Modular/Coordinated control for TCPNs ⋆

C. Renato Vázquez, * **Jan H. van Schuppen,** ** **Manuel Silva** *

* *Dep. de Informática e Ingeniería de Sistemas, CPS, Universidad de Zaragoza, María de Luna 1, E-50018 Zaragoza, Spain (e-mail: {cvazquez,silva}@unizar.es)*
** *Centrum voor Wiskunde en Informatica, NL-1090 GB Amsterdam, The Netherlands (e-mail: J.H.van.Schuppen@cwi.nl)*

Abstract: Timed continuous Petri net ($TCPN$) systems are piecewise-linear models with input constraints that can approximate the dynamical behavior of a class of timed discrete-event systems. This work is devoted to the synthesis of a coordinated control strategy for large $TCPN$ systems that can be seen as a set of T-disjoint $TCPN$ modules interconnected by places. The goal of the control scheme is to transfer the system in such a way that each module reaches a desired marking. The resulting scheme consists of a set of affine local controllers, one per module, and a coordinator that receives and sends information to the local controllers.

1. INTRODUCTION

Fluid Petri nets are continuous relaxations of (discrete) Petri nets. These relaxations are models that can be analyzed by using techniques from both Petri nets and control theories, overcoming the state explosion problem that frequently appears in discrete-event systems (DES). Different approaches to fluid Petri net models can be found in the literature (for instance, Alla and David [1998], Silva and Recalde [2002]). In this work, timed continuous Petri net ($TCPN$) models under infinite server semantics are considered. Mahulea, Recalde and Silva [2006] showed that, frequently, this particular semantics provides a better approximation to the average behavior of the original discrete Petri net. The continuous model thus obtained has three main characteristics: 1) it is piecewise-linear, 2) the input must be nonnegative and upper bounded by a piecewise-linear function of the state, and 3) models with a real meaning may be high-order systems (with tens or even hundreds of state-variables).

Regarding control in fluid Petri nets, different authors have proposed control techniques ranging from fuzzy logic control (Hennequin, Lefebvre and El-Moudni [1999]), feedback control synthesis based on linear matrix inequalities (Kara et al. [2009]), model predictive control (Mahulea et al. [2008]), gradient-based controllers (Lefebvre et al. [2007]), etc. In all of those cases a centralized controller is synthesized, mostly assuming that all the transitions are controllable, in order to drive the system towards a desired target marking (a steady state), by means of modifying (reducing) the transition's flow (i.e., the speed at which the transitions fire). This control objective is similar to a classical set-point control problem in continuous-state systems (see, for instance, Chen [1984]), and it is not equivalent to fulfilling safety specifications which is commonly addressed by DES's controllers. Enforcing a desired target marking in the continuous Petri net is equivalent to reaching an average marking in the original discrete model (assuming that the continuous model approximates the discrete one), which may be interesting in several kind of systems. This idea has been illustrated by Amrah, Zerhouni and El-Moudni [1997] for deterministically timed models of manufacturing lines, and by Vázquez and Silva [2009] for the stock-level control in an automotive assembly line.

In this work, an affine centralized control law for $TCPN$ systems is firstly derived. Later, a coordinated control strategy is proposed for a *modular* view of $TCPN$s (a $TCPN$ system is considered as composed of several T-disjoint $TCPN$s, named *modules*, interconnected by places, named *buffers*), in order to reduce the complexity involved in the synthesis. Recently, a distributed control strategy has been proposed for this problem (Apaydin et al. [2010]), assuming particular PN subclasses for the modules. In this work, a coordinated control scheme is introduced for general subclasses of Petri nets. The resulting scheme consists of a set of *local* controllers, each one synthesized for each module, and a *coordinator* that receives and sends information to the local controllers.

This work is organized as follows: in Section 2, basic concepts and definitions on timed continuous Petri nets are introduced. Later, some results regarding affine control laws, that constitute the base for the local controllers, are recalled from the work of Habets and van Schuppen [2004] in Section 3. The contributions of this work are presented in Section 4 and 5. Section 4 is devoted to the synthesis of centralized controllers for $TCPN$s. In Section 5, those results are extended in order to proposed a coordinated control strategy for modular continuous Petri nets. Finally, some conclusions are given in Section 6.

2. CONTINUOUS PETRI NETS

In the sequel, given a matrix $\mathbf{A}$ and a set of indices $I = \{i_1, ..., i_n\}$ and $J = \{j_1, ..., j_m\}$, it will be denoted

⋆ This work was partially supported by project CICYT and FEDER DPI2006-15390.
The research leading to these results has received funding from the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement no 224498.

as $\mathbf{A}[I, J]$ the matrix built with the elements in the rows indicated by I and the columns indicated by J.

Definition 1. A *continuous PN* is defined having a structure $\mathcal{N} = \langle P, T, \mathbf{Pre}, \mathbf{Post}\rangle$ like in discrete PNs, i.e., P and T are finite disjoint sets of places and transitions, respectively, $\mathbf{Pre}$ and $\mathbf{Post}$ are $|P| \times |T|$ sized, natural valued, *pre- and post- incidence matrices*. The difference is in the evolution rule: in continuous PNs, the firing is not restricted to integer amounts, and so the marking $\mathbf{m}$ is not forced to be integer. Instead, a transition t_i is *enabled* at $\mathbf{m}$ iff for every $p_j \in {}^\bullet t_i$, $\mathbf{m}[p_i] > 0$; and its *enabling degree* is $enab(t_i, \mathbf{m}) = \min_{p_j \in {}^\bullet t_i}\{\mathbf{m}[p_j]/\mathbf{Pre}[p_j, t_i]\}$. The firing of t_i in a certain amount $\alpha \leq enab(t_i, \mathbf{m})$ leads to a new marking $\mathbf{m'} = \mathbf{m} + \alpha \cdot \mathbf{C}[P, t_i]$, where $\mathbf{C} = \mathbf{Post} - \mathbf{Pre}$ is the token-flow matrix.

Right and left rational annulers of $\mathbf{C}$ are called T- and *P-flows*, respectively. If there exists $\mathbf{y} > \mathbf{0}$ ($\mathbf{x} > \mathbf{0}$) s.t. $\mathbf{y}^T\mathbf{C} = \mathbf{0}$ ($\mathbf{Cx} = \mathbf{0}$), the net is said to be *conservative* (*consistent*). Bases for T- and *P-flows* are denoted as $\mathbf{B}_x$ and $\mathbf{B}_y$, respectively. A set of places Σ is a *siphon* iff ${}^\bullet\Sigma \subseteq \Sigma^\bullet$ (i.e., the set of input transitions is included in the set of output transitions). For reachability, the limit concept is used, and a marking reached in the limit of an infinitely long sequence is considered reachable.

Definition 2. A *Timed Continuous Petri Net* (TCPN) system is the tuple $\langle \mathcal{N}, \boldsymbol{\lambda}, \mathbf{m}_0\rangle$, consisting of a continuous PN $\mathcal{N}$, a vector $\boldsymbol{\lambda} \in \mathbb{R}^{|T|}_{>0}$ and the initial marking $\mathbf{m}_0$. Here, *infinite server semantics* (or *variable speed*, see Silva and Recalde [2002]) is considered. Accordingly, if a transition t_i has an enabling degree $enab(t_i, \mathbf{m}) > 0$, then it fires with a speed, named *flow*, equal to $\mathbf{f}(\mathbf{m})[t_i] = \boldsymbol{\lambda}[t_i] \cdot enab(t_i, \mathbf{m}) = \boldsymbol{\lambda}[t_i] \cdot \min_{p_j \in {}^\bullet t_i}\{\mathbf{m}[p_j]/\mathbf{Pre}[p_j, t_i]\}$, where $\boldsymbol{\lambda}[t_i]$ is a constant value that denotes the *rate* of t_i.

For the flow to be well defined, we will assume that $\forall t \in T, |{}^\bullet t| \geq 1$. The "*min*" in the flow definition leads to the concept of *configurations*: a configuration assigns to each transition one place that for some markings will control its firing rate (i.e., it is constraining that transition). The number of configurations is upper bounded by $\prod_{t \in T} |{}^\bullet t|$.

The flow through the transitions can be written as $\mathbf{f}(\mathbf{m}) = \boldsymbol{\Lambda}\boldsymbol{\Pi}(\mathbf{m})\mathbf{m}$, where $\boldsymbol{\Lambda}$ is a diagonal matrix whose elements are those of $\boldsymbol{\lambda}$, and $\boldsymbol{\Pi}(\mathbf{m})$ is the configuration operator matrix at $\mathbf{m}$, defined by elements as

$$\boldsymbol{\Pi}(\mathbf{m})[i, j] = \begin{cases} \dfrac{1}{\mathbf{Pre}[p_j, t_i]}, & \text{if } p_j \text{ is constraining } t_i, \\ 0, & \text{otherwise} \end{cases}$$

If more than one place is constraining the flow of a transition, any of them can be used, but only one is taken (let us adopt the convention of taking the place with the lowest index).

Control actions may only be a reduction of the flow through the transitions. That is, transitions (machines for example) cannot work faster than their nominal speed. Transitions in which a control action can be applied are called *controllable*. The effective flow through a transition that is being controlled can be represented as: $\mathbf{w}_i(\tau) = \boldsymbol{\lambda}[t_i] \cdot enab(\tau)\,[t_i] - u(\tau)[t_i]$, where $0 \leq u(\tau)[t_i] \leq \boldsymbol{\lambda}[t_i] \cdot enab(\tau)[t_i]$. The control vector $\mathbf{u} \in \mathbb{R}^{|T|}$ is defined s.t. $\mathbf{u}[i]$ represents the control action on t_i. If t_i is not controllable then $\mathbf{u}[i] = 0$. The forced flow vector is expressed as $\mathbf{w}(\mathbf{m}, \mathbf{u}) = \boldsymbol{\Lambda}\boldsymbol{\Pi}(\mathbf{m})\mathbf{m} - \mathbf{u}$. The set of all controllable transitions is denoted by T_c, and the set of uncontrollable transitions is $T_{nc} = T - T_c$.

