Decentralized Supervisory Control of Petri Nets with Monitor Places

Francesco Basile, Alessandro Giua, Carla Seatzu

Abstract

In this paper we study the problem of determining a set of decentralized monitors for place/transition nets to enforce a global specification on the net behavior given in terms of Generalized Mutual Exclusion Constraints (GMECs). We generalize our previous results in this topic. In particular, the novel contribution here consists in removing the restrictive assumption that the weights of the GMECs must be positive, while we still assume that all transitions are controllable and observable, and the support of each decentralized GMEC is a singleton. The main feature of the proposed solution is that it guarantees fairness among places.

Published as: F. Basile, A. Giua, C. Seatzu, "Decentralized supervisory control of Petri nets with monitor places," 3^{rd} Annual IEEE Conf. on Automation Science and Engineering (Scottsdale, Arizona), September 2007.

Francesco Basile is with Dip. Ing. dell'Informazione e Ing. Elettrica, Università di Salerno, Via Ponte don Melillo, 84084 Fisciano (Salerno), Italy. `fbasile@unisa.it`

Alessandro Giua, Carla Seatzu are with the Dip. di Ing. Elettrica ed Elettronica, Università di Cagliari, Piazza d'Armi, 09123 Cagliari, Italy. `{giua,seatzu}@diee.unica.it`

I. INTRODUCTION

In the context of supervisory control of Discrete Event Systems (DES) decentralized control problems have received a great attention in the last decade [10]. Several original approaches have been proposed to solve this problem by means of formal languages approaches using automata [1], [8], [11], [12]. On the contrary, Petri Nets (PNs) have not received much attention in this context. Their compact state representation and their intrinsically distributed nature may potentially help in reducing the complexity of decentralized supervisory control problems.

As control specification we consider a state predicate formulation. In particular, we study the problem of determining a set of local supervisors when the global specification is given by a set of Generalized Mutual Exclusion Constraints (GMECs) $(\boldsymbol{W}, \boldsymbol{k})$, where $\boldsymbol{W} = [\boldsymbol{w}_1^T, \boldsymbol{w}_2^T, ...\boldsymbol{w}_{n_c}^T]^T$ and $\boldsymbol{k} = [k_1, k_2, ...k_{n_c}]^T$ and then the set of legal markings is $\mathcal{M}(\boldsymbol{W}, \boldsymbol{k}) = \{\boldsymbol{m} \in \mathbb{N}^m \mid \boldsymbol{W} \cdot \boldsymbol{m} \leq \boldsymbol{k}\}$. Few works have considered a state predicates formulation for decentralized control. In [6] global specifications are implemented by local supervisors with communication. In [4] a central coordinator is also present but specifications are assumed to be given from the beginning in distributed form. In [7] global specifications without central coordination are considered and a sufficient condition is given for a state predicate formulated in terms of GMECs to be enforced in a decentralized setting (d-admissibility); the transformation of inadmissible decentralized constraints into admissible ones is posed either in terms of the minimization of communication costs or in terms of the transformation of the constraints into a set of more restrictive ones but d-admissible.

A control architecture without central coordinator and communication between local supervisors is here considered. The set of places is partitioned into ν disjoint sets P_j, and the j-th local supervisor may enforce only places in P_j to assume a certain set of values.

In [2] under the assumption that (A1) all weights are positive, i.e., $\boldsymbol{W} \geq \boldsymbol{0}$, $\boldsymbol{k} \geq \boldsymbol{0}$, (A2) all transitions are controllable and observable, (A3) the support of each decentralized GMEC is a singleton, thus $\nu = m$ and $P_j = \{p_j\}$, for $j = 1, \ldots, m$, it was shown that this problem can be solved by computing an *integer inner box* $\mathcal{I}(\boldsymbol{u}) = \{\boldsymbol{m} \in \mathbb{N}^m \mid \boldsymbol{m} \leq \boldsymbol{u}\}$ included in the set of legal markings defined by the global GMEC $\mathcal{M}(\boldsymbol{W}, \boldsymbol{k})$.

In this paper assumption (A1) is removed, i.e. the weights of GMECs may also be negative. It is shown that the problem can be solved by computing an *integer inner box* $\mathcal{B}(\boldsymbol{l}, \boldsymbol{u}) = \{\boldsymbol{m} \in$

$\mathbb{N}^m \mid \boldsymbol{l} \leq \boldsymbol{m} \leq \boldsymbol{u}\}$ included in the set of legal markings defined by the global GMEC $\mathcal{M}(\boldsymbol{W}, \boldsymbol{k})$. In particular, the problem of finding a *maximal integer inner box* $\mathcal{B} \subseteq \mathcal{M}(\boldsymbol{W}, \boldsymbol{k})$, i.e. an inner box such that there does not exist a box $\tilde{\mathcal{B}} \neq \mathcal{B}$ and $\mathcal{B} \subsetneq \tilde{\mathcal{B}} \subseteq \mathcal{M}(\boldsymbol{W}, \boldsymbol{k})$ is here considered. A solution that aims to guarantee fairness among places, and that can be computed using a simple iterative algorithm, is proposed.

II. Basic definitions

A. Petri nets

In this section we recall the formalism used in the paper. For more details on Petri nets we address to [9].

A Place/Transition (P/T) net is a structure $N = (P, T, \boldsymbol{Pre}, \boldsymbol{Post})$ where: P is a set of m *places* represented by circles; T is a set of n *transitions* represented by bars; $P \cap T = \emptyset$, $P \cup T \neq \emptyset$; $\boldsymbol{Pre}$ ($\boldsymbol{Post}$) is the $m \times n$ sized, natural valued, pre-(post-)incidence matrix. For instance, $\boldsymbol{Pre}(p, t) = w$ (resp., $\boldsymbol{Post}(p, t) = w$) means that there is an arc from p to t (resp., from t to p) with weight w. The incidence matrix $\boldsymbol{C}$ of the net is defined as $\boldsymbol{C} = \boldsymbol{Post} - \boldsymbol{Pre}$. A *marking* is a $m \times 1$ vector $\boldsymbol{m} : P \rightarrow \mathbb{N}$ that assigns to each place of a P/T net a non-negative integer number of tokens. A P/T system or net system $\langle N, \boldsymbol{m}_0 \rangle$ is a P/T net N with an initial marking $\boldsymbol{m}_0$. A transition $t \in T$ is enabled at a marking $\boldsymbol{m}$ iff $\boldsymbol{m} \geq \boldsymbol{Pre}(\cdot, t)$. If t is enabled, then it may fire yielding a new marking $\boldsymbol{m}' = \boldsymbol{m} + \boldsymbol{Post}(\cdot, t) - \boldsymbol{Pre}(\cdot, t) = \boldsymbol{m} + \boldsymbol{C}(\cdot, t)$. The notation $\boldsymbol{m}[t\rangle\boldsymbol{m}'$ means that an enabled transition t may fire at $\boldsymbol{m}$ yielding $\boldsymbol{m}'$. A *firing sequence* from $\boldsymbol{m}_0$ is a (possibly empty) sequence of transitions $\sigma = t_1, \ldots, t_k$ such that $\boldsymbol{m}_0[t_1\rangle\boldsymbol{m}_1[t_2\rangle\boldsymbol{m}_2 \ldots [t_k\rangle\boldsymbol{m}_k$. A marking $\boldsymbol{m}$ is reachable in $\langle N, \boldsymbol{m}_0 \rangle$ iff there exists a firing sequence σ such that $\boldsymbol{m}_0[\sigma\rangle\boldsymbol{m}$. Given a net system $\langle N, \boldsymbol{m}_0 \rangle$ the set of reachable markings is denoted $R(N, \boldsymbol{m}_0)$.

