Designing Reversibility-Enforcing Supervisors of Polynomial Complexity for Bounded Petri Nets through the Theory of Regions

Spyros A. Reveliotis[1] and Jin Young Choi[2]

[1] School of Industrial & Systems Engineering
Georgia Institute of Technology
spyros@isye.gatech.edu
[2] Digital Communications Infra Division
Samsung Networks Inc.
jin_young.choi@samsung.com

Abstract. This paper proposes an analytical method for the synthesis of reversibility-enforcing supervisors for bounded Petri nets. The proposed method builds upon recent developments from (i) the theory of regions, that enables the design of Petri nets with pre-specified behavioral requirements, and (ii) the theory concerning the imposition of generalized mutual exclusion constraints on the net behavior through monitor places. The derived methodology takes the form of a Mixed Integer Programming formulation, which is readily solvable through canned optimization software. The last part of the paper discusses extensions of the presented method so that it accommodates uncontrollable behavior and any potential complications arising from the large-scale nature of the underlying plant nets and their behavioral spaces. Finally, the relevance and the efficacy of the proposed approach is demonstrated through its application in the synthesis of liveness-enforcing supervisors for process-resource nets.

1 Introduction

Reversibility is a well-characterized and important property in many contemporary technological applications and it implies the ability of the studied system to retrieve its initial state from any state that this system can reach during its operation. Clearly, under this basic definition, reversibility applies to the entire range of systems that can be modelled through dynamical system theory; however, in this work we deal with the concept of reversibility in the more restricted class of systems that can be modelled by bounded Petri nets (PN) [5]. For systems modelled in this representational framework, we seek to develop a methodology that will support the design of controllers (i) enforcing the reversibility of the underlying system and (ii) possessing an "on-line" computational cost that is polynomial with respect to the size of this system. The pursued approach is a combination of (i) Petri net supervisory control based on the theory of monitor places [3, 4] and (ii) the design of Petri nets with a desired / pre-specfied topology for their reachability space through the theory of regions [1]. In this

sense, our work presents considerable similarity with the works of [2, 9], which also sought to develop monitor-based supervisors for some PN sub-classes modelling sequential resource allocation, while exploiting insights and results coming from the theory of regions. However, the main differentiator of our work from those past efforts is the emphasis that we place on the (polynomial) complexity of the derived solutions. More specifically, in the previous works, the authors sought to derive a set of monitor places that would encode the *maximally permissive* supervisor for the considered application context, where the latter was initially computed through state space-based approaches (typically, Ramadge & Wonham's supervisory control theory [7]). Two significant implications of that approach were that (i) the derived supervisor might employ a number of monitor places that was a super-polynomial function of the size of the underlying Petri net, and (ii) there might be cases that the approach would fail to return a supervisor since it might not be possible to enforce the target behavior through a set of monitor places. Contrary to those past works, in our approach we define *a priori* the maximum number of monitor places that we want to use in the derived solution, and we seek to develop a supervisor that will guarantee "best performance" under this size restriction. The applied performance criterion can be quite general; for the purposes of the subsequent discussion, we shall assume that it can be expressed by a sum of values / weights taken over the set of states that are admitted by the derived supervisor. By restricting the number of the considered monitor places to be a polynomial function of the underlying system size, our approach can guarantee the polynomial "on-line" complexity of the derived solution. Furthermore, as it will be established in the following, the overall design problem reduces to the solution of a mathematical programming (MP) formulation consisting of the aforementioned performance objective and a set of constraints expressing the requirement for reversibility of the controlled system. This formulation essentially constitutes an implicit search for a best supervisor over the entire set of supervisors that can enforce the system reversibility while observing the imposed size constraint, and therefore, it is richer in feasible solutions than the earlier developed approaches. In addition, the explicit parameterization of the proposed approach through the maximum number of the employed monitor places allows the systematic study of the trade-off between the computational complexity of the applied supervisor and the attained performance. Finally, in principle, this approach can still enable the computation of the maximally permissive supervisor – assuming that this supervisor is implementable by a set of monitor places – by setting the number of the provided monitor places to a sufficiently large value.