The behavior of a TCPN forced system is described by the state equation:

$$\begin{aligned} \overset{\bullet}{\mathbf{m}} &= \mathbf{C}\boldsymbol{\Lambda}\boldsymbol{\Pi}(\mathbf{m})\mathbf{m} - \mathbf{C}\mathbf{u} \\ \mathbf{0} &\leq \mathbf{u} \leq \boldsymbol{\Lambda}\boldsymbol{\Pi}(\mathbf{m})\mathbf{m} \end{aligned} \tag{1}$$

A control action that fulfills the required constraints, i.e., $\mathbf{0} \leq \mathbf{u} \leq \boldsymbol{\Lambda}\boldsymbol{\Pi}(\mathbf{m})\mathbf{m}$, is called *suitably bounded* (s.b.). If an input is not s.b. then it cannot be applied.

In this work, it is assumed that all the transitions are controllable, so $T_c = T$. In this case, (1) can be seen as a linear system without state-feedback, i.e.,

$$\begin{aligned} \overset{\bullet}{\mathbf{m}} &= \mathbf{C}\mathbf{w} \\ \mathbf{0} &\leq \mathbf{w} \leq \boldsymbol{\Lambda}\boldsymbol{\Pi}(\mathbf{m})\mathbf{m} \end{aligned} \tag{2}$$

where $\mathbf{w}$ represents the control action. The constraint $\mathbf{0} \leq \mathbf{w} \leq \boldsymbol{\Lambda}\boldsymbol{\Pi}(\mathbf{m})\mathbf{m}$ is equivalent to $\mathbf{0} \leq \mathbf{u} = \mathbf{f}(\mathbf{m}) - \mathbf{w} \leq \boldsymbol{\Lambda}\boldsymbol{\Pi}(\mathbf{m})\mathbf{m}$, i.e., $\mathbf{u}$ is s.b..

A marking $\mathbf{m}$, for which $\exists\mathbf{u}$ s.b., s.t. $\dot{\mathbf{m}} = \mathbf{C}(\boldsymbol{\Lambda}\boldsymbol{\Pi}(\mathbf{m})\mathbf{m} - \mathbf{u}) = \mathbf{0}$ is called *equilibrium marking*, and $\mathbf{u}$ and $\mathbf{w}(\mathbf{m}, \mathbf{u})$ are said to be its *equilibrium input* and *flow*, respectively.

2.1 Controllability on continuous Petri nets

It is well known that *P-flows* in PNs induce linear dependencies in the marking of the places (for any reachable marking $\mathbf{m}$, $\mathbf{y}^T\mathbf{m} = \mathbf{y}^T\mathbf{m}_0$), meaning state invariants. Therefore, systems with P-flows are not controllable in the classical sense. In the sequel, this state invariant will be denoted as $Class(\mathbf{m}_0) = \{\mathbf{m} \geq \mathbf{0} | \mathbf{B}_y^T\mathbf{m} = \mathbf{B}_y^T\mathbf{m}_0\}$. Every reachable marking belongs to $Class(\mathbf{m}_0)$, but the reverse is not true for timed models.

Due to the presence of state invariants and input constraints, Vázquez et al. [2008] proposed an adaptation of the classical controllability definition.

Definition 3. The $TCPN$ system $\langle \mathcal{N}, \boldsymbol{\lambda}, \mathbf{m}_0\rangle$ is *controllable* with bounded input (BIC) over $S \subseteq Class(\mathbf{m}_0)$ if for any $\mathbf{m}_1, \mathbf{m}_2 \in S$ there exists an input $\mathbf{u}$ that transfers the system from $\mathbf{m}_1$ to $\mathbf{m}_2$ in finite or infinite time, and it is suitably bounded along the marking trajectory.

In case that all the transitions are controllable, the controllability property can be decided by the structure and initial marking, something that can be checked in polynomial time. Recalling from (Vázquez et al. [2008]),

Theorem 1. The $TCPN$ system $\langle \mathcal{N}, \boldsymbol{\lambda}, \mathbf{m_0}\rangle$ is BIC over the interior of $Class(\mathbf{m}_0)$ iff $\mathcal{N}$ is consistent. It is BIC over the complete $Class(\mathbf{m}_0)$ iff the net is consistent and there do not exist empty siphons at any marking in $Class(\mathbf{m_0})$.

3. AFFINE CONTROL LAWS FOR SIMPLICES

A $TCPN$ can be seen as a particular class of piecewise-affine hybrid system with additional input constraints. In this work, the techniques introduced by Habets et al. [2006], regarding the control synthesis of such hybrid systems, will be extended for $TCPN$ models. For this purpose, let us recall a few results through this section.

A *polyhedral* set is a subset of $\mathbb{R}^m$, described by a finite number of linear inequalities. A bounded polyhedral set is called a *polytope*. Alternatively, a polytope can be characterized as the convex hull of a finite number of points: the *vertices* of the polytope. A *face* of a polyhedral set is the intersection of the set with one of its supporting hyperplanes. If a polyhedral set $\mathcal{P}$ has dimension m, the faces of $\mathcal{P}$ of dimension $m-1$ are called *facets*. A description of several problems encountered during the synthesis and analysis in polyhedrals and several algorithms for their resolution, including the computation of the vertices, are provided by Fukuda [2000]. An m-dimensional polytope with exactly $m+1$ vertices is called a *simplex*. The number of facets in a simplex is equal to the number of its vertices, i.e., $m+1$.

Definition 4. An *affine system* in a polytope $\mathcal{X}$ is:

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} + \mathbf{a} \tag{3}$$

with the restrictions $\mathbf{x} \in \mathcal{X}$ and $\mathbf{u} \in \mathcal{U}$, where $\mathcal{U}$ is a polytope of admissible inputs.

An admissible *affine control law* is an affine function $\mathbf{u} : \mathcal{X} \to \mathcal{U}$ characterized by $\mathbf{u}(\mathbf{x}) = \mathbf{F}\mathbf{x} + \mathbf{g}$.

3.1 Synthesis of an affine control law

Here, a couple of conditions, regarding the control synthesis for affine systems (3), will be introduced.

Let $\mathcal{S}$ denote a closed full-dimensional simplex in $\mathbb{R}^m$ with vertices $\mathbf{v}_1, ..., \mathbf{v}_{m+1}$. Let $F_1, ..., F_{m+1}$ denote the facets of S, and assume that the facets are numbered in such a way that for $i = 1, .., m+1$, $\mathbf{v}_i$ is the only vertex not belonging to facet F_i. For $i = 1, .., m+1$, let $\mathbf{n}_i$ denote the outward unit normal vector of facet F_i. Considering an affine control law $\mathbf{u}(\mathbf{x}) = \mathbf{F}\mathbf{x} + \mathbf{g}$, the evaluation of this at the vertices of S will be denoted as $\mathbf{u}_j = \mathbf{u}(\mathbf{v}_j)$, $\forall j \in \{1, ..., m+1\}$.

Due to the linearity of both the system and the control law inside $\mathcal{S}$, the evaluation of the control law at the vertices of $\mathcal{S}$ must fulfill

$$[\mathbf{F}, \mathbf{g}] \cdot \begin{bmatrix} \mathbf{v}_1 & ... & \mathbf{v}_{m+1} \\ 1 & ... & 1 \end{bmatrix} = [\mathbf{u}_1, ..., \mathbf{u}_{m+1}] \tag{4}$$

Since $\mathcal{S}$ is a full-dimensional simplex then the matrix $\begin{bmatrix} \mathbf{v}_1 & ... & \mathbf{v}_{m+1} \\ 1 & ... & 1 \end{bmatrix}$ is square and has full rank. Therefore, there is a bijection between $(\mathbf{F}, \mathbf{g})$ and the set of values $\mathbf{u}_j$ (with $j \in \{1, ..., m+1\}$). In the sequel, for the synthesis of a control law, the values of the input $\mathbf{u}_j$ at the vertices are first computed. Once these are obtained, the pair $(\mathbf{F}, \mathbf{g})$ can be computed by using (4).

Consider the following conditions for the values $\mathbf{u}_j$:

Condition 1. From Theorem 4.12 in Habets et al. [2006], a facet F_i is *disabled* by the control action $\mathbf{u}(\mathbf{m}) = \mathbf{F}\mathbf{x} + \mathbf{g}$, whose valuation at vertices $\mathbf{v}_1, ..., \mathbf{v}_{m+1}$ is given by $\mathbf{u}_1, ..., \mathbf{u}_{m+1}$, iff

$$\forall j \in \{1, ..., m+1\} \backslash \{i\} \quad \mathbf{n}_i^T(\mathbf{A}\mathbf{v}_j + \mathbf{B}\mathbf{u}_j + \mathbf{a}) \leq 0 \tag{5}$$

Condition 2. Let $\mathbf{x}_f \in S$, and $(\mu_1, ..., \mu_{m+1})$ be s.t. $\sum_{j=1}^{m+1} \mu_j \mathbf{v}_j = \mathbf{x}_f$. According to Theorem 4.19 in Habets et al. [2006], $\mathbf{x}_f$ is the *unique equilibrium point* in $\mathcal{S}$ (in closed loop) iff

$$i)\ \mathbf{B} \sum_{j=1}^{m+1} \mu_j \mathbf{u}_j = -\mathbf{A}\mathbf{x}_f - \mathbf{a} \qquad \{\text{i.e., } \dot{\mathbf{x}}_f = \mathbf{0}\} \tag{6}$$

$$ii)\ span(\{\mathbf{A}\mathbf{v}_j + \mathbf{B}\mathbf{u}_j | j = 1, ..., m+1\}) = \mathbb{R}^m \tag{7}$$

According to Theorems 4.18-4.19 in Habets et al. [2006], conditions (5-7) can be combined in order to compute a control law that fulfills different requirements. Here, we are interested in two particular problems:

Problem 1.a) Find an admissible affine control law such that for every initial state $\mathbf{x}_0 \in S$, the corresponding state trajectory $\mathbf{x}(t, \mathbf{x}_0)$ of the closed-loop system satisfies $\forall t \geq 0$, $\mathbf{x}(t, \mathbf{x}_0) \in S$, i.e., the system remains inside $\mathcal{S}$.
Problem 1.b) Additionally, it holds $lim_{t\to\infty}\mathbf{x}(t, \mathbf{x}_0) = \mathbf{x}_f$.