B. Generalized Mutual Exclusion Constraint

A *Generalized Mutual Exclusion Constraint* (GMEC) is a couple $(\boldsymbol{w}, k)$ where $\boldsymbol{w} : P \rightarrow \mathbb{Z}$ is an m dimensional row vector and $k \in \mathbb{Z}$. A GMEC defines a set of *legal markings*:

$$\mathcal{M}(\boldsymbol{w}, k) = \{\boldsymbol{m} \in \mathbb{N}^m \mid \boldsymbol{w} \cdot \boldsymbol{m} \leq k\}.$$

The markings that are not legal are called *forbidden markings*. A controlling agent, called *supervisor*, must ensure the forbidden markings will be not reached. So the set of legal markings under control is $\mathcal{M}_c(\boldsymbol{w}, k) = \mathcal{M}(\boldsymbol{w}, k) \cap R(N, \boldsymbol{m}_0)$. We call *support* of $(\boldsymbol{w}, k)$ the set

$Q_{\boldsymbol{w}} = \{p \in P | \boldsymbol{w}(p) \neq 0\}$. A set of GMECs $(\boldsymbol{W}, \boldsymbol{k})$, with $\boldsymbol{W} = [\boldsymbol{w}_1^T, \boldsymbol{w}_2^T, ...\boldsymbol{w}_{n_c}^T]^T$, and $\boldsymbol{k} = [k_1, k_2, ...k_{n_c}]^T$, defines the set of legal markings $\mathcal{M}(\boldsymbol{W}, \boldsymbol{k}) = \{\boldsymbol{m} \in \mathbb{N}^m \mid \boldsymbol{W} \cdot \boldsymbol{m} \leq \boldsymbol{k}\}$. We call *support* of $(\boldsymbol{W}, \boldsymbol{k})$ the set $Q_{\boldsymbol{W}} = \{p \in P \cap \left(\cup_{j=1}^{n_c} Q_{\boldsymbol{w}_j}\right)\}$.

It has been shown in [5] that a set of n_c GMECs can be enforced adding to the controlled net a set of n_c places called *monitors*, provided that the initial marking is legal. A simple rule to determine the monitors that guarantee the maximal permissiveness of the closed loop net was also given in [5], under the assumption that all transitions are controllable and observable.

C. Geometrical definitions

We represent a *convex polyedron* as $\mathcal{P} = \{\boldsymbol{x} \in \mathbb{R}^d | \boldsymbol{A}\boldsymbol{x} \leq \boldsymbol{b}\}$, where $\boldsymbol{A}$ is a real $r \times d$ matrix and $\boldsymbol{b}$ is a real d-vector. The set of legal markings defined by $\mathcal{M}(\boldsymbol{W}, \boldsymbol{k})$ is included in a *convex polyedron*. An *interior point* of $\mathcal{M}(\boldsymbol{W}, \boldsymbol{k})$ is a point $\hat{\boldsymbol{m}}$ such that $\boldsymbol{W}\hat{\boldsymbol{m}} < \boldsymbol{k}$. A polyedron if full dimensional if it has an interior point; otherwise, it is embedded in a lower dimensional affine space.

A *box* is a set of real vectors defined as $\mathcal{B}(\boldsymbol{l}, \boldsymbol{u}) = \{\boldsymbol{x} \in \mathbb{R}^d | \boldsymbol{l} \leq \boldsymbol{x} \leq \boldsymbol{u}\}$, where $\boldsymbol{l}$ and $\boldsymbol{u}$ are real d-vectors.

If $\boldsymbol{x} \in \mathbb{Z}^d$ we call $\mathcal{B}(\boldsymbol{l}, \boldsymbol{u})$ *integer box*. If $\boldsymbol{l} = \boldsymbol{0}$, we call $\mathcal{B}(\boldsymbol{0}, \boldsymbol{u})$ *positive integer box* and we denote it simply as $\mathcal{I}(\boldsymbol{u})$.

An *hypercube* is a box such that $\boldsymbol{u} = \boldsymbol{l} + \lambda\boldsymbol{e}$, where λ is a scalar and $\boldsymbol{e}$ denotes the d-vector of ones.

III. Problem statement

Let $\langle N, \boldsymbol{m}_{p0} \rangle$ be a P/T system to be controlled, where $N = (P, T, \boldsymbol{Pre}, \boldsymbol{Post})$.

Assume that a global specification is given in terms of a GMEC $(\boldsymbol{W}, \boldsymbol{k})$. Without loss of generality we take $Q_{\boldsymbol{W}} = P$, i.e., all places are bounded by the constraint. If such is not the case, we can simply apply the proposed procedure to the projection on $Q_{\boldsymbol{W}}$.

Assume that the set of places P is partitioned into ν subsets $P_1, \ldots, P_\nu$, i.e., $P_i \cap P_j = \emptyset$ if $i \neq j$, and $\cup_{i=1}^{\nu} P_i = P$.

We want to determine a set of *decentralized* GMECs $(\boldsymbol{W}^{(i)}, \boldsymbol{k}^{(i)})$ whose support is P_i, with $i = 1, \ldots, \nu$, such that

$$\cap_{i=1}^{\nu} \mathcal{M}(\boldsymbol{W}^{(i)}, \boldsymbol{k}^{(i)}) \subseteq \mathcal{M}(\boldsymbol{W}, \boldsymbol{k}). \quad (1)$$

The choice of the decentralized GMECs is obviously not unique, and depends in general on the dimension $n_c^{(i)}$ of the decentralized GMECs $(\boldsymbol{W}^{(i)}, \boldsymbol{k}^{(i)})$.

In this paper we make the following assumptions:

(A1) All transitions are controllable and observable.

(A2) The support of each decentralized GMEC is a singleton, thus $\nu = m$ and $P_i = \{p_i\}$, for $i = 1, \ldots, m$.

By assumption (A2) it follows that $n_c^{(i)} = 2$ and the effect of each decentralized GMEC is that of imposing a lower bound and an upper bound on the corresponding place. Thus, the set

$$\cap_{i=1}^{\nu} \mathcal{M}(\boldsymbol{W}^{(i)}, \boldsymbol{k}^{(i)})$$

can be regarded as the integer box $\mathcal{B}(\boldsymbol{l}, \boldsymbol{u})$ where $\boldsymbol{l}$ and $\boldsymbol{u}$ are m-integer vectors whose i-th component denotes respectively the lower and upper bound induced by the i-th and the $i+m$-th, respectively, decentralized GMEC on place p_i.

Our goal here is that of trying to determine a systematic procedure to select $\boldsymbol{l}$ and $\boldsymbol{u}$ in order to guarantee fairness among places.

IV. Preliminary results

In this section we present some definitions and preliminary results that will be useful in the following.

We denote as $\mathcal{C}(\boldsymbol{A}, \boldsymbol{b})$ a generic convex set containing the origin, i.e., $\mathcal{C}(\boldsymbol{A}, \boldsymbol{b}) = \{\boldsymbol{m} \in \mathbb{R}^m \mid \boldsymbol{A}\boldsymbol{m} \leq \boldsymbol{b}\}$ where $\boldsymbol{A} \in \mathbb{Z}^{n_c \times m}$, and $\boldsymbol{b} \in \mathbb{N}^{n_c}$. Moreover, we denote as $\boldsymbol{a}_i$ the i-th row of matrix $\boldsymbol{A}$, and b_i the i-th component of vector $\boldsymbol{b}$. Finally, since $\boldsymbol{a}_i$ is a row vector of dimension m, and in the rest of the paper we are interested in the case of m equal to the number of places, we denote as $a_i(p)$ the component of vector $\boldsymbol{a}_i$ relative to place p.

Definition 1: An integer box $\mathcal{B}(\boldsymbol{l}, \boldsymbol{u}) \subseteq \mathcal{C}(\boldsymbol{A}, \boldsymbol{b})$ is a *maximal integer inner box* if there does not exist an inner box $\mathcal{B}(\tilde{\boldsymbol{l}}, \tilde{\boldsymbol{u}}) \neq \mathcal{B}(\boldsymbol{l}, \boldsymbol{u})$ such that $\mathcal{B}(\boldsymbol{l}, \boldsymbol{u}) \subsetneq \mathcal{B}(\tilde{\boldsymbol{l}}, \tilde{\boldsymbol{u}}) \subseteq \mathcal{C}(\boldsymbol{A}, \boldsymbol{b})$. ■

Note that the maximal integer inner box is in general not unique.