¿From a more historical standpoint, this research falls within a broader initiative of ours, seeking to derive polynomial-complexity, monitor-based, reversibility-enforcing supervisors for a class of PN's modelling sequential resource allocation. This class of PN's and the currently available results on its reversibility-enforcing supervision through monitor-based approaches, are presented in [8, 11].[3] Yet, one

[3] In fact, one of the main results for this PN sub-class is that the net liveness and reversibility are equivalent concepts; hence, in [8, 11], the aforementioned supervi-

of the open research questions raised in [8] is the analytical characterization of the entire set of supervisors that can enforce the reversibility of any given instance of the considered PN sub-class, while employing a pre-specified number of monitor places. This question is resolved in this paper through the constraint set of the aforementioned MP formulation.

In the light of the above introduction of the presented work, the rest of the paper is organized as follows: Section 2 first reviews the basic PN concepts and results that are necessary for the development of this work, and subsequently it summarizes the key elements of the theory of regions, according to the perspective adopted in [2], and of the monitor-based Petri net control theory, developed in [4]. Section 3 develops the supervisor design approach proposed in this work, while Section 4 demonstrates the modelling and analytical power of this approach through a detailed example. Section 5 discusses some enhancements and extensions of the basic methodology presented in Section 3, and, finally, Section 6 concludes the paper and highlights directions for future work.

2 Preliminaries

2.1 Petri net fundamentals

Petri net Definition [5] A *(marked) Petri net (PN)* is defined by a quadruple $\mathcal{N} = (P, T, W, M_0)$, where

- P is the set of *places*,
- T is the set of *transitions*,
- $W : (P \times T) \cup (T \times P) \rightarrow Z_0^+$ is the *flow relation*,[4] and
- $M_0 : P \rightarrow Z_0^+$ is the net *initial marking*, assigning to each place $p \in P$, $M_0(p)$ *tokens*.

Also, for the purposes of the subsequent analysis, the *size* of PN $\mathcal{N} = (P, T, W, M_0)$ is defined as $|\mathcal{N}| \equiv |P| + |T| + \sum_{p \in P} M_0(p)$.

The first three items in the above definition essentially constitute a *weighted bipartite digraph* representing the system *structure* that governs its underlying dynamics. The last item determines the system *initial state*. A conventional graphical representation of the net structure and its marking depicts nodes corresponding to places by empty circles, nodes corresponding to transitions by bars, and the tokens located at the various places by small filled circles. The flow relation W is depicted by directed edges that link every nodal pair for which the corresponding W-value is non-zero. These edges point from the first node of the corresponding pair to the second, and they are also labelled – or, *"weighed"* – by the corresponding W-value. By convention, absence of a label for any edge implies that the corresponding W-value is equal to unity.

sory control problem is characterized as *liveness* rather than reversibility-enforcing supervision.

[4] In this work, Z_0^+ denotes the set of nonnegative integers, and $\Re$ denotes the set of reals.

Some structure-related PN concepts For computational purposes, the net flow relation W is encoded by two $|P| \times |T|$ matrices, Θ^+ and Θ^-, with $\Theta^+(p,t) = W(t,p)$ and $\Theta^-(p,t) = W(p,t)$. The difference $\Theta^+ - \Theta^-$ is known as the net *flow matrix* and it is denoted by Θ. A PN is said to be *pure* if and only if (*iff*) $\forall p \in P,\ \forall t \in T,\ \Theta^-(p,t)\Theta^+(p,t) = 0$. Notice that for pure PN's, the net flow relation, W, is completely characterized by the net flow matrix, Θ.

Given a transition $t \in T$, the set of places p for which $(p,t) > 0$ (resp., $(t,p) > 0$) is known as the set of *input* (resp., *output*) places of t. Similarly, given a place $p \in P$, the set of transitions t for which $(t,p) > 0$ (resp., $(p,t) > 0$) is known as the set of *input* (resp., *output*) transitions of p. It is customary in the PN literature to denote the set of input (resp., output) transitions of a place p by ${}^\bullet p$ (resp., $p^\bullet$). Similarly, the set of input (resp., output) places of a transition t is denoted by ${}^\bullet t$ (resp., $t^\bullet$). This notation is also generalized to any set of places or transitions, X, e.g. ${}^\bullet X = \bigcup_{x \in X} {}^\bullet x$.

The ordered set $X =< x_1 \dots x_n > \in\ (P \cup T)^*$ is a *path iff* $x_{i+1} \in x_i^\bullet, i = 1, \dots, n-1$. Furthermore, a path X is characterized as a *circuit iff* $x_1 \equiv x_n$.