Solution: Problem 1.a is solved by computing values for the control law at the vertices $\mathbf{u}_j$ in such a way that (5) is fulfilled for all the facets of $\mathcal{S}$. Problem 1.b is solved if additionally *condition 2* is fulfilled (i.e., (6) and (7)). Notice that (5) and (6) are linear inequalities (only the values of $\mathbf{u}_j$ are unknown, while other vectors and parameters are known) then the computation of values $\mathbf{u}_j$ satisfying (5) and (6) can be done in polynomial time. Once these are computed, it must be checked wether (7) holds or not, in a negative case, $\mathbf{u}_j$ should be computed again (the set of solutions $\mathbf{u}_j$ that do not fulfill (7) is on a smaller dimension manifold, so, it is improbable to obtain such values during the computation). Finally, the control law, i.e., the pair $(\mathbf{F}, \mathbf{g})$, can be obtained by solving (4).

4. CENTRALIZED CONTROL FOR $TCPN$S

The results previously recalled were derived for simplices. Through this section, those results will be extended to polytopes, and later, they will be used for the synthesis of controllers for $TCPN$s (polytopes are mostly encountered during the synthesis of controllers for $TCPN$s).

A possible solution for the synthesis of affine controllers in polytopes was proposed by Habets and van Schuppen [2004], by decomposing the polytope into simplices and synthesizing a proper affine control law for each of them. In this section, a different approach will be introduced, by synthesizing a unique (global) affine control law for the complete polytope. The resulting control scheme is more conservative, but it will be demonstrated that such control law can always be computed for $TCPN$ systems.

4.1 Affine control on polytopes

In polytopes, the matrix $\begin{bmatrix} \mathbf{v}_1 & ... & \mathbf{v}_{m+1} \\ 1 & ... & 1 \end{bmatrix}$ in (4) has, in general, more columns than rows (i.e., the number of vertices is larger than the dimension of the polytope plus 1). Therefore, computing the values $\mathbf{u}_j$ that fulfill the required conditions may not be sufficient for obtaining a control law, since the existence of a pair ($\mathbf{F}$,$\mathbf{g}$) that satisfies (4) for the obtained values $\mathbf{u}_j$ is not guaranteed.

In this case, it is additionally required that the same linear dependencies that involve the vertices $\mathbf{v}_j$ be also effective for the values $\mathbf{u}_j$.

Proposition 2. Consider a polytope of dimension $k-1$ with m vertices, and the values for the input at those ($\mathbf{u}_j$, $j \in \{1, ..., m\}$). Assume, without loss of generality, that the first k vertices define a simplex of dimension $k-1$. For each vertex $\mathbf{v}_j \neq \mathbf{v}_1$ there exists a unique column vector $\boldsymbol{\gamma}_i$ s.t. $\mathbf{v}_j - \mathbf{v}_1 = [\mathbf{v}_2 - \mathbf{v}_1, \mathbf{v}_3 - \mathbf{v}_1, ..., \mathbf{v}_k - \mathbf{v}_1]\boldsymbol{\gamma}_j$. Then, there exists a pair $(\mathbf{F}, \mathbf{g})$ that fulfills (4) iff:

$$\begin{array}{l}\forall j \in \{k+1, ..., m\} \\ \qquad \mathbf{u}_j - \mathbf{u}_1 = [\mathbf{u}_2 - \mathbf{u}_1, \mathbf{u}_3 - \mathbf{u}_1, ..., \mathbf{u}_k - \mathbf{u}_1]\,\boldsymbol{\gamma}_j\end{array} \tag{8}$$

Proof. By hypothesis, the first k vertices define a simplex of dimension $k-1$, the same dimension of the whole polytope. Therefore, all the vertices belong to the hyperplane defined by the first k vertices. Then, for each $\mathbf{v}_j \neq \mathbf{v}_1$ it must exist a vector $\boldsymbol{\gamma}_j$ s.t. $\mathbf{v}_j - \mathbf{v}_1 = [\mathbf{v}_2 - \mathbf{v}_1, ..., \mathbf{v}_k - \mathbf{v}_1]\boldsymbol{\gamma}_j$ (see fig. 1(a)). Furthermore, the matrix $[\mathbf{v}_2 - \mathbf{v}_1, ..., \mathbf{v}_k - \mathbf{v}_1]$ has full column rank, which implies that $\boldsymbol{\gamma}_j$ is unique.
Now, (4) is equivalent to $\forall j \;\; \mathbf{u}_j = \mathbf{F}\mathbf{v}_j + \mathbf{g}$. Since the first k vertices define a full-dimensional simplex then there always exists a unique pair $(\mathbf{F}, \mathbf{g})$ that fulfills $\mathbf{u}_j = \mathbf{F}\mathbf{v}_j + \mathbf{g}$, $\forall j \in \{1, .., k\}$. Such pair also fulfills $\mathbf{u}_j = \mathbf{F}\mathbf{v}_j + \mathbf{g}$, $\forall j \in \{k+1, .., m\}$, iff $\mathbf{u}_j - \mathbf{u}_1 = \mathbf{F}(\mathbf{v}_j - \mathbf{v}_1)$, $\forall j \in \{k+1, .., m\}$. Substituting $(\mathbf{v}_j - \mathbf{v}_1)$ by the expression previously obtained, it results $\mathbf{u}_j - \mathbf{u}_1 = \mathbf{F}[\mathbf{v}_2 - \mathbf{v}_1, ..., \mathbf{v}_k - \mathbf{v}_1]\boldsymbol{\gamma}_j$. Finally, this equation is equivalent to $\mathbf{u}_j - \mathbf{u}_1 = [\mathbf{u}_2 - \mathbf{u}_1, ..., \mathbf{u}_k - \mathbf{u}_1]\boldsymbol{\gamma}_j$. $\square$

For polytopes, the indices of the vertices in (5) need to be reconsidered. If F_i is a facet that must be disabled (*condition 1*) then, denoting as $\mathcal{I}_{F_i}$ the indices of the vertices related to F_i, (5) is rewritten as

$$\forall j \in \mathcal{I}_{F_i} \quad \mathbf{n}_i^T(\mathbf{A}\mathbf{v}_j + \mathbf{B}\mathbf{u}_j + \mathbf{a}) \leq 0 \tag{9}$$

Similarly, (6) can be rewritten, by describing $\mathbf{x}_f$ as a linear combination of linearly independent vertices. Assume, without loss of generality, that the first k vertices define a simplex having the same dimension of the polytope. Then, define $(\mu_1, ..., \mu_k)$ such that $\sum_{j=1}^{k} \mu_j \mathbf{v}_j = \mathbf{x}_f$. Thus, (6) is transformed into

$$\mathbf{B}\sum_{j=1}^{k} \mu_j \mathbf{u}_j = -\mathbf{A}\mathbf{x}_f - \mathbf{a} \tag{10}$$

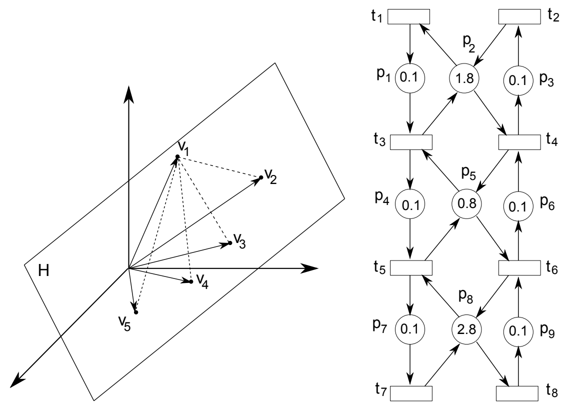
In this way, the results recalled in Section 3, regarding the control problems 1.a and 1.b, are extended to polytopes.

4.2 Affine control laws for TCPN systems

Consider a $TCPN$ system, controllable over the interior of $Class(\mathbf{m}_0)$ (thus, according to Theorem 1, the net is consistent). Define the set $int_\epsilon\{Class(\mathbf{m}_0)\} = \{\mathbf{m} \in Class(\mathbf{m}_0) | \mathbf{m} \geq \mathbf{1} \cdot \epsilon\}$, for an arbitrarily small $\epsilon > 0$. This subsection is devoted to the following control problem:

Problem 2) Find a s.b. control law for driving the $TCPN$ system towards the desired marking $\mathbf{m}_f$, assuming $\mathbf{m}_f, \mathbf{m}_0 \in int_\epsilon\{Class(\mathbf{m}_0)\}$.

Procedure 1. Synthesis of a s.b. centralized control law for driving the system towards $\mathbf{m}_f$.



(a) (b)

Fig. 1. (a) Fig. for the proof of proposition 2. Vertices belong to the hyperplane H. Dashed lines represent vectors ($\mathbf{v}_j$-$\mathbf{v}_1$). (b) A consistent $TCPN$ system.

==================================
Synthesis of the control law (off-line):

I . Compute the vertices $\{\mathbf{v}_1, ..., \mathbf{v}_m\}$ of $int_\epsilon\{Class(\mathbf{m}_0)\}$ and enumerate them s.t. the first k define a simplex having the same dimension of the polytope.

II . Compute the vectors $\mathbf{n}_i$ normal to the corresponding facets of $Class(\mathbf{m}_0)$ and pointing outwards this.

III. For the system $\dot{\mathbf{m}} = \mathbf{C}\mathbf{w}$ where $\mathbf{w}$ is the control input, compute some values $\mathbf{w}_j$ for the input at the vertices (i.e., with $\mathbf{w}_j$ instead $\mathbf{u}_j$) simultaneously fulfilling:
- $\mathbf{w}_j \geq \mathbf{0}$,
- condition (8),
- condition (9) for all the facets,
- condition (10) with $\mathbf{x}_f = \mathbf{m}_f$.