Definition 2: Let us consider an integer box $\mathcal{B}(\boldsymbol{l}, \boldsymbol{u})$ included in a convex set $\mathcal{C}(\boldsymbol{A}, \boldsymbol{b})$, where both $\mathcal{B}(\boldsymbol{l}, \boldsymbol{u})$ and $\mathcal{C}(\boldsymbol{A}, \boldsymbol{b})$ contain the origin. We denote as $\boldsymbol{x}$ the vector defined as follows

$$x_i(p) = \begin{cases} l(p), & \text{if } a_i(p) \leq 0, \\ u(p), & \text{if } a_i(p) > 0. \end{cases}$$ ■

Proposition 3: Let us consider an integer box $\mathcal{B}(\boldsymbol{l},\boldsymbol{u})$ included in a convex set $\mathcal{C}(\boldsymbol{A},\boldsymbol{b})$, where both $\mathcal{B}(\boldsymbol{l},\boldsymbol{u})$ and $\mathcal{C}(\boldsymbol{A},\boldsymbol{b})$ contain the origin. The integer box $\mathcal{B}(\boldsymbol{l},\boldsymbol{u})$ is a *maximal integer inner box* in $\mathcal{C}(\boldsymbol{A},\boldsymbol{b})$, if and only if $\forall\ p \in P$:

$$0 \leq \min_{i\in\{1,\ldots,n_c\}} \frac{b_i - \sum_{p\in P} a_i(p)x_i(p)}{|a_i(p)|} < 1 \tag{2}$$

Proof: The first part of the inequality is trivial because it ensures the satisfaction of the constraints defining $\mathcal{C}(\boldsymbol{A},\boldsymbol{b})$. Let us now discuss the second part of the inequality.

(if) Let p be any place in P, and $\bar{\imath}$ be the corresponding value of index i that minimizes (2). It holds

$$\frac{b_{\bar{\imath}} - \sum_{p\in P} a_{\bar{\imath}}(p)x_{\bar{\imath}}(p)}{|a_{\bar{\imath}}(p)|} < 1 \quad \Rightarrow b_{\bar{\imath}} - \sum_{p\in P} a_{\bar{\imath}}(p)x_{\bar{\imath}}(p) < |a_{\bar{\imath}}(p)|.$$

Now, since by assumption $\mathcal{B}(\boldsymbol{l},\boldsymbol{u})$ contains the origin, then $\boldsymbol{l} \leq \boldsymbol{0}$ and $\boldsymbol{u} \geq \boldsymbol{0}$. Moreover,

- if $a_{\bar{\imath}}(p) < 0$ then $\boldsymbol{a}_{\bar{\imath}} \cdot \tilde{\boldsymbol{x}}_{\bar{\imath}} > b_{\bar{\imath}}$ where $\tilde{x}_{\bar{\imath}}(p_j) = l_{\bar{\imath}}(p_j)$ for all $p_j \neq p$, and $\tilde{x}_{\bar{\imath}}(p) = l(p) - 1$;
- if $a_{\bar{\imath}}(p) \geq 0$ then $\boldsymbol{a}_{\bar{\imath}} \cdot \tilde{\boldsymbol{x}}_{\bar{\imath}} > b_{\bar{\imath}}$ where $\tilde{x}_{\bar{\imath}}(p_j) = u_{\bar{\imath}}(p_j)$ for all $p_j \neq p$, and $\tilde{x}_{\bar{\imath}}(p) = u(p) + 1$.

This means that, if the lower or the upper bound on p is increased or decreased of one unity this would lead to the violation of the $\bar{\imath}$-th GMEC. Since this is true for any place $p \in P$, we conclude that $\mathcal{B}(\boldsymbol{l},\boldsymbol{u})$ is a maximal inner box in $\mathcal{C}(\boldsymbol{A},\boldsymbol{b})$.

(only if) We prove this by contradiction. Assume that $\mathcal{B}(\boldsymbol{l},\boldsymbol{u})$ is a maximal integer inner box in $\mathcal{C}(\boldsymbol{A},\boldsymbol{b})$, but $\exists\ p \in P$ such that

$$\min_{i\in\{1,\ldots,n_c\}} \frac{b_i - \sum_{p\in P} a_i(p)x_i(p)}{|a_i(p)|} \geq 1.$$

This implies that $\forall i \in \{1,\ldots,n_c\}$, $b_i - \boldsymbol{a}_i \cdot \boldsymbol{x}_i \geq |a_i(p)|$. Thus, given an arbitrary place $p \in P$, we can define a vector $\tilde{\boldsymbol{x}}_i$ as in the previous statement that satisfies all the constraints, i.e., such that $\boldsymbol{a}_i \cdot \tilde{\boldsymbol{x}}_i \leq b_i$ for all $i = 1,\ldots,n_c$. This clearly leads to a contradiction. □

In simple words the above proposition means that an integer box $\mathcal{B}(\boldsymbol{l},\boldsymbol{u})$ is maximal if and only if in each direction there exists at least one constraint that is saturated.

Finally, the following proposition provides a criterion to determine the *maximal integer hypercube* in $\mathcal{C}(\boldsymbol{A},\boldsymbol{b})$ with center in the origin.

Proposition 4: Let $\bar{\mathcal{C}} = \mathcal{C}(\boldsymbol{A},\boldsymbol{b})$ be a convex set containing the origin and thus $\boldsymbol{b} \geq \boldsymbol{0}$. Let us denote as

$$\tau(\bar{\mathcal{C}}) = \max\ \{\tau \in \mathbb{N} \mid \mathcal{B}(-\tau\boldsymbol{e}, \tau\boldsymbol{e}) \subseteq \bar{\mathcal{C}}\}.$$

It holds $\tau(\bar{\mathcal{C}}) = \min_{i=1,\ldots,n_c} \tau(i, \bar{\mathcal{C}})$ where

$$\tau(i, \bar{\mathcal{C}}) = \left\lfloor \frac{b_i}{\sum_{p \in P} |a_i(p)|} \right\rfloor$$

and $\lfloor \, \rfloor$ denotes the floor operator.

Proof: The above statement follows from a result presented in [3] where the problem of maximizing the volume of hypercubes included in polytopes was considered. Note however that in [3] the floor operator was not present. It is used here being $\mathcal{B}(-\tau \boldsymbol{e}, \tau \boldsymbol{e})$ an integer hypercube.

■

V. The proposed solution to the decentralized control problem

In this section we discuss the main steps of our solution to the decentralized control problem presented in Section III, that consists in determining the maximal integer inner box $\mathcal{B}(\boldsymbol{l}, \boldsymbol{u}) \subseteq \mathcal{M}(\boldsymbol{W}, \boldsymbol{k})$ under assumptions (A1) and (A2).

The main idea behind the proposed approach may be summarized in the following steps.

(S1) We first determine an appropriate interior point $\boldsymbol{c} \in \mathcal{M}(\boldsymbol{W}, \boldsymbol{k})$.

(S2) We define a new coordinate system centered on $\boldsymbol{c}$. By mapping $\boldsymbol{c}$ into the origin with the coordinate translation $\boldsymbol{m}' = \boldsymbol{m} - \boldsymbol{c}$, the set of legal markings $\mathcal{M}(\boldsymbol{W}, \boldsymbol{k})$ is transformed into the equivalent one $\mathcal{M}(\tilde{\boldsymbol{W}}', \tilde{\boldsymbol{k}}') = \{\boldsymbol{m}' \in \mathbb{N}^m \mid \tilde{\boldsymbol{W}}' \cdot \boldsymbol{m}' \leq \tilde{\boldsymbol{k}}'\}$ with $\tilde{\boldsymbol{W}}' = \boldsymbol{W}$ and $\tilde{\boldsymbol{k}}' = \boldsymbol{k} - \boldsymbol{W} \cdot \boldsymbol{c}$. Moreover, since $\boldsymbol{m}$ is a net marking vector, it can only assume positive values. Thus, we need to impose an additional constraint, namely $-\boldsymbol{m}' \leq \boldsymbol{c}$.

We denote by $(\tilde{\boldsymbol{W}}, \tilde{\boldsymbol{k}})$ the resulting set of GMECs given by $(\boldsymbol{W}, \boldsymbol{k})$ plus the GMECs corresponding to the non-negativity constraints.

Notice that $\tilde{\boldsymbol{k}}' \geq \boldsymbol{0}$ since the origin belongs to the set $(\tilde{\boldsymbol{W}}', \tilde{\boldsymbol{k}}')$.

