fig. 10. Results of the simulation of case study, with $L_i = 25m\ \forall i$, $V_{max} = 3m/s$ and $a = 2m/s^2$: a) masses position evolution; b) position and speed of mass $M_3$.

## REFERENCES

[1] R. David and H. Alla, *Discrete, Continuous and Hybrid Petri Nets*, Springer, Ed., 2005.

[2] S. Pettersson and B. Lennartson, "Hybrid modelling focused on hybrid Petri nets," in *in 2nd European Workshop on Real-time and Hybrid systems*, 1995, pp. 303–309.

[3] I. Demongodin and N. Koussoulas, "Differential Petri nets: representing continuous systems in a discrete-event world," *Automatic Control, IEEE Transactions on*, vol. 43, no. 4, pp. 573 –579, Apr. 1998.

[4] F. Balduzzi, A. Giua, and G. Menga, "First-order hybrid Petri nets: a model for optimization and control," *Robotics and Automation, IEEE Transactions on*, vol. 16, no. 4, pp. 382 –399, Aug. 2000.

[5] A. Giua and E. Usai, "Modeling hybrid system by high-level Petri nets," *API - JESA*, vol. 32, no. 9, pp. 9–10, 1998.

[6] R. Champagnat, H. Pingaud, and R. Valette, "An extension of high-level Petri nets for modelling batch systems," in *IMACS/IEEE CSCC*, 1999, pp. 2621–2625.

[7] O. Lejri and M. Tagina, "Hybrid reconfigurable Petri nets for modelling hybrid reconfigurable manufacturing systems," *Journal of Studies on Manufacturing*, vol. 1, no. 2–3, pp. 75–84, 2010.

[8] R. Champagnat, P. Esteban, H. Pingaud, and R. Valette, "Modelling hybrid systems by means of high-level Petri nets: benefits and limitations," in *IFAC-IFIP-IMACS conference on Control of industrial systems*, vol. 1, May 1997, pp. 469–474.

[9] F. Amato, F. Basile, C. Carbone, and P. Chiacchio, "An approach to control automated warehouse systems," *Control Eng. Practice*, vol. 13, pp. 1223–1241, 2003.

[10] M. Dotoli and M. Fanti, "A coloured Petri net model for automated storage and retrieval systems serviced by rail-guided vehicles: A control perspective." *International Journal of Computer Integrated Manufacturing*, vol. 18, no. 2, pp. 122–136, 2005.

[11] F. Basile, P. Chiacchio, and J. Coppola, "A hybrid model for the control and the analysis of complex automated warehouse systems," $16^{th}$ *IEEE Conf. on Emerging Technologies and Factory Automation (ETFA 2011), Toulouse, France*, 2011.

[12] "Pnetlab," by Automatic Group of Salerno, http://www.automatica.unisa.it/PnetLab.html.

[13] M. Drighiciu, A. Petrisor, and M. Popescu, "A Petri nets approach for hybrid systems modeling," *International Journal of Circuits, Systems and Signal Processing*, 2009.

[14] T. Murata, "Petri nets: Properties, analysis and applications," *Proceedings of IEEE*, vol. 77, no. 4, pp. 541–580, April 1989.

[15] K. Jensen, *Colored Petri nets. Basic concepts, analysis methods and pratical use. Volume 1*, ser. Monographs on Theoretical Computer Science. New York: Springer Verlag, 1995.