IV. Compute $\mathbf{F}$ and $\mathbf{g}$ that fulfill (4).

V . Verify that $\mathbf{m}_f$ is the unique equilibrium point in the closed-loop system that belongs to $Class(\mathbf{m}_0)$, by using (7) (the condition (6) is already fulfilled, since it was transformed into (10)). Otherwise, compute another values $\mathbf{u}_j$ and $(\mathbf{F}$,$\mathbf{g})$.

==================================
Application of the control law (on-line):

I . Define $\eta(\mathbf{m}) = min(\mathbf{f}(\mathbf{m})./(\mathbf{F}\mathbf{m} + \mathbf{g}))$, where ./ denotes the element-wise division operator.

II. Apply the control law:

$$\mathbf{u}(\mathbf{m}) = \mathbf{f}(\mathbf{m}) - \eta(\mathbf{m}) \cdot (\mathbf{F}\mathbf{m} + \mathbf{g}) \tag{11}$$

==================================

Step I of the synthesis procedure corresponds to the *vertex enumeration problem*. This is not (in general) a polynomial-time problem, nevertheless, there exist efficient tools for its resolution (e.g., Fukuda [2000]) for polytopes of considerable dimension. Vectors $\mathbf{n}_i$ computed in step II are also normal to the facets of $int_\epsilon\{Class(\mathbf{m}_0)\}$. The computation of these as stated in step II has the advantage that it depends only on the original polytope $Class(\mathbf{m}_0)$. All the constraints for the values $\mathbf{w}_j$ in step III are linear. Then, step III can be achieved in polynomial time. In particular, consider the quadratic problem: $min \sum(\mathbf{f}(\mathbf{v}_j) - \mathbf{w}_j)^T(\mathbf{f}(\mathbf{v}_j) - \mathbf{w}_j)$, subject to the constraints in step III. This will usually lead to a fast control law, since the values $\mathbf{w}_j$ thus computed will be close to their upper bounds $\mathbf{f}(\mathbf{v}_j)$ (i.e., values that allow the maximum flow). In any case, the scalar function $\eta(\mathbf{m})$ ensures that the control action is always s.b.

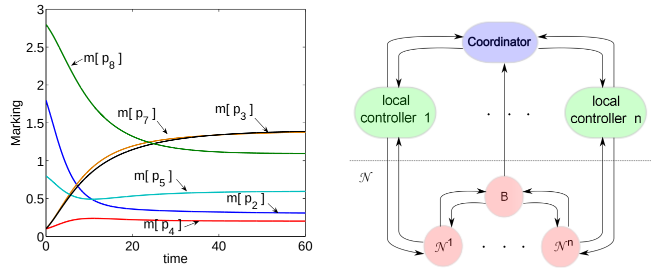
Proposition 3. It is always possible to compute a control law (11) by using *Procedure 1*.

Proof. Let us construct a particular control law. Consider the polytope $int_{\epsilon}\{Class(\mathbf{m}_0)\}$. Enumerate its vertices s.t. the first k define a simplex $\mathcal{S}$, having the same dimension of the polytope, that includes $\mathbf{m}_f$.
First, for the vertices of such simplex (i.e., $\{\mathbf{v}_1, ..., \mathbf{v}_k\}$), define the vectors $\mathbf{d}_j = \mathbf{m}_f - \mathbf{v}_j$. Since the net is consistent then, for each $\mathbf{d}_j$ there exists $\mathbf{w}_j$ s.t. $\mathbf{d}_j = \mathbf{C} \cdot \mathbf{w}_j$ and $\mathbf{w}_j \geq \mathbf{0}$. These $\mathbf{w}_j$ define an affine control law ($\mathbf{F}$,$\mathbf{g}$) according to (4). Notice that, considering the system as $\dot{\mathbf{m}} = \mathbf{C}\mathbf{w}$, such values $\mathbf{w}_j$ fulfill with (9), since the field vector at the vertices $\{\mathbf{v}_1, ..., \mathbf{v}_k\}$ points towards $\mathbf{m}_f$ (i.e., inside the simplex, thus the polytope). Furthermore, by linearity of the model ($\dot{\mathbf{m}} = \mathbf{C}\mathbf{w}$), the field vector at $\mathbf{m}_f$ is null, so, the condition (10) holds. Moreover, (7) also holds since the field vector at the vertices of the simplex constitutes a basis (by definition, $span\{\mathbf{d}_1, ..., \mathbf{d}_k\} = span\{\mathbf{v}_1, ..., \mathbf{v}_k\}$).
Given such $\mathbf{w}_j$ for $j \in \{1, ..., k\}$, the values of $\mathbf{w}_j$ for $j \in \{k+1, ..., N\}$ are uniquely determined according to (8). Let us show that these also fulfill (9).
For each vertex $\mathbf{v}_j \in \{\mathbf{v}_{k+1}, ..., \mathbf{v}_m\}$, define $\mathbf{v}'_j$ as the intersection of the segment ($\mathbf{m}_f$,$\mathbf{v}_j$) with the frontier of the simplex $\mathcal{S}$. In this way, there must exist $\boldsymbol{\gamma} \geq \mathbf{0}$ s.t. $[\mathbf{v}_1, ..., \mathbf{v}_k]\boldsymbol{\gamma} = \mathbf{v}'_j$ and $\mathbf{1} \cdot \boldsymbol{\gamma} = 1$. According to this, by linearity, $\mathbf{d}'_j = \mathbf{m}_f - \mathbf{v}'_j = [\mathbf{d}_1, ..., \mathbf{d}_k]\boldsymbol{\gamma}$. Similarly, by linearity, the value of the input at $\mathbf{v}'_j$, denoted as $\mathbf{w}'_j$, is s.t. $\mathbf{w}'_j = [\mathbf{w}_1, ..., \mathbf{w}_k]\boldsymbol{\gamma} \geq \mathbf{0}$ and $\mathbf{d}'_j = \mathbf{C} \cdot \mathbf{w}'_j$, i.e., the field vector at $\mathbf{v}'_j$ points towards $\mathbf{m}_f$. Finally, since $\mathbf{v}'_j$ is in the segment ($\mathbf{m}_f$,$\mathbf{v}_j$) and the field vector at $\mathbf{m}_f$ is null, then the field vector at $\mathbf{v}_j$ is also pointing towards $\mathbf{m}_f$ and the input at this fulfills $\mathbf{w}_j \geq \mathbf{0}$. Thus, the input at this vertex also fulfills (9). Therefore, all the constraints enumerated in steps I-V in Procedure 1 are fulfilled. □

Proposition 4. Control law (11) drives the system towards the required marking $\mathbf{m}_f$ while the control action is s.b. along the trajectory.

Proof. Consider the state equation of the $TCPN$ system as in (2). This model is linear, then the current control problem is similar to the problem 2.b of Section 3 with $\mathbf{x}_f = \mathbf{m}_f$, but in a polytope instead a simplex, and with the input constraint $\mathbf{0} \leq \mathbf{w} \leq \mathbf{f}(\mathbf{m})$ instead $\mathbf{u} \in U$. Therefore, according to the results shown in the previous subsections, the control law $\mathbf{w}(\mathbf{m}) = \mathbf{F}\mathbf{m} + \mathbf{g}$ (equivalently, $\mathbf{u} = \mathbf{f}(\mathbf{m}) - (\mathbf{F}\mathbf{m} + \mathbf{g})$), where $(\mathbf{F}, \mathbf{g})$ are obtained through the steps I-V of the previous procedure, would drive the system towards $\mathbf{m}_f$ through a trajectory inside $int_{\epsilon}\{Class(\mathbf{m}_0)\}$. Since the closed-loop system is affine, then convergency to $\mathbf{m}_f$ means asymptotic stability.
The closed-loop system with such input is equivalent to the closed-loop system with (11) and $\eta = 1$. In such case, since $\mathbf{m}_f$ is asymptotically stable, there exists a quadratic Lyapunov function $\mathbf{V}(\mathbf{m}) = (\mathbf{m} - \mathbf{m}_f)^T\mathbf{P}(\mathbf{m} - \mathbf{m}_f)$ whose derivative is negative, i.e., $\dot{\mathbf{V}}(\mathbf{m}) = -(\mathbf{m} - \mathbf{m}_f)^T\mathbf{Q}(\mathbf{m} - \mathbf{m}_f) < 0$, where the matrix $\mathbf{Q} = -[(\mathbf{CF})^T\mathbf{P} + \mathbf{P}(\mathbf{CF})]$ is positive definite ($\mathbf{x}^T\mathbf{Q}\mathbf{x} > 0\ \forall \mathbf{x} \neq \mathbf{0}$). By using the same Lyapunov function for the closed-loop system under (11) (thus $\eta \neq 1$), its derivative can be computed as $\dot{\mathbf{V}}(\mathbf{m}) = -\eta(\mathbf{m})(\mathbf{m} - \mathbf{m}_f)^T\mathbf{Q}(\mathbf{m} - \mathbf{m}_f)$. This is negative (meaning that the system will be driven towards $\mathbf{m}_f$) whenever $\eta(\mathbf{m}) > 0$. This holds since $\mathbf{f}(\mathbf{m}) > \mathbf{0}$ (because the close-loop system remains inside $int\{Class(\mathbf{m}_0)\}$) and $\mathbf{F}\mathbf{m} + \mathbf{g} \geq \mathbf{0}$ (due to the constraint $\mathbf{w}_j \geq \mathbf{0}$).
Finally, since $\eta(\mathbf{m}) \cdot (\mathbf{F}\mathbf{m} + \mathbf{g}) \geq \mathbf{0}$ (so $\mathbf{f}(\mathbf{m}) \geq \mathbf{u}(\mathbf{m})$) and $\eta(\mathbf{m}) = min(\mathbf{f}(\mathbf{m})./(\mathbf{F}\mathbf{m} + \mathbf{g}))$ implies $\dot{\eta}(\mathbf{m})(\mathbf{F}\mathbf{m} + \mathbf{g}) \leq \mathbf{f}(\mathbf{m})$ (so $\mathbf{u}(\mathbf{m}) \geq \mathbf{0}$), then the input is s.b. □

Example 1. Consider the $TCPN$ system depicted in fig. 1(b), with initial marking $\mathbf{m}_0 = [0.1, 1.8, 0.1, 0.1, 0.8, 0.1, 0.1, 2.8, 0.1]^T$ and timing $\boldsymbol{\lambda} = [1, 1, 1, 1, 1, 1, 1, 1]^T$. It is desired to drive this system towards $\mathbf{m}_f = [0.3, 0.3, 1.4, 0.2, 0.6, 0.2, 1.4, 1.1, 0.5]^T$. This $TCPN$ can be seen as a piecewise-linear system with 16 different modes (configurations), according to the state equation (1). Nevertheless, by following *Procedure 1*, a unique affine control was obtained (11), by computing a gain matrix $\mathbf{F}$, of order 9×8, and a vector $\mathbf{g} = \mathbf{0}$. This control law was applied to the system. Fig. 2(a) shows the resulting marking trajectories. It can be observed that the control law successfully drives the system towards the desired marking.