(S3) We generalize our results in [2] to compute the maximal integer inner box in $\mathcal{M}(\tilde{\boldsymbol{W}}, \tilde{\boldsymbol{k}})$. This point is discussed in detail in the following Section VI.

(S4) Using the inverse coordinate transformation $\boldsymbol{m} = \boldsymbol{m}' + \boldsymbol{c}$ we determine the decentralized GMECs for the original Petri net system.

Let us now discuss step S1.

A. Interior point determination

Different criteria can be chosen to appropriately select an interior point $\boldsymbol{c}$ in $\mathcal{M}(\boldsymbol{W}, \boldsymbol{k})$. In this paper we suggest to select $\boldsymbol{c}$ as the interior point of $\mathcal{M}(\boldsymbol{W}, \boldsymbol{k})$ that coincides with the center of the maximal integer hypercube in $\mathcal{M}(\boldsymbol{W}, \boldsymbol{k})$. This choice is motivated by our requirement of guaranteeing the maximal fairness among places.

In such a case, $\boldsymbol{c}$ can be easily computed by solving a linear integer programming problem (LIPP) as proved by the following proposition.

Proposition 5: Let us consider a set of legal markings $\mathcal{M}(\boldsymbol{W}, \boldsymbol{k})$. Assume that $\mathcal{M}(\boldsymbol{W}, \boldsymbol{k})$ is bounded in at least one direction p_i, $i = 1, \ldots, m$.

The center $\boldsymbol{c}$ and the edge 2τ of the maximal integer hypercube in $\mathcal{M}(\boldsymbol{W}, \boldsymbol{k})$ can be computed by solving the following LIPP:

$$\max_{\boldsymbol{c},\tau} \tau \tag{3}$$

$$s.t. \quad \begin{cases} \boldsymbol{w}_i \ \boldsymbol{c} + \sum_{p \in P} |w_i(p)| \tau \leq k_i \quad \forall i = 1 \ldots n_c & (a) \\ \tau \cdot \mathbf{1_m} \leq \boldsymbol{c} & (b) \\ \tau \in \mathbb{R}_0^+, \quad \boldsymbol{c} \in \mathbb{R}^m & (c) \end{cases}$$

where $\mathbf{1_m}$ is an m-dimensional column vector of ones.

Proof: Constraint (a) ensures that all vertices of the hypercube (namely all points of coordinate $\boldsymbol{c} \pm \tau \boldsymbol{e}$, for any canonical basis vector $\boldsymbol{e}$) satisfy the constraints. In particular, since $\boldsymbol{c} \in \mathcal{M}(\boldsymbol{W}, \boldsymbol{k})$, then $k_i - \boldsymbol{w}_i \ \boldsymbol{c} \geq 0$ and it is sufficient to check that $\boldsymbol{w}_i \ \boldsymbol{c} + \sum_{p \in P} |w_i(p)| \tau \leq k_i$, while constraints $\boldsymbol{w}_i \ \boldsymbol{c} - \sum_{p \in P} |w_i(p)| \tau \leq k_i$ are trivially verified.

Now, being $\mathcal{M}(\boldsymbol{W}, \boldsymbol{k})$ a convex set, this obviously implies that $\forall \boldsymbol{m}$ within the hypercube it holds $\boldsymbol{W}\boldsymbol{m} \leq \boldsymbol{k}$.

Constraint (b) ensures that all points $\boldsymbol{m}$ within the hypercube have nonnegative components.

The value of the performance index depends on the fact that we want to determine the maximal integer hypercube in $\mathcal{M}(\boldsymbol{W}, \boldsymbol{k})$.

Finally, let us observe that the assumption that $\mathcal{M}(\boldsymbol{W}, \boldsymbol{k})$ is bounded in at least one direction p_i, $i = 1, \ldots, m$, guarantees that the solution is not at the infinity. □

An important remark needs to be done. The above result holds under the assumption that $\mathcal{M}(\boldsymbol{W}, \boldsymbol{k})$ is bounded in at least one direction p_i, $i = 1, \ldots, m$. This is a main requirement to ensure that the solution is not at the infinity. Note however, that this is not a restrictive assumption in real applications, because in practice we always have physical limitations in the content of places. Thus, if $\mathcal{M}(\boldsymbol{W},\boldsymbol{k})$ is unbounded in any direction of p_i, $i = 1,\ldots,m$, we can always rewrite the set of legal markings by adding a constraint that limits the flow content in at least one direction.

VI. Maximal integer inner box computation

In this section we show how to solve the above step 3, namely how to determine the maximal integer inner box in $\mathcal{M}(\tilde{\boldsymbol{W}},\tilde{\boldsymbol{k}})$, where $\mathcal{M}(\tilde{\boldsymbol{W}},\tilde{\boldsymbol{k}})$ is a convex set that includes the origin as an interior point. Different criteria can be used.

A. A simple solution

The most immediate criterion is briefly summarized in the following algorithm that looks at all places in an arbitrary order, and assigns them the *largest* upper bound and the *smallest* lower bound that guarantee the satisfaction of all the constraints.

Algorithm 6: [Maximal integer inner box, a simple solution]

1. Let $\boldsymbol{k}^0 = \tilde{\boldsymbol{k}}$, $U_0 = \{1,\ldots,m\}$.

2. For $s = 1$ to m do

2.1. let $\bar{\jmath}_s$ be an index arbitrarily chosen in U_{s-1}

2.2. let $\bar{\imath} = \text{argmin}_{i\in\{1,\ldots,n_c\}} \left\lfloor \dfrac{k_i^{s-1}}{|\tilde{w}_i(p_{\bar{\jmath}_s})|} \right\rfloor$

2.3. if $\dfrac{k_{\bar{\imath}}^{s-1}}{\tilde{w}_{\bar{\imath}}(p_{\bar{\jmath}_s})} < 0$, then

let $l(p_{\bar{\jmath}_s}) = \left\lfloor \dfrac{k_{\bar{\imath}}^{s-1}}{\tilde{w}_{\bar{\imath}}(p_{\bar{\jmath}_s})} \right\rfloor$,

let $u(p_{\bar{\jmath}_s}) = \min\limits_{\substack{i \in \{1,\ldots,n_c\} \\ : \frac{k_i^{s-1}}{\tilde{w}_i(p_{\bar{\jmath}_s})} \geq 0}} \left\lfloor \dfrac{k_i^{s-1}}{\tilde{w}_i(p_{\bar{\jmath}_s})} \right\rfloor$

else

let $u(p_{\bar{\jmath}_s}) = \left\lfloor \dfrac{k_{\bar{\imath}}^{s-1}}{\tilde{w}_{\bar{\imath}}(p_{\bar{\jmath}_s})} \right\rfloor$, let $l(p_{\bar{\jmath}_s}) = \min\limits_{\substack{i \in \{1,\ldots,n_c\} \\ : \frac{k_i^{s-1}}{\tilde{w}_i(p_{\bar{\jmath}_s})} < 0}} \left\lfloor \frac{k_i^{s-1}}{\tilde{w}_i(p_{\bar{\jmath}_s})} \right\rfloor$

endif

2.4. for $i = 1$ to n_c do

if $\frac{k_i^{s-1}}{\tilde{w}_i(p_{\bar{\jmath}_s})} < 0$, do

let $k_i^s = k_i^{s-1} - l(p_{\bar{\jmath}_s}) w_i(p_{\bar{\jmath}_s})$

else

let $k_i^s = k_i^{s-1} - u(p_{\bar{\jmath}_s}) w_i(p_{\bar{\jmath}_s})$

endif

2.5. let $U_s = U_{s-1} \setminus \{\bar{\jmath}_s\}$. ■

In simple words, given a place $p_{\bar{\jmath}_s}$ arbitrarily selected at step 2.1 we look for the most restrictive constraint in the direction of $p_{\bar{\jmath}_s}$. If the most restrictive constraint corresponds to a negative intersection with the coordinate axes, then we first assign the lower bound to $p_{\bar{\jmath}_s}$; the upper bound should be computed by looking only at the constraints that provide a positive intersection with the axes in the direction of $p_{\bar{\jmath}_s}$. A dual reasoning repeates if the most restrictive constraint corresponds to a positive intersection with the coordinate axes.

Finally, we have to update the right hand side term of the constraints. We do this at step 2.4 by simply looking at the intersection of each constraint with the axes in the direction of $p_{\bar{\jmath}_s}$, using either the lower or the upper bound, depending on the sign of the ratio $k_i^{s-1}/\tilde{w}_i(p_{\bar{\jmath}_s})$.

Obviously, in this way we are not ensuring fairness among places and major chance to saturate the constraints is given to places that are firstly considered.

Example 7: Let us consider the following set of GMECs:

$$\mathcal{M}(\boldsymbol{W}, \boldsymbol{k}) = \{ \quad \boldsymbol{m} \in \mathbb{N}^2 \quad | \, m(p_1) + 2m(p_2) \leq 8, \\ -2m(p_1) + m(p_2) \leq 0\}.$$

From (3) it results that the maximal hypercube with center in $\mathcal{M}(\boldsymbol{W}, \boldsymbol{k})$ has edge $\tau = 1$ and center in $\boldsymbol{c} = (2, 1)$.

By mapping $\boldsymbol{c}$ into the origin with the coordinate translation $\boldsymbol{m}' = \boldsymbol{m} - \boldsymbol{c}$, the set of legal markings $\mathcal{M}(\boldsymbol{W}, \boldsymbol{k})$ is transformed into the equivalent one

$$\mathcal{M}(\tilde{\boldsymbol{W}}', \tilde{\boldsymbol{k}}') = \{ \quad \boldsymbol{m}' \in \mathbb{N}^2 \;\; | \, m'(p_1) + 2m'(p_2) \leq 4, \quad -2m'(p_1) + m'(p_2) \leq 3\}.$$

Finally, we need to impose an additional constraint, namely $-\boldsymbol{m}' \leq \boldsymbol{c}$,

$$\begin{cases} -m'(p_1) \leq 2 \\ -m'(p_2) \leq 1. \end{cases}$$

Assume that Algorithm 6 is used to design the decentralized monitors. If $\bar{\jmath}_1 = 1$ and $\bar{\jmath}_2 = 2$, i.e., we first assign the upper bound to p_1, then we get:

$$\begin{cases} -1 \leq m'(p_1) \leq 4 \\ -1 \leq m'(p_2) \leq 0 \end{cases} \Leftrightarrow \begin{cases} 1 \leq m(p_1) \leq 6 \\ 0 \leq m(p_2) \leq 1 \end{cases}$$

On the contrary, if $\bar{\jmath}_1 = 2$ and $\bar{\jmath}_2 = 1$, we obtain:

$$\begin{cases} 0 \leq m'(p_1) \leq 0 \\ -1 \leq m'(p_2) \leq 2 \end{cases} \Leftrightarrow \begin{cases} 2 \leq m(p_1) \leq 2 \\ 0 \leq m(p_2) \leq 3 \end{cases}$$

B. A solution to guarantee fairness among places

We now look for different criteria that ensure fairness among places. In particular, we provide a first algorithm to compute an integer inner box $\mathcal{B}(\boldsymbol{l}^*, \boldsymbol{u}^*) \subseteq \mathcal{M}(\tilde{\boldsymbol{W}}, \tilde{\boldsymbol{k}})$, that can be summarized in the following items. Then, we show under which assumptions such an algorithm guarantees that the resulting integer inner box is maximal. In the case that the maximality is not guaranteed we show how to modify it in order to do so.

- The algorithm is based on $2m$ iterative steps. At each step s we define a GMEC $(\boldsymbol{W}^s, \boldsymbol{k}^s)$, choosing at the initial step $(\boldsymbol{W}^0, \boldsymbol{k}^0) = (\tilde{\boldsymbol{W}}, \tilde{\boldsymbol{k}})$. We denote $\mathcal{M}_s = \mathcal{M}(\boldsymbol{W}^s, \boldsymbol{k}^s)$.
- At step s we compute the maximal integer hypercube in $\mathcal{M}_{s-1}$ using Proposition 4, and denote τ_s the corresponding edge.
- At each step we assign either the lower or the upper bound to one place belonging to the support of the current GMEC, whose magnitude coincide with the edge of the current hypercube. Thus, if $p_{\bar{\jmath}_s}$ is the place we have selected at step s, it results $u^*(p_{\bar{\jmath}_s}) = \tau_s$ or $l^*(p_{\bar{\jmath}_s}) = -\tau_s$, depending on the sign of $\tilde{w}_i(p_{\bar{\jmath}_s})$.
- The choice of the place to consider is essential to make sure that, at least under an appropriate condition that is discussed in the following, a maximal inner box is obtained.

Assume we are considering a constraint $\tilde{\boldsymbol{w}}_i \cdot \boldsymbol{m} \leq \tilde{k}_i$, and an integer hypercube with edge τ satisfying it. We define the *slack* of constraint i as $s_i = \tilde{k}_i - \tau|\tilde{\boldsymbol{w}}_i| \cdot \mathbf{1}$ where $\mathbf{1}$ is an m–dimensional column vector of ones, and $|\tilde{\boldsymbol{w}}_i|$ is an m–dimensional row vector whose generic j–th component, $j = 1, \ldots, m$, is equal to $|\tilde{w}_i(p_j)|$.

The *relative slack* of constraint i with respect to place p_j is defined as

$$s_{i,j} = \frac{\tilde{k}_i - \tau|\tilde{\boldsymbol{w}}_i| \cdot \mathbf{1}}{|\tilde{w}_i(p_j)|}.$$

We choose to assign either the lower or the upper bound at each step to the place that corresponds to the smallest relative slack, that we denote as $s_{\bar{\imath}_s,\bar{\jmath}_s}$. We discuss in Proposition 11 under which condition this choice leads to a maximal inner box.

- We define a new set of GMECs whose supports may not include those places to which a bound has already been assigned in the previous steps. As an example, if we assign an upper bound to $p_{\bar{\jmath}_s}$ (in such a case $pos = 1$) and $\tilde{w}_i(p_{\bar{\jmath}_s}) > 0$, then we eliminate $p_{\bar{\jmath}_s}$ from the support of the i-th constraint, otherwise we keep it. If we eliminate it, then the weights associated to the remaining places do not change, while $\boldsymbol{k}^s$ is updated to $k_i^s = k_i^{s-1} - \tau_s \tilde{w}_i(p_{\bar{\jmath}_s}) \leq k_i^{s-1}$. Now, if we denote as $U_s \cup L_s$ the set of indexes of places in the support of $\mathcal{M}_{s-1}$ to which an upper bound or a lower bound has not been assigned at iteration s, two different cases may occur:
  - if $\tilde{w}_{\bar{\imath}_s}(p_{\bar{\jmath}_s}) > 0$ the two constraints
$$\begin{cases} \displaystyle\sum_{j \in U_s \cup L_s} \tilde{w}_i^s(p_j) m(p_j) \leq k_i^s - \tau_s \tilde{w}_i^s(p_{\bar{\jmath}_s}) \\ m(p_{\bar{\jmath}_s}) \leq \tau_s \end{cases} \tag{4}$$
guarantee that
$$\sum_{j \in U_{s-1} \cup L_{s-1}} \tilde{w}_i^s(p_j) m(p_j) \leq k_i^s; \tag{5}$$
  - if $\tilde{w}_{\bar{\imath}_s}(p_{\bar{\jmath}_s}) < 0$ the two constraints
$$\begin{cases} \displaystyle\sum_{j \in U_s \cup L_s} \tilde{w}_i^s(p_j) m(p_j) \leq k_i^s + \tau_s \tilde{w}_i^s(p_{\bar{\jmath}_s}) \\ m(p_{\bar{\jmath}_s}) \geq -\tau_s \end{cases} \tag{6}$$
guarantee that $\displaystyle\sum_{j \in U_{s-1} \cup L_{s-1}} \tilde{w}_i^s(p_j) m(p_j) \leq k_i^s$.

  Namely, (4) or (6) guarantee the satisfaction of the GMEC at the previous step.

Formally, the algorithm can be written as follows.