(a) (b)

Fig. 2. (a) Evolution of the marking of the controlled system of fig. 1(b). (b) Modular-coordinated control scheme.

5. COORDINATED CONTROL

The main drawback of the control law derived in the previous section is the fact that the number of vertex increases exponentially w.r.t. the dimension of the polytope, which is determined by the number of places (state variables). In order to deal with such complexity, a modular-coordinated control strategy will be derived through this section.

Definition 5. Given a PN $\mathcal{N} = \langle P, T, \mathbf{Pre}, \mathbf{Post}\rangle$, a *modular view* of this is a set of n PNs, named *modules*, denoted as $\mathcal{N}^i = \langle P^i, T^i, \mathbf{Pre}^i, \mathbf{Post}^i, \mathbf{m}_0^i\rangle$, where $\mathbf{m}_0^i = \mathbf{m}_0[P^i]$, $\mathbf{Pre}^i = \mathbf{Pre}[P^i, T^i]$ and $\mathbf{Post}^i = \mathbf{Post}[P^i, T^i]$, for each $i \in \{1, ..., n\}$. These modules are interconnected by places, called buffers B, so P is the disjoint union of $P^1, ..., P^n$ and B, and T is the disjoint union of $T^1, ..., T^n$. We assume that the following conditions hold:

(1) For every $i, j \in \{1, ..., n\}$, if $i \neq j$ then $\mathbf{Pre}[P^i, T^j] = \mathbf{Post}[P^i, T^j] = \mathbf{0}$.
(2) For each buffer $b \in B$, $|b^{\bullet}| \geq 1$.
(3) For every $i \in \{1, ..., n\}$, the module $\mathcal{N}^i$ is consistent.
(4) The net model $\mathcal{N}$ is consistent.

In the sequel, it will be assumed, without loss of generality, that the incidence matrix of the modular net model has a structure like:

$$\mathbf{C} = \begin{bmatrix} \mathbf{C}^1 & \mathbf{0} & ... & \mathbf{0} \\ \mathbf{0} & \mathbf{C}^2 & ... & \mathbf{0} \\ \vdots & \vdots & ... & \vdots \\ \mathbf{0} & \mathbf{0} & ... & \mathbf{C}^n \\ \mathbf{C}_B^1 & \mathbf{C}_B^2 & ... & \mathbf{C}_B^n \end{bmatrix}$$

where $\mathbf{C}^i = \mathbf{C}[P^i, T^i]$ and $\mathbf{C}_B^i = \mathbf{C}[B, T^i]$. In this way, the marking is represented as $\mathbf{m} = [(\mathbf{m}^1)^T, .., (\mathbf{m}^n)^T, (\mathbf{m}^B)^T]^T$, where $(\mathbf{m}^i)^T$ is the transpose of $\mathbf{m}^i = \mathbf{m}[P^i]$, and similarly $\mathbf{m}^B = \mathbf{m}[B]$. Furthermore, $\mathbf{B}_x^i \geq \mathbf{0}$ represents a basis for the T-semiflows of $\mathbf{C}^i$. In this section, we are interested in the following control problem:

Problem 3) Given a $TCPN$ system $\langle \mathcal{N}, \boldsymbol{\lambda}, \mathbf{m}_0 \rangle$, where $\mathcal{N}$ is a modular PN, find a *coordinated-control scheme* for driving concurrently each module $\mathcal{N}^i$ of $\mathcal{N}$ towards a desired marking $\mathbf{m}_f^i \in Class(\mathbf{m}_0^i)$ by means of s.b. control actions, assuming they are concurrently reachable while the buffers remain marked.

Fig. 2(b) shows the structure of a coordinated-control scheme. This consists of a set of local controllers and an upper-level controller, named *coordinator*. A local controller is synthesized for each module, receiving information from the coordinator and having local information: the marking of the corresponding module and the neighboring buffers. The coordinator receives and sends minimum information from and to the local controllers, but it does not apply control actions into the system. Furthermore, the coordinator can observe the marking of the buffers. Let us remark that different control techniques can be considered for the local controllers, nevertheless, in this work we will focus on piecewise affine control laws.

5.1 The need for coordination

By hypothesis, a modular $TCPN$ system is consistent, thus controllable over the corresponding $int\{Class(\mathbf{m}_0)\}$. The same holds for each of the modules $\mathcal{N}^i$. Thus, if the initial marking of the buffers were large enough (i.e., if $\mathbf{m}_0^B >> \mathbf{0}$), each module could be driven towards its corresponding desired marking $\mathbf{m}_f^i \in Class(\mathbf{m}_0^i)$ by means of a *local control law* like (11), using only local information (i.e., $\mathbf{m}^i$). In such case, it would be obtained a completely decentralized control scheme with neither communication between local controllers nor with the coordinator.

Example 2. Consider the modular $TCPN$ system depicted in fig. 3, consisting of three modules interconnected by four buffers. The rates for the transitions are $\boldsymbol{\lambda}^1 = [4, 1, 1, 1]$ for $\mathcal{N}^1$, $\boldsymbol{\lambda}^2 = [1, 1, 1]$ for $\mathcal{N}^2$ and $\boldsymbol{\lambda}^3 = [1, 1, 1, 1]$ for $\mathcal{N}^3$. The initial markings are $\mathbf{m}_0^1 = [9.7, 0.1, 0.1, 0.1]^T$ for $\mathcal{N}^1$, $\mathbf{m}_0^2 = [0.1, 0.1, 4.8]^T$ for $\mathcal{N}^2$ and $\mathbf{m}_0^3 = [4.8, 0.1, 0.1]^T$ for $\mathcal{N}^3$. The initial marking at the buffers is $\mathbf{m}_0^B = [0.1, 0.1, 2.5, 4]^T$. Consider the control problem of transferring the modules towards $\mathbf{m}_f^1 = [1, 1, 1, 7]$, $\mathbf{m}_f^2 = [0.5, 3, 1.5]$ and $\mathbf{m}_f^3 = [1.5, 0.5, 3]$, respectively. For each module, gain matrices $\mathbf{F}^i$ and vectors $\mathbf{g}^i$ were computed, by using *Procedure* 1. Later, the resulting control laws (11) were simultaneously applied to the corresponding modules. Fig. 4(a) shows the obtained trajectories for the marking of $\mathcal{N}^1$. It can be observed that the module got stuck at a marking different than the desired one. The same has occurred to the other modules, because the buffers b_1 and b_2 were emptied, as shown in fig. 4(b) ($\mathcal{N}^3$ requires marking from b_1 while $\mathcal{N}^1$ and $\mathcal{N}^2$ require marking from b_2). Thus, although the net system is live (without control), the applied control scheme stops the activity in all the modules.

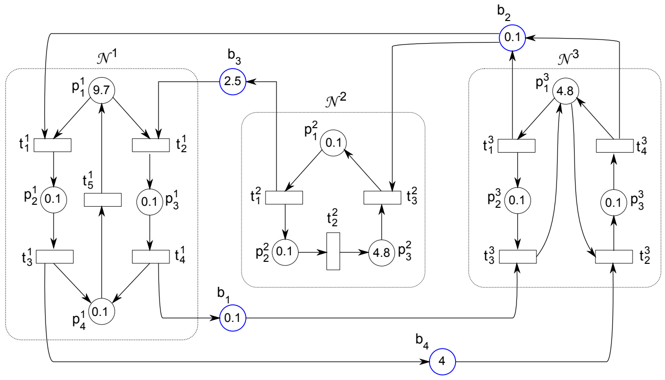
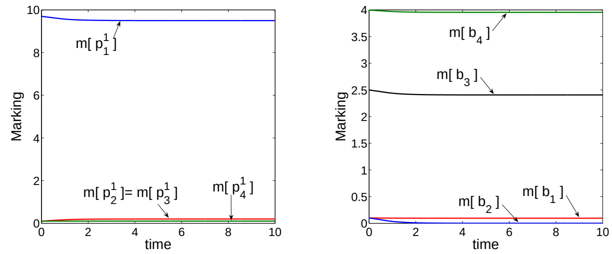


Fig. 3. A consistent and conservative $TCPN$ system. Buffer b_2 is not output private (it supplies tokens to $\mathcal{N}^1$ and $\mathcal{N}^2$).



(a) (b)

Fig. 4. (a) Marking of Module 1 in closed-loop. (b) Marking in the buffers.

In general, the marking at the buffers is limited (frequently, they are part of a *P-component*), and so, they impose constraints to the control actions that are not taken into account during the synthesis of the local controllers (e.g., a module $\mathcal{N}^i$ has the constraint $\mathbf{w}^i \leq \mathbf{f}^i(\mathbf{m}^i, \mathbf{m}^B)$, where $\mathbf{m}^B$ represents the marking at the buffers, but the control law obtained with *Procedure 1* only fulfills $\mathbf{w}^i \leq \mathbf{f}^i(\mathbf{m}^i)$). In the worst case, a set of buffers may empty, consequently, the local controllers will stop the activity of the modules that require tokens from them. This does not mean that the system is actually in a deadlock, on the contrary, it might be possible to recover the system. Nevertheless, the local controllers are not synthesized in order to increase the marking at the buffers. Therefore, it is important to have a certain *coordination* between the local controllers, in order to make the modules to evolve with a suitable speed so that the buffers do not empty (i.e., imposing a *fairness* relation between modules). This will be the task of the *coordinator*.