Algorithm 8: [Inner box computation]

1. Let $\tau_0 = 0$, $\boldsymbol{W}^0 = \tilde{\boldsymbol{W}}$, $\boldsymbol{k}^0 = \tilde{\boldsymbol{k}}$,

$$L_0 = \{1, \dots, m\},\ U_0 = \{1, \dots, m\}$$

2. For $s = 1$ to $2m$ do

2.1. let $\mathcal{M}_{s-1} = \mathcal{M}(\boldsymbol{W}^{s-1}, \boldsymbol{k}^{s-1})$

2.2. let $\tau_s = \tau(\mathcal{M}_{s-1})$ (see Proposition 4)

2.3. let $\bar{\imath}_s, \bar{\jmath}_s$ be a couple of indexes arbitrarily chosen in

$$J_s = \left\{ (\bar{\imath}, \bar{\jmath}) \in \mathbb{N} \;\middle|\; s_{\bar{\imath},\bar{\jmath}}^{s-1} = \min_{\substack{j \in L_{s-1} \cup U_{s-1} \\ i \in \{1, \dots, n_c\}}} s_{i,j}^{s-1} \right\}$$

where $s_{i,j}^{s-1} = \dfrac{k_i^{s-1} - \tau |\boldsymbol{w}_i^{s-1}| \cdot \boldsymbol{1}}{|\tilde{w}_i(p_j)|}$

2.4. if $\tilde{w}_{\bar{\imath}_s}(p_{\bar{\jmath}_s}) < 0$, then

let $l_s(p_{\bar{\jmath}_s}) = -\tau_s$

let $L_s = L_{s-1} \setminus \{\bar{\jmath}_s\}$, $U_s = U_{s-1}$

let $neg = 1$, $pos = 0$

else

let $u_s(p_{\bar{\jmath}_s}) = \tau_s$

let $U_s = U_{s-1} \setminus \{\bar{\jmath}_s\}$, $L_s = L_{s-1}$

let $neg = 0$, $pos = 1$

endif

2.5. for $i = 1, \dots, n_c$ do

if $\tilde{w}_i(p_{\bar{\jmath}_s}) < 0$ and $neg = 1$, then

let $k_i^s = k_i^{s-1} + \tau_s \tilde{w}_i(p_{\bar{\jmath}_s})$

let $w_i^s = \begin{cases} 0 & \text{if } j = \bar{\jmath}_s \\ w_i^{s-1}(p_j) & \text{otherwise} \end{cases}$

elseif $\tilde{w}_i(p_{\bar{\jmath}_s}) > 0$ and $pos = 1$, then

let $k_i^s = k_i^{s-1} - \tau_s \tilde{w}_i(p_{\bar{\jmath}_s})$

let $w_i^s = \begin{cases} 0 & \text{if } j = \bar{\jmath}_s \\ w_i^{s-1}(p_j) & \text{otherwise} \end{cases}$

else

let $k_i^s = k_i^{s-1}$

let $w_i^s(p) = w_i^{s-1}(p), \forall p \in P$

endif

3. let $\boldsymbol{l}^* = \boldsymbol{l}_{2m}$, $\boldsymbol{u}^* = \boldsymbol{u}_{2m}$. ■

We now formally prove a rather intuitive result that will be used in the following.

Proposition 9: At each step of the previous algorithm it results $\tau_s \geq \tau_{s-1}$.

Proof: By definition, at the iteration $s+1$, it holds

$$\begin{aligned}
\tau_{s+1} &= \min_{i=1,\dots,n_c} \left\lfloor \frac{k_i^s}{\sum_{j \in U_s \cup L_s} |w_i^s(p_j)|} \right\rfloor \\
&= \min_{i=1,\dots,n_c} \left\lfloor \frac{k_i^{s-1} - \tau_s |w_i^{s-1}(p_{\bar{\jmath}_s})|}{\sum_{j \in U_s \cup L_s} |w_i^s(p_j)| + |w_i^{s-1}(p_{\bar{\jmath}_s})| - |w_i^{s-1}(p_{\bar{\jmath}_s})|} \right\rfloor \\
&\geq \min_{i=1,\dots,n_c} \left\lfloor \frac{\tau_s (\sum_{j \in U_{s-1} \cup L_{s-1}} |w_i^{s-1}(p_j)| - |w_i^{s-1}(p_{\bar{\jmath}_s}))|}{\sum_{j \in U_{s-1} \cup L_{s-1}} |w_i^{s-1}(p_j)| - |w_i^{s-1}(p_{\bar{\jmath}_s})|} \right\rfloor \\
&= \tau_s
\end{aligned}$$

where the inequality follows from the observation that by definition of τ_s, $\sum_{j \in U_{s-1} \cup L_{s-1}} |w_i^{s-1}(p_j)| \tau_s \leq k_i^{s-1}$. □

We can now prove the following results.

Proposition 10: Let $(\tilde{\boldsymbol{W}}, \tilde{\boldsymbol{k}})$ be a centralized constraint, $\boldsymbol{l}^*$ and $\boldsymbol{u}^*$ be the lower and upper bound vector determined by Algorithm 8. Then $\mathcal{B}(\boldsymbol{l}^*, \boldsymbol{u}^*) \subseteq \mathcal{M}(\tilde{\boldsymbol{W}}, \tilde{\boldsymbol{k}})$.

Proof: We first observe that it holds

$$\begin{aligned}
0 \leq \bar{k}_i^{2m} &= \tilde{k}_i - \textstyle\sum_{s=1}^{2m} |\tilde{w}_i(p_{\bar{\jmath}_s})| \tau_s \\
&= \tilde{k}_i - \textstyle\sum_{s=1}^{2m} \tilde{w}_i(p_{\bar{\jmath}_s}) x_i^*(p_{\bar{\jmath}_s}) \\
&= \tilde{k}_i - \tilde{\boldsymbol{w}}_i \cdot \boldsymbol{x}_i^*
\end{aligned}$$

Since for all $\boldsymbol{m}' \in \mathcal{B}(\boldsymbol{l}^*, \boldsymbol{u}^*)$ it is possible to write $\tilde{\boldsymbol{w}}_i \cdot \boldsymbol{m}' \leq \tilde{\boldsymbol{w}}_i \cdot \boldsymbol{x}_i^* = \tilde{k}_i - \bar{k}_i^{2m} \leq \tilde{k}_i$, it results $\boldsymbol{m}' \in \mathcal{M}(\tilde{\boldsymbol{W}}, \tilde{\boldsymbol{k}})$. □

In Proposition 9 we have shown that the sequence of edges τ_i of the maximal integer hypercubes determined by Algorithm 8 is nondecreasing. Next proposition shows that if this sequence is *strictly* increasing (with the possible exception of the tail of the sequence that may remain constant) a maximal inner box is obtained.

Proposition 11: Let $(\tilde{\boldsymbol{W}}, \tilde{\boldsymbol{k}})$ be a centralized constraint and $\boldsymbol{l}^*$, $\boldsymbol{u}^*$ be the final lower and upper bound vectors computed by Algorithm 8. If there exists an index $\mu \leq 2m$ such that the sequence of τ's computed by Algorithm 8 satisfies the condition

$$\tau_1 < \tau_2 < \cdots < \tau_\mu = \tau_{\mu+1} = \cdots = \tau_{2m} \tag{7}$$

then $\mathcal{B}(\boldsymbol{l}^*, \boldsymbol{u}^*)$ is a *maximal inner box* included in $\mathcal{M}(\tilde{\boldsymbol{W}}, \tilde{\boldsymbol{k}})$.

Proof: Proposition 10 has already shown that the box is included in $\mathcal{M}(\tilde{\boldsymbol{W}}, \tilde{\boldsymbol{k}})$. We will prove by contradiction that it is also maximal if condition (7) holds.

Suppose that there exists an inner box $\mathcal{B}(\tilde{\boldsymbol{l}}, \tilde{\boldsymbol{u}})$ such that $\mathcal{B}(\boldsymbol{l}^*, \boldsymbol{u}^*) \subsetneq \mathcal{B}(\tilde{\boldsymbol{l}}, \tilde{\boldsymbol{u}}) \subseteq \mathcal{M}(\boldsymbol{W}, \boldsymbol{k})$, i.e., such that $\tilde{\boldsymbol{l}} \lneq \boldsymbol{l}^*$ or $\tilde{\boldsymbol{u}} \gneq \boldsymbol{u}^*$. Then, there must exist an index h such that $\tilde{l}(p_h) < l^*(p_h)$ or $\tilde{u}(p_h) > u^*(p_h)$. Assume, without loss of generality that $\tilde{u}(p_h) = u^*(p_h) + 1$ and $\tilde{u}(p_j) = u^*(p_j), \forall j \neq h$ and $\tilde{l}(p_j) = l^*(p_j)$, $\forall j$. Suppose that $u^*(p_h)$ has been fixed at the l-th step of Algorithm 8, i.e., $u^*(p_h) = \tau_l$.