5.2 Modular-coordinated control

In sequel, it will be assumed that both the initial and the final markings belong to $int_\epsilon\{Class(\mathbf{m}_0)\}$ for a given small

$\epsilon > 0$. Similarly, it will be assumed that the operation of each local controller and the coordinator are ruled by a global clock and they always receive and send the required information without loss or delays.

As shown in the previous subsection, certain coordination is required in order to make the buffers to remain marked. This can be achieved by applying a modular control scheme, like the one shown in fig. 2(b), that drives the system through a linear (straight) trajectory (due to the convexity of $Class(\mathbf{m}_0)$, it is always possible to drive the system to any reachable marking, through a straight trajectory). Let us describe this. Compute firstly a proper final marking for the buffers $\mathbf{m}_f^B > \mathbf{1}\epsilon$ such that $\mathbf{m}_f = [(\mathbf{m}_f^1)^T, ..., (\mathbf{m}_f^n)^T, (\mathbf{m}_f^B)^T]^T$ is reachable (such marking exists by hypothesis). Next, compute for each module (task for the local controllers) a vector $\mathbf{d}^i > \mathbf{0}$ s.t. $(\mathbf{m}_f^i - \mathbf{m}^i) = \mathbf{C}^i\mathbf{d}^i$. Furthermore, compute (task for the coordinator) a vector $\boldsymbol{\gamma} > \mathbf{0}$ s.t. $(\mathbf{m}_f^B - \mathbf{m}^B) = \sum \mathbf{C}_B^i\mathbf{d}^i + [\mathbf{C}_B^1\mathbf{B}_x^1, ..., \mathbf{C}_B^n\mathbf{B}_x^n]\boldsymbol{\gamma}$. In this way, if each local controller applies $\mathbf{w}^i = \mathbf{d}^i + \mathbf{B}_x^i\boldsymbol{\gamma}[T^i]$ then the closed-loop behavior of each module will be $\dot{\mathbf{m}}^i = \mathbf{C}^i\mathbf{d}^i + \mathbf{C}^i\mathbf{B}_x^i\boldsymbol{\gamma}[T^i] = (\mathbf{m}_f^i - \mathbf{m}^i)$, i.e., each field vector will be pointing towards the corresponding $\mathbf{m}_f^i$, consequently, the modules will be driven towards their final states describing linear trajectories. Moreover, the closed-loop behavior of the marking of the buffers will be $\dot{\mathbf{m}}^B = \sum \mathbf{C}_B^i\mathbf{d}^i + [\mathbf{C}_B^1\mathbf{B}_x^1, ..., \mathbf{C}_B^n\mathbf{B}_x^n]\boldsymbol{\gamma} = (\mathbf{m}_f^B - \mathbf{m}^B)$, thus, the marking of the buffers will converge to $\mathbf{m}_f^B$ describing a linear trajectory, so, the buffers will remain marked.

This control scheme can be extended by adding an affine control element to the local control laws. In detail, define $\mathbf{w}'^i = \mathbf{F}^i\mathbf{m}^i + \mathbf{g}^i$, where $(\mathbf{F}^i, \mathbf{g}^i)$ is a proper affine control law computed (for each module) by using *Procedure 1.* Defining a *linearizing factor* $\psi \in [0, 1]$, the following local control law is proposed for each module: $\mathbf{w}^i = \mathbf{d}^i\psi + \mathbf{w}'^i(1 - \psi) + \mathbf{B}_x^i\boldsymbol{\gamma}^i$. Notice that, if $\psi = 1$ then the local controllers will drive their modules toward the corresponding $\mathbf{m}_f^i$ describing linear trajectories (it is actually the control scheme described in the previous paragraph). On the other hand, with $\psi = 0$ the control laws obtained are the evaluations of the affine control laws $\mathbf{w}'^i$, which corresponds to the decentralized scheme with local affine controllers used in *example 2.* In order to avoid the buffers to become empty, the linearizing factor must be properly computed. One possibility is to impose the constraint $\dot{\mathbf{m}}^B = \sum \mathbf{C}_B^i[\mathbf{d}^i\psi + \mathbf{w}'^i(1 - \psi) + \mathbf{B}_x^i\boldsymbol{\gamma}^i] > \mathbf{1}\epsilon - \mathbf{m}^B$. In this way, the field vector of the marking of the buffers is always pointing towards a positive marking $> \mathbf{1}\epsilon$ (given the definition of $\boldsymbol{\gamma}$, previous constraint is equivalent to $\sum \mathbf{C}_B^i(\mathbf{w}'^i - \mathbf{d}^i)(1 - \psi) > \mathbf{1}\epsilon - \mathbf{m}_f^B$).

Finally, in order to make the control laws to be s.b. (i.e., $\mathbf{w}^i \leq \mathbf{f}^i(\mathbf{m}^i, \mathbf{m}^B)$), a proper global scale factor η will be applied to the control actions, i.e., each local controller will apply $\mathbf{w}^i = [\mathbf{d}^i\psi + \mathbf{w}'^i(1 - \psi = +\mathbf{B}_x^i\boldsymbol{\gamma}^i]\eta$, where $\eta > 0$ is the maximum scalar s.t. $\mathbf{w}^i \leq \mathbf{f}^i(\mathbf{m}^i, \mathbf{m}^B)$ for all the modules. The factor must be the same for all the local controllers, so the direction of the global field vector is not modified. Then, η must be computed by the coordinator by using information from the local controllers, regarding the maximum control action allowed with respect to each of the three components $\mathbf{d}_i$, $\mathbf{B}_x^i$ and $\mathbf{w}'^i$ (codified in three factors η_d^i, $\eta_{x_j}^i$ and η_w^i, respectively). Combining all these issues, the following control scheme is proposed:

Procedure 2. Synthesis of a coordinated control scheme.
==================================
Synthesis of local control laws (planing step):
==================================

- *Coordinator:* Compute a suitable desired marking for the buffers s.t. all of them are marked $\mathbf{m}_f^B > \mathbf{1}\epsilon$ and $\mathbf{m}_f = [(\mathbf{m}_f^1)^T, ..., (\mathbf{m}_f^n)^T, (\mathbf{m}_f^B)^T]^T$ is reachable. Compute a T-semiflow $\mathbf{0} \leq \mathbf{x} \leq \mathbf{f}(\mathbf{m}_f)$.

- *Each local controller:* Compute an affine control law $(\mathbf{F}^i, \mathbf{g}^i)$, by using Procedure 1, for driving the module $\mathcal{N}^i$ to the corresponding $\mathbf{m}_f^i$, with the additional constraint $\mathbf{w}(\mathbf{m}_f^i) = \mathbf{x}[T^i]$.

==================================
Dynamic control (on-line, in discrete time):
==================================
Coordinator:

- ***Receive** from the local controllers the values $\eta_{x_j}^i$, η_d^i, η_w^i, $\mathbf{C}_B^i\mathbf{d}^i$ and $\mathbf{C}_B^i \cdot (\mathbf{w}'^i - \mathbf{d}^i)$.*

I . Compute a vector $\boldsymbol{\gamma} > \mathbf{0}$ solution for

$$(\mathbf{m}_f^B - \mathbf{m}^B) - \sum \mathbf{C}_B^i\mathbf{d}^i = [\mathbf{C}_B^1\mathbf{B}_x^1, ..., \mathbf{C}_B^n\mathbf{B}_x^n]\boldsymbol{\gamma} \quad (12)$$

II . Compute $\psi = min\ \alpha$ s.t. $\alpha \in [0, 1]$ and $\sum \mathbf{C}_B^i(\mathbf{w}'^i - \mathbf{d}^i)(1 - \alpha) > \mathbf{1}\epsilon - \mathbf{m}_f^B$.

III. Define, for each module, $\boldsymbol{\gamma}^i = \boldsymbol{\gamma}[T^i]$ and $\eta^i = 1/((1 - \psi)/\eta_w^i + \psi/\eta_d^i + \sum \boldsymbol{\gamma}^i[j]/\eta_{x_j}^i)$.

IV. Evaluate $\eta = min(\eta^1, ..., \eta^n)$.

- ***Send**, to each local controller, the values η, ψ and $\boldsymbol{\gamma}^i$.*

For each local controller:

- ***Receive** from the coordinator the new values for ψ, η and $\boldsymbol{\gamma}^i$.*

I . Compute a vector $\mathbf{d}^i > \mathbf{0}$ s.t. $(\mathbf{m}_f^i - \mathbf{m}^i) = \mathbf{C}^i\mathbf{d}^i$.

II . Evaluate the affine control action: $\mathbf{w}'^i = \mathbf{F}^i\mathbf{m}^i + \mathbf{g}^i$.

III. Apply the control action:

$$\mathbf{u}^i = \mathbf{f}^i(\mathbf{m}^i, \mathbf{m}^B) - \left[\mathbf{w}'^i(1 - \psi) + \mathbf{d}^i\psi + \mathbf{B}_x^i\boldsymbol{\gamma}^i\right]\eta \quad (13)$$

IV. Compute the values:
  - $\eta_d^i = max\,\alpha$ s.t. $\mathbf{d}^i\alpha \leq \mathbf{f}^i(\mathbf{m}^i, \mathbf{m}^B)$,
  - $\eta_w^i = max\,\alpha$ s.t. $\mathbf{w}'^i\alpha \leq \mathbf{f}^i(\mathbf{m}^i, \mathbf{m}^B)$,
  - $\eta_{x_j}^i = max\,\alpha$ s.t. $\mathbf{x}_j\alpha \leq \mathbf{f}^i(\mathbf{m}^i, \mathbf{m}^B)$, for each column (T-semiflow) $\mathbf{x}_j$ of $\mathbf{B}_x^i$.