Furthermore, since $\mathcal{B}(\tilde{\boldsymbol{l}}, \tilde{\boldsymbol{u}}) \subseteq \mathcal{M}(\tilde{\boldsymbol{W}}, \tilde{\boldsymbol{k}})$ and by definition $\tilde{\boldsymbol{x}} \in \mathcal{B}(\tilde{\boldsymbol{l}}, \tilde{\boldsymbol{u}})$, it holds for all $i = 1, \ldots, n_c$,

$$\begin{aligned} &\sum_{j \in U_{l-1} \cup L_{l-1}} \tilde{w}_i(p_j)\tilde{x}_i(p_j) \\ \leq k_i \quad &- \sum_{j \in \{U_0 \setminus U_{l-1}\} \cup \{L_0 \setminus L_{l-1}\}} \tilde{w}_i(p_j)\tilde{x}_i(p_j) \\ = k_i \quad &- \sum_{j \in \{U_0 \setminus U_{l-1}\} \cup \{L_0 \setminus L_{l-1}\}} \tilde{w}_i(p_j)x_i^*(p_j) = k_i^{l-1} \end{aligned} \tag{8}$$

where $U_0 \setminus U_{l-1}$ and $L_0 \setminus L_{l-1}$ contains the indexes of the places to which an upper or an upper bound respectively has been assigned in the first $l-1$ iterations of the algorithm.

We consider two cases.

Case I: $l < \mu$. Condition (7) implies that $\forall j \in U_l$ it holds $\tilde{u}(p_j) = u^*(p_j) \geq u^*(p_h)+1 = \tau_l+1$. Since $\tilde{u}(p_h) = \tau_l + 1$, we can also conclude that for all $j \in L_{l-1} = L_l$ it holds $|\tilde{l}(p_j)| \geq \tau_l + 1$, i.e, from (8) we have that for all $i = 1, \ldots, n_c$, $\sum_{j \in U_{l-1} \cup L_{l-1}} |w_i^{l-1}(p_j)|(\tau_l + 1) \leq k_i^{l-1}$. This means that at step l an hypercube with edge $\tau_l + 1$ should have been chosen by the algorithm. Clearly this leads to a contradiction.

Case II: $l \geq \mu$. First we note that in this case for all $j \in U_{l-1}$ it holds $u^*(p_j) = \tau_l = \tau_{2m} = \tau$ and for all $j \in L_{l-1}$ it holds $|l^*(p_j)| = \tau_l = \tau_{2m} = \tau$, and for all $i = 1, \ldots, n_c$, we can rewrite (8) as

$$|w_i^{l-1}(p_h)| + \sum_{j \in U_{l-1} \cup L_{l-1}} |w_i^{l-1}(p_j)|\tau \leq k_i^{l-1}. \tag{9}$$

Then, using the fact that the algorithm eliminates at each step the place with minimal relative slack, we prove that it also holds

$$|w_i^{l-1}(p_{h'})| + \sum_{j \in U_{l-1} \cup L_{l-1}} |w_i^{l-1}(p_j)|\tau \leq k_i^{l-1}, \tag{10}$$

where $p_{h'}$ is the place removed at step $2m$ of the algorithm. In fact, it is not difficult to see that (9) implies that the relative slacks of places p_h satisfy, for all $i = 1, \ldots, n_c$, $s_{i,h}^{l-1} \geq 1$ and since it also holds $s_{i,h'}^{l-1} \geq \min_{i=1,\ldots,n_c} s_{i,h}^{l-1} \geq 1$ we obtain (10), that in turn can be rewritten, for all $i = 1, \ldots, n_c$,

$$\begin{aligned} &|w_i^{l-1}(p_{h'})|(\tau+1) \leq \\ &\leq k_i^{l-1} - \sum_{j \in U_{l-1} \cup L_{l-1} \setminus \{h'\}} |w_i^{l-1}(p_j)|\tau = k_i^{2m-1}. \end{aligned}$$

Hence we observe that at the last step the algorithm should have assigned to place $p_{h'}$ a bound whose absolute value is equal to $\tau + 1$, thus reaching a contradiction. □

Example 12: Let us consider again the set of GMECs $\mathcal{M}(\tilde{\boldsymbol{W}}', \tilde{\boldsymbol{k}}')$ of Example 7.

Applying Algorithm 8 the resulting inner box is

$$\begin{cases} -1 \leq m'(p_1) \leq 2 \\ -1 \leq m'(p_2) \leq 1 \end{cases} \Leftrightarrow \begin{cases} 1 \leq m(p_1) \leq 4 \\ 0 \leq m(p_2) \leq 2 \end{cases}$$

i.e. $\mathcal{B}(\boldsymbol{l}^*, \boldsymbol{u}^*)$ with $\boldsymbol{l}^* = [1\ 1]^T$ and $\boldsymbol{u}^* = [4\ 2]^T$.

The sequence of maximal edges computed by the algorithm is $\tau_1 = 1, \tau_2 = 1, \tau_3 = 2$ that does not satisfy condition (7). However, from Proposition 3 we can conclude that the inner box $\mathcal{B}(\boldsymbol{l}^*, \boldsymbol{u}^*)$ is maximal. ■

Algorithm 8 may be easily modified in order to guarantee that the resulting inner box is maximal.

Algorithm 13: [Maximal inner box computation]

1. Run Algorithm 8. Assume that the sequence of τ's is $\tau_1 \leq \tau_2 \cdots \leq \tau_\mu = \cdots = \tau_{2m}$.

2. Let $\bar{\boldsymbol{l}}^0 = \boldsymbol{l}_{2m}$, $\bar{\boldsymbol{u}}^0 = \boldsymbol{u}_{2m}$

3. For $s = 1$ to $\mu - 1$ do

3.1. for $j = 1, \ldots, 2m$, $j \neq \bar{\jmath}_s$
let $\bar{u}^s(p_j) = \bar{u}^{s-1}(p_j)$
let $\bar{l}^s(p_j) = \bar{l}^{s-1}(p_j)$

3.2. if $u_{2m}(p_{\bar{\jmath}_s}) = \tau_s$ let $pos = 1$ else let $neg = 1$

3.3. if pos let

$$\bar{u}^s(p_{\bar{\jmath}_s}) = \begin{cases} \bar{u}^{s-1}(p_{\bar{\jmath}_s}) & \text{if } \tau_s < \tau_{s+1} \\ \bar{u}^{s-1}(p_{\bar{\jmath}_s}) + \\ \quad + \min_{i \in \{1,\ldots,n_c\}} \left\lfloor \dfrac{k_i - \tilde{\boldsymbol{w}}_i \cdot \bar{\boldsymbol{x}}_i^{s-1}}{|\tilde{w}_i(p_{\bar{\jmath}_s})|} \right\rfloor \\ & \text{if } \tau_s = \tau_{s+1} \end{cases}$$

else

$$\bar{l}^s(p_{\bar{\jmath}_s}) = \begin{cases} \bar{l}^{s-1}(p_{\bar{\jmath}_s}) & \text{if } \tau_s < \tau_{s+1} \\ \bar{l}^{s-1}(p_{\bar{\jmath}_s}) + \\ \quad - \min\limits_{i \in \{1,\dots,n_c\}} \left\lfloor \dfrac{k_i - \tilde{\boldsymbol{w}}_i \cdot \bar{\boldsymbol{x}}_i^{s-1}}{|\tilde{w}_i(p_{\bar{\jmath}_s})|} \right\rfloor \\ & \text{if } \tau_s = \tau_{s+1} \end{cases}$$

4. let $\boldsymbol{l}^* = \bar{\boldsymbol{l}}^{\mu-1}$, $\boldsymbol{u}^* = \bar{\boldsymbol{u}}^{\mu-1}$. ■

The main idea behind the new steps of the algorithm is the following. The solution computed using Algorithm 8 provides a maximal inner box when the sequence of τ's is strictly increasing, apart from the tail of the sequence that may keep constant. On the contrary, no guarantee is given if two or more τ's that are not in the tail are equal. Therefore, we look for all variables to which it corresponds the same upper bound that is different from τ_m, and we verify if their upper or lower bounds may be respectively further increased or decreased. If so, we increase or decrease them as much as possible in accordance with the given constraints, and go further with our exploration.