- ***Send** the values $\eta_{x_j}^i$, η_d^i, η_w^i, $\mathbf{C}_B^i \cdot \mathbf{d}^i$ and $\mathbf{C}_B^i \cdot (\mathbf{w}'^i - \mathbf{d}^i)$ to the coordinator.*

==================================

In the appendix, an algorithm for the computation of $\mathbf{m}_f^B$ is provided. The computation of $\mathbf{d}^i$, achieved by each local controller at step I at each sampling, and $\boldsymbol{\gamma}$ achieved by the coordinator at step I, can be done with an algorithm provided in the appendix, having a linear complexity.

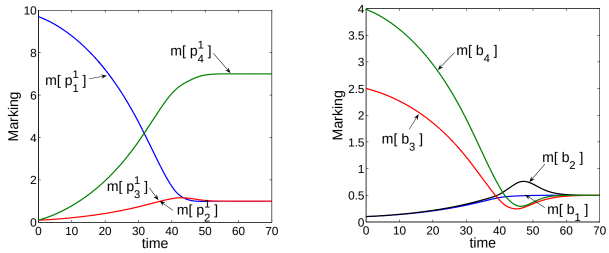
Similarly, an efficient algorithm for the computation of values $\eta^i_{x_j}$, η^i_w, η^i_d, η^i_T and ψ is also provided.

Proposition 5. Procedure 2 is well defined (i.e., all the required information is available and all the conditions for the computation are satisfied).

Proof. Since the net $\mathcal{N}$ is consistent and $\mathbf{m}_f$ is assumed reachable, then $\exists \boldsymbol{\sigma} > \mathbf{0}$ s.t. $(\mathbf{m}^i_f - \mathbf{m}^i) = \mathbf{C}^i\boldsymbol{\sigma}[T^i]$, for each module, and $(\mathbf{m}^B_f - \mathbf{m}^B) = \sum \mathbf{C}^i_B\boldsymbol{\sigma}[T^i]$. Consider a module $\mathcal{N}^i$. Since this is consistent, then there always exists a particular solution $\mathbf{d}^i > \mathbf{0}$ for $(\mathbf{m}^i_f - \mathbf{m}^i) = \mathbf{C}^i\mathbf{d}^i$. Furthermore, the general solution for $\boldsymbol{\sigma}[T^i]$ is given by $\boldsymbol{\sigma}[T^i] = \mathbf{d}^i + \mathbf{B}^i_x\boldsymbol{\gamma}^i$, with $\boldsymbol{\gamma}^i > \mathbf{0}$. Then, given particular solutions $\mathbf{d}^i$ for each module, there always exists $\boldsymbol{\gamma}$ s.t. $(\mathbf{m}^B_f - \mathbf{m}^B) = \sum \mathbf{C}^i_B\mathbf{d}^i + \sum \mathbf{C}^i_B\boldsymbol{\gamma}[T^i]$, i.e., solution for (12). Therefore, it is always possible to compute vectors $\mathbf{d}^i$ and $\boldsymbol{\gamma}$ according to the control procedure.
On the other hand, notice that $\psi = 1$ is a trivial solution for $\sum \mathbf{C}^i_B(\mathbf{w}'^i - \mathbf{d}^i)(1-\psi) > \mathbf{1}\epsilon - \mathbf{m}^B_f$. Moreover, the step *III* for the operation of the local controllers implies that the buffers remain marked (this is proven in detail in the proof of proposition 6). Thus, $\mathbf{f}^i(\mathbf{m}^i, \mathbf{m}^B) > \mathbf{0}$ for each module. Since $\mathbf{d}^i > \mathbf{0}$ and $\boldsymbol{\gamma} > \mathbf{0}$, then scalars η^i_w, η^i_d, $\eta^i_{x_j}$, η^i and η can always be computed and they are positive.
Finally, η is computed in such a way that each local control action is s.b., so it can be applied. Let us prove this by showing that η is computed in such a way that $\mathbf{w}^i = [\mathbf{w}'^i(1-\psi) + \mathbf{d}^i\psi + \mathbf{B}^i_x\boldsymbol{\gamma}^i]\eta \le \mathbf{f}^i(\mathbf{m}^i, \mathbf{m}^B)$ ($\mathbf{w}^i \ge \mathbf{0}$ already since $\mathbf{w}'^i > \mathbf{0}$, $\mathbf{d}^i > \mathbf{0}$, $\mathbf{B}^i_x \ge \mathbf{0}$ and $\boldsymbol{\gamma}^i > \mathbf{0}$). By using the definitions of η^i_d, η^i_w and $\eta^i_{x_j}$, computed by the local controllers, and η^i computed by the coordinator, it can be proved that $[\mathbf{w}'^i(1-\psi) + \mathbf{d}^i\psi + \mathbf{B}^i_x\boldsymbol{\gamma}^i]\eta^i \le \mathbf{f}^i(\mathbf{m}^i, \mathbf{m}^B)$. Later, since $0 < \eta \le \eta^i$, then $\mathbf{w}^i \le \mathbf{f}^i(\mathbf{m}^i, \mathbf{m}^B)$ for all the modules. □

Proposition 6. If the coordinated control scheme of *procedure 2* is applied, each module will be driven towards the corresponding $\mathbf{m}^i_f$.

Proof. For the sake of simplicity, the analysis is achieved in continuous-time, but an analogous reasoning can be used for the discrete-time case. Firstly let us show that the buffers remain marked. Consider a module $\mathcal{N}^i$. By definition of ψ, $\sum \mathbf{C}^i_B(\mathbf{w}'^i - \mathbf{d}^i)(1-\psi) > \mathbf{1}\epsilon - \mathbf{m}^f_B$. Combining this equation with (12), it is obtained $\mathbf{m}^B + \sum \mathbf{C}^i_B[\mathbf{w}'^i(1-\psi) + \mathbf{d}^i\psi + \mathbf{B}^i_x\boldsymbol{\gamma}^i] > \mathbf{1}\epsilon$. Substituting (13) it is obtained $\mathbf{m}_B + (1/\eta)\sum \mathbf{C}^i_B[\mathbf{f}^i(\mathbf{m}^i, \mathbf{m}^B) - \mathbf{u}^i] > \mathbf{1}\epsilon$. Furthermore, substituting $\dot{\mathbf{m}}^B = \sum \mathbf{C}^i_B[\mathbf{f}^i(\mathbf{m}^i, \mathbf{m}^B) - \mathbf{u}^i]$ (given by definition) into the previous equation, it is obtained $(1/\eta)\dot{\mathbf{m}}^B > \mathbf{1}\epsilon - \mathbf{m}^B$. This means that, in case $\mathbf{m}^B[j] \le \epsilon$ for some b_j, the field vector of the marking of b_j points towards a marking $\mathbf{m}[b_j] > \epsilon$, i.e., ψ is computed in such a way that the buffers remain marked (the analysis can also be achieved in discrete-time, obtaining the analogous expression $(1/\eta)(\mathbf{m}^B_{\tau+1} - \mathbf{m}^B_\tau)/\Delta\tau > \mathbf{1}\epsilon - \mathbf{m}^B_\tau$, having the same interpretation).
Now, suppose that $\psi = 0$. Then, as shown for the centralized affine control scheme, there exists a quadratic Lyapunov function $\mathbf{V}(\mathbf{m}^i) = (\mathbf{m}^i - \mathbf{m}^i_f)^T\mathbf{P}(\mathbf{m}^i - \mathbf{m}^i_f)$ whose derivative is negative, i.e., the matrix $\mathbf{Q} = -[(\mathbf{CF})^T\mathbf{P} + \mathbf{P}(\mathbf{CF})]$ is positive definite and so $\dot{\mathbf{V}}(\mathbf{m}^i) = -\eta(\mathbf{m}^i - \mathbf{m}^i_f)^T\mathbf{Q}(\mathbf{m}^i - \mathbf{m}^i_f) < 0$, $\forall \mathbf{m}^i \neq \mathbf{m}^i_f$. In general, independently of the values received from the coordinator, the derivative of the same Lyapunov function with $\psi \neq 0$ can be computed as $\dot{\mathbf{V}}(\mathbf{m}^i) = -\eta(1-\psi)(\mathbf{m}^i - \mathbf{m}^i_f)^T\mathbf{Q}(\mathbf{m}^i - \mathbf{m}^i_f) - \eta\psi(\mathbf{m}^i - \mathbf{m}^i_f)^T\mathbf{P}(\mathbf{m}^i - \mathbf{m}^i_f)$. Since $\mathbf{P}$ and $\mathbf{Q}$ are definite positive and $\eta > 0$, then the derivative of the Lyapunov function is negative for $\psi \in [0, 1]$, and so the module will asymptotically converge to $\mathbf{m}^i_f$ (a similar result can be obtained for the discrete-time case, by using the Lyapunov function $\Delta\mathbf{V}(\mathbf{m}^i_\tau) = (\mathbf{m}^i_{\tau+1} - \mathbf{m}^i_f)^T\mathbf{P}(\mathbf{m}^i_{\tau+1} - \mathbf{m}^i_f) - (\mathbf{m}^i_\tau - \mathbf{m}^i_f)^T\mathbf{P}(\mathbf{m}^i_\tau - \mathbf{m}^i_f)$ and assuming $\Delta\tau << 1$). □



(a) (b)

Fig. 5. (a) Marking of Module 1 of the system of fig. 3 in closed-loop behavior, under the coordinated control of *Procedure 2*. (b) Marking in the buffers.

Example 3. Consider again the *TCPN* system depicted in fig. 3 with the same rates and initial marking. The desired markings for the modules are given by $\mathbf{m}^1_f = [1, 1, 1, 7]$, $\mathbf{m}^2_f = [0.5, 3, 1.5]$ and $\mathbf{m}^3_f = [1.5, 0.5, 3]$, respectively. Consider a desired marking for the buffers $\mathbf{m}^B_f = [0.5, 0.5, 0.5, 0.5]^T$. For each module, gain matrices $\mathbf{F}^i$ and vectors $\mathbf{g}^i$ were computed according to *Procedure 1*. Later, the coordinated control strategy of *Procedure 2* was applied. Fig. 5(a) shows the trajectories for the marking of $\mathcal{N}^1$. It can be observed that the module was successfully driven to the desired marking $\mathbf{m}^1_f$. Similarly, modules $\mathcal{N}^2$ and $\mathcal{N}^3$ were driven towards $\mathbf{m}^2_f$ and $\mathbf{m}^3_f$, respectively. Fig. 5(b) shows the marking evolution of the buffers. Notice that the control makes the buffers remain marked by converging to $\mathbf{m}^B_f > \mathbf{1}\epsilon$.