Note that, by Proposition 9, at step 1 of Algorithm 13, only places whose bound has been assigned in consecutive steps may have equal upper bounds.

Proposition 14: Let $(\tilde{\boldsymbol{W}}, \tilde{\boldsymbol{k}})$ be a centralized constraint. Let $\boldsymbol{l}^*$, $\boldsymbol{u}^*$ be the final lower and upper bound vectors determined by Algorithm 13. Then $\mathcal{B}(\boldsymbol{l}^*, \boldsymbol{u}^*)$ is a *maximal inner box* included in $\mathcal{M}(\tilde{\boldsymbol{W}}, \tilde{\boldsymbol{k}})$.

Proof: If $\tau_s < \tau_{s+1}$ for all $s = 1, \dots, \mu - 1$, and $\tau_s = \tau_{s+1}$ for all $s = \mu, \dots, 2m - 1$, the solution provided by Algorithm 13 coincides with that of Algorithm 8 and the result follows from Proposition 11.

Now, let r be the smallest value of $s \in \{1, \dots, \mu - 1\}$ such that $\tau_s = \tau_{s+1}$.

Suppose without loss of generality that at step s of Algorithm 8 it results $u_{2m}(p_{\bar{\jmath}_s}) \neq \tau_s$, i.e. a lower bound has been assigned to $p_{\bar{\jmath}_s}$.

Then, at step 3.3 we impose:

$$\begin{aligned} \bar{l}^r(p_{\bar{\jmath}_r}) &= \bar{l}^{r-1}(p_{\bar{\jmath}_r}) - \min_{i \in \{1,\dots,n_c\}} \left\lfloor \frac{k_i - \tilde{\boldsymbol{w}}_i \cdot \bar{\boldsymbol{x}}_i^{s-1}}{|\tilde{w}_i(p_{\bar{\jmath}_r})|} \right\rfloor \\ &= \bar{l}^{r-1}(p_{\bar{\jmath}_r}) - \min_{i \in \{1,\dots,n_c\}} \left\lfloor \frac{k_i - \tilde{\boldsymbol{w}}_i \cdot \bar{\boldsymbol{x}}_i^{0}}{|\tilde{w}_i(p_{\bar{\jmath}_r})|} \right\rfloor \end{aligned}$$

while $\bar{\boldsymbol{l}}^r(p_j) = \bar{\boldsymbol{l}}^0(p_j)$ for all $j \neq \bar{\jmath}_r$.

Let $\bar{k}_i^0 = k_i - \tilde{\boldsymbol{w}}_i \cdot \bar{\boldsymbol{x}}_i^0$ and $\bar{k}_i^r = k_i - \tilde{\boldsymbol{w}}_i \cdot \bar{\boldsymbol{x}}_i^r$, thus

$$\begin{aligned}\bar{k}_i^r &= k_i - \tilde{\boldsymbol{w}}_i \cdot \bar{\boldsymbol{x}}_i^0 - |\tilde{w}_i(p_{\bar{\jmath}_r})| \min_{i \in \{1,\ldots,n_c\}} \left\lfloor \frac{k_i - \tilde{\boldsymbol{w}}_i \cdot \bar{\boldsymbol{x}}^0}{|\tilde{w}_i(p_{\bar{\jmath}_r})|} \right\rfloor \\ &= \bar{k}_i^0 - |w_i(p_{\bar{\jmath}_r})| \min_{i \in \{1,\ldots,n_c\}} \left\lfloor \frac{\bar{k}_i^0}{|w_i(p_{\bar{\jmath}_r})|} \right\rfloor\end{aligned}$$

and

$$\frac{\bar{k}_i^r}{|\tilde{w}_i(p_{\bar{\jmath}_r})|} = \frac{\bar{k}_i^0}{|\tilde{w}_i(p_{\bar{\jmath}_r})|} - \min_{i \in \{1,\ldots,n_c\}} \left\lfloor \frac{\bar{k}_i^0}{|\tilde{w}_i(p_{\bar{\jmath}_r})|} \right\rfloor < 1.$$

Therefore, by Proposition 3, $\mathcal{B}(\bar{\boldsymbol{l}}_r, \bar{\boldsymbol{l}}_r)$ is a maximal inner box. Similarly, if we denote as q the smallest value of $s \in \{r+1, \ldots \mu - 1\}$ such that $\tau_s = \tau_{s+1}$, we can prove that

$$\frac{\bar{k}_i^q}{|\tilde{w}_i(p_{\bar{\jmath}_q})|} < 1,$$

thus, iteratively repeating the same reasoning until all places have been considered, we conclude that $\mathcal{B}(\bar{\boldsymbol{l}}^{\mu-1}, \bar{\boldsymbol{u}}^{\mu-1})$ is a maximal inner box. □

An important remark needs to be done.

Remark 15: For simplicity of notation, in Algorithm 13 we have assumed that the upper and lower bounds are increased, when possible, following the same order in which they have been assigned in step 2. Clearly, this is not the only admissible solution. Variables that share the same upper or lower bound may be examined in any order, and this in general provides different decentralized constraints. In any case Proposition 14 still applies, and all the resulting solutions are maximal inner boxes. ■

VII. Conclusions and Future Works

In this paper we have investigated the problem of determining a set of decentralized GMECs that are able to impose a specification on the net behavior given in terms of a global set of GMECs. The proposed solution is based on the assumption that all transitions are controllable and observable, and that the support of each decentralized GMEC is a singleton. Our future efforts will be devoted to generalize these results by removing such assumptions.

References

[1] G. Barret and S. Lafortune. Decentralized supervisory control with communicating controllers. *IEEE Transaction on Automatic Control*, 45(9):1620–1638, 2000.

[2] F. Basile, A. Giua, and C. Seatzu. Supervisory control of Petri nets with decentralized monitor places. 26^{th} *American Control Conference (ACC'07), New York, USA*, pages 4957–4962, July 2007.

[3] A. Bemporad, C. Filippi, and F.D. Torrisi. Inner and outer approximations of polytopes using boxes. *Computational Geometry*, 27:151–178, 2004.

[4] H. Chen and H. Baosheng. Distributed control of discrete event systems described by a class of controlled Petri nets. *IFAC International Symposium on Distributed Intelligence Systems*, 1991.

[5] A. Giua, F. DiCesare, and M. Silva. Generalized mutual exclusion constraints on nets with uncontrollable transitions. *1992 IEEE Int. Conf. on Systems, Man, and Cybernetics (Chigago, Illinois)*, pages 974–979, October 1992.

[6] X. Guan and L.E. Holloway. Control of distributed discrete event systems modeled as Petri nets. *Proc. 1997 American Control Conference, Albuquerque, New Mexico*, pages 2342–2347, June 1997.

[7] M.V. Iordache and P.J. Antsaklis. Decentralized control of Petri nets with constraint tansformation. *IEEE Transaction on Automatic Control*, 51(2):376–381, February 2006.

[8] F. Lin and W.M. Wonham. Decentralized control and coordination of discrete-event systems with partial observation. *IEEE Transaction on Automatic Control*, 35(12):1330–1337, 1990.

[9] T. Murata. Petri nets: Properties, analysis and applications. *Proceedings of IEEE*, 77(4):541–580, April 1989.

[10] K. Rudie. The current state of decentralized discrete-event control systems. *Proc. 2002 IEEE mediterranean Conference on Control and Automation, Lisbon, Portugal*, June 2002.

[11] K. Rudie and W.M. Wonham. Think globally, act locally: Decentralized supervisory control. *IEEE Transaction on Automatic Control*, 37(11):1692–1708, 1992.

[12] S. Takai, S. Kodama, and T. Ushio. Decentralized state feedback control of discrete event systems. *Systems and Control Letters*, 22(5):369–375, 1994.