6. CONCLUSIONS

In this work, conditions for an affine centralized control law for *PWA* systems in polytopes is derived and applied to *TCPN*s. Later, those results are extended in order to provide a modular-coordinated control strategy. The resulting scheme consists of a set of affine local controllers and a coordinator that receives and sends information to the local controllers. Feasibility and convergency to the required markings have been proved. Furthermore, the application of the control scheme is achieved in polynomial time (the complexity appears during the synthesis of the local controllers, i.e., off-line, since the computation of the vertices of a polytope is not polynomial). It is left for a future research, to investigate the performance of the proposed control scheme under lost or delayed information conditions. An interesting future extension of the presented work would be the control of systems with uncontrollable transitions. Nevertheless, this is not a trivial task, since the controllability in such case is much more complex (Vázquez et al. [2008]).

REFERENCES

H. Alla and R. David, "Continuous and hybrid Petri nets", *Journal of Circuits, Systems, and Computers*, vol. 8, no. 1, pp. 159–188, 1998.

A. Amrah, N. Zerhouni, A. El-Moudni, "Control of discrete event systems modeled by continuous Petri Nets: case of opened assembly manufacturing lines", *Proc. of ECC'97, European Control Conf. (Brussels, Belgium)*, 1997.

H. Apaydin-Özkan, J. Júlvez, C. Mahulea, M. Silva, "A control method for timed distributed continuous Petri nets", *Proc. of the 2010 American Control Conference, Baltimore, USA*, 2010.

C. Chen, *Linear system theory and design.* Oxford University Press, 1984.

K. Fukuda, "Frequently asked questions in polyhedral computation", Zürich. http://www.ifor.math.ethz.ch/~fukuda/polyfaq/polyfaq.html.

S. Hennequin, D. Lefebvre and A. El-Moudni, "Fuzzy control of variable speed continuous Petri nets", *Proccedings of the 38th IEEE Conference on Decision and Control*, vol. 2, pp. 1352–1356, 1999.

R. Kara, M. Ahmane, J.J. Loiseau and S. Djennoune, "Constrained regulation of continuous Petri nets ", *Nonlinear Analysis: Hybrid Systems*, vol. 3, no. 4, pp. 738–748, 2009.

D. Lefebvre, D. Catherine, E. Leclerq, F. Druaux, "Some contributions with Petri nets for the modelling, analysis and control of HDS ", *Proccedings of the International Conference on Hybrid Systems and Applications, LA, USA*, vol. 1, no. 4, pp. 451–465, 2007.

C. Mahulea, A. Giua, L. Recalde, C. Seatzu, and M. Silva, "Optimal model predictive control of Timed Continuous Petri nets ", *IEEE Transactions on Automatic Control*, vol. 53, no. 7, pp. 1731–1735, 2008.

C. Mahulea, L. Recalde and M. Silva, "Basic Server Semantics and Performance Monotonicity of Continuous Petri Nets", *Discrete Event Dynamic Systems: Theory and Applications*, vol. 19, no. 2, pp. 189 – 212, 2008.

M. Silva, and L. Recalde, "Petri nets and integrality relaxations: A view of continuous Petri nets ", *IEEE Trans. on Systems, Man, and Cybernetics*, vol. 32, no. 4, pp. 314–327, 2002.

L.C.G.J.M. Habets, J.H. van Schuppen, "A control problem for affine dynamical systems on a full-dimensional polytope", *Automatica*, vol. 40, pp 21–35, 2004.

L.C.G.J.M. Habets, P.J. Collins, J.H. van Schuppen, "Reachability and control synthesis for piecewise-affine hybrid systems on simplices ", *IEEE Trans. Automatic Control*, vol. 51, pp. 938–948, 2006.

C.R. Vázquez, A. Ramírez, L. Recalde and M. Silva, "On controllability of timed continuous Petri nets ", *In Egerstedt M., Mishra B. eds.: Proceedings of the 11th International Workshop Hybrid Systems: Computational and Control (HSCC08)*, vol. 4981, pp. 528-541, 2008.

C.R. Vázquez and M. Silva, "Performance Control of Markovian Petri Nets via Fluid Models: A Stock-Level Control Example ", *In 5th IEEE Conference on Automation Science and Engineering (IEEE CASE)*, Bangalore, India, pp. 30–36, 2009.

Appendix A. ALGORITHMS

The computation of $\mathbf{m}_f^B$, achieved during the synthesis stage in *Procedure 2*, can be done by setting any $\mathbf{m}_f^B$ fulfilling two linear constraints (so it can be computed in polynomial time): i) $\mathbf{m}_f^B > \mathbf{1}\epsilon$, ii) $\mathbf{B}_y^T[B]\mathbf{m}_f^B = \mathbf{B}_y^T[B]\mathbf{m}_0^B + \sum \mathbf{B}_y^T[P^i](\mathbf{m}_0^i - \mathbf{m}_f^i)$. This equality constraint is equivalent to $\mathbf{B}_y^T\mathbf{m}_f = \mathbf{B}_y^T\mathbf{m}_0$, where $\mathbf{m}_f$ is the global final marking. In this way, $\mathbf{m}_f \in Class_\epsilon(\mathbf{m}_0)$. Thus, since the net is consistent, $\mathbf{m}_f$ is reachable.

On the other hand, the computation of $\eta_{x_j}^i$, η_w^i, η_d^i, η_T^i and ψ in *Procedure 2* is equivalent to solving: maximum α s.t. $\mathbf{a}\alpha \leq \mathbf{b}$, with $\mathbf{a}, \mathbf{b} \geq \mathbf{0}$. For computing this, define the set of indices $S = \{i | [\mathbf{a}]_i > 0\}$. Next, if there exists a solution, this is given by $\alpha = min\{[\mathbf{b}]_i/[\mathbf{a}]_i | i \in S\}$.

The computation of $\mathbf{d}^i$ and $\boldsymbol{\gamma}$ in *Procedure 2* is equivalent to the problem of solving $\mathbf{x} > \mathbf{0}$ s.t. $\mathbf{Ax} = \mathbf{b}$, assuming that there exists a solution and $\exists \mathbf{a} > \mathbf{0}$ s.t. $\mathbf{Aa} = \mathbf{0}$. For the computation of $\mathbf{d}^i$ and $\boldsymbol{\gamma}$, the matrices that are represented by $\mathbf{A}$ are fixed. On the contrary, the vectors represented by $\mathbf{b}$ change during the evolution of the system.

Procedure 3. Computation of $\mathbf{x} > \mathbf{0}$ s.t. $\mathbf{Ax} = \mathbf{b}$.

==================================

Computation off-line:

==================================

- Compute permutation matrices $\mathbf{P}_1$ and $\mathbf{P}_2$ so
$$\mathbf{P}_1\mathbf{A}\mathbf{P}_2 = \begin{bmatrix} \mathbf{A}_{11} & \mathbf{A}_{12} \\ \mathbf{A}_{21} & \mathbf{A}_{22} \end{bmatrix},$$
$\mathbf{A}_{11}$ is invertible and has the same rank, ρ, that $\mathbf{A}$.

==================================

Operation on-line:

==================================

I . Define $\mathbf{b}' = \mathbf{P}_1^{-1}\mathbf{b}$ and $\mathbf{b}'_1$ as the vector built with the first ρ elements of $\mathbf{b}'$.

II . Compute $\mathbf{x}^p = \mathbf{P}_2 \cdot [(\mathbf{A}_{11}^{-1}\mathbf{b}'_1)^T, \mathbf{0}^T]^T$.

III. Taking any scalar $\alpha > |min(\mathbf{x}'./\mathbf{a})|$ (where the division of vectors is element-wise), then $\mathbf{x} = \mathbf{x}^p + \alpha\mathbf{a} > \mathbf{0}$ and it is a solution for $\mathbf{Ax} = \mathbf{b}$.

==================================

Proof. Since $\exists \mathbf{x}^p$ s.t. $\mathbf{Ax}^p = \mathbf{b}$, then $\exists \mathbf{x}' = \mathbf{P}_2^{-1}\mathbf{x}^p$ s.t. $\mathbf{P}_1\mathbf{AP}_2\mathbf{x}' = \mathbf{b}'$, where $\mathbf{b}' = \mathbf{P}_1\mathbf{b}$. On the other hand, the rows of $[\mathbf{A}_{21}, \mathbf{A}_{22}]$ are linear combinations of those of $[\mathbf{A}_{11}, \mathbf{A}_{12}]$. Combining these two facts, $\mathbf{x}'$ that fulfills $[\mathbf{A}_{11}, \mathbf{A}_{12}]\mathbf{x}' = \mathbf{b}'_1$ is also a solution for $\mathbf{P}_1\mathbf{AP}_2\mathbf{x}' = \mathbf{b}'$. Then, $\mathbf{x}' = [(\mathbf{A}_{11}^{-1}\mathbf{b}'_1)^T, \mathbf{0}^T]^T$ is a particular solution of $\mathbf{P}_1\mathbf{AP}_2\mathbf{x}' = \mathbf{b}'$, and so, $\mathbf{x}^p = \mathbf{P}_2\mathbf{x}'$ is a particular solution for $\mathbf{Ax}^p = \mathbf{b}$. Finally, given α s.t. $\alpha\mathbf{a} > |\mathbf{x}'|$ and $\mathbf{Aa} = \mathbf{0}$, then $\mathbf{x} = \mathbf{x}^p + \alpha\mathbf{a}$ is a positive solution for $\mathbf{Ax}^p = \mathbf{b}$. □

Notice that $\mathbf{b}'$ and $\mathbf{x}^p$ are just evaluations, while for α it is only required to find the minimum element of a vector, so the complexity is linear in the number of elements of $\mathbf{x}$.