The particular class of PN's with a flow relation W mapping onto $\{0,1\}$ are characterized as *ordinary*. An ordinary PN with $|t^\bullet| = |{}^\bullet t| = 1,\ \forall t \in T$, is characterized as a *state machine*, while an ordinary PN with $|p^\bullet| = |{}^\bullet p| = 1$, $\forall p \in P$, is characterized as a *marked graph*.

Some dynamics-related PN concepts In the PN modelling framework, the system state is represented by the net *marking* M, i.e., a function from P to Z_0^+ that assigns a *token* content to the various net places. The net marking M is initialized to marking M_0, introduced in the PN definition provided at the beginning of this section, and it subsequently evolves through a set of rules summarized in the concept of *transition firing*. A concise characterization of this concept has as follows: Given a marking M, a transition t is *enabled iff* for every place $p \in {}^\bullet t,\ M(p) \geq W(p,t)$, or equivalently, $M \geq \Theta^-(\cdot, t)$, and this fact is denoted by $M[t\rangle$. $t \in T$ is said to be *disabled* by a place $p \in {}^\bullet t$ at M *iff* $M(p) < W(p,t)$, or, equivalently, $M(p) < \Theta^-(p,t)$. Given a marking M, a transition t can be *fired* only if it is enabled in M, and firing such an enabled transition t results in a new marking M', which is obtained from M by removing $W(p,t)$ tokens from each place $p \in {}^\bullet t$, and placing $W(t,p')$ tokens in each place $p' \in t^\bullet$. The marking evolution incurred by the firing of a transition t can be concisely expressed by the *state equation*:

$$M' = M + \Theta \cdot \mathbf{1}_t \tag{1}$$

where $\mathbf{1}_t$ denotes the unit vector of dimensionality $|T|$ and with the unit element located at the component corresponding to transition t.

Given a PN $\mathcal{N}$, a sequence of transitions, $\sigma = t_1 t_2 \dots t_n$, is *fireable* from some marking M *iff* $M[t_1\rangle M_1[t_2\rangle M_2 \dots M_{n-1}[t_n\rangle\ M_n$; we shall also denote this fact by $M \xrightarrow{\sigma} M_n$. The *length* of σ is defined by the number of transitions in it, and it will be denoted by $|\sigma|$. Also, the *Parikh vector* of σ is a $|T|$-dimensional vector, $\bar{\sigma}$, with each component $\bar{\sigma}(t),\ t \in T$, stating the number of appearances of transition t in σ.

The set of markings reachable from the initial marking M_0 through any *fireable* sequence of transitions is denoted by $R(\mathcal{N}, M_0)$ and it is referred to as the net *reachability space*. Equation 1 implies that a necessary condition for $M \in R(\mathcal{N}, M_0)$ is that the following system of equations is feasible in z:

$$M = M_0 + \Theta z \tag{2}$$

$$z \in (Z_0^+)^{|T|} \tag{3}$$

The *reachability graph*, $\mathcal{G}(\mathcal{N}, M_0)$, of $\mathcal{N}$, is a labelled directed graph with its node set being equal to $R(\mathcal{N}, M_0)$, and its edge set being defined by the nodal pairs $(M, M') \in R(\mathcal{N}, M_0) \times R(\mathcal{N}, M_0)$ for which there exists $t \in T$ such that $M[t\rangle M'$; the edges of $\mathcal{G}(\mathcal{N}, M_0)$ are labelled by the corresponding transitions.

A PN $\mathcal{N} = (P, T, W, M_0)$ is said to be *bounded iff* all markings $M \in R(\mathcal{N}, M_0)$ are bounded. $\mathcal{N}$ is said to be *structurally bounded iff* it is bounded for any initial marking M_0. $\mathcal{N}$ is said to be *reversible iff* $M_0 \in R(\mathcal{N}, M)$, for all $M \in R(\mathcal{N}, M_0)$, and any marking $M \in R(\mathcal{N}, M_0)$ such that $M_0 \in R(\mathcal{N}, M)$ is a *co-reachable* marking of $\mathcal{N}$. A transition $t \in T$ is said to be *live iff* for all $M \in R(\mathcal{N}, M_0)$, there exists $M' \in R(\mathcal{N}, M)$ such that $M'[t\rangle$; non-live transitions are said to be *dead* at those markings $M \in R(\mathcal{N}, M_0)$ for which there is no $M' \in R(\mathcal{N}, M)$ such that $M'[t\rangle$. PN $\mathcal{N}$ is *quasi-live iff* for all $t \in T$, there exists $M \in R(\mathcal{N}, M_0)$ such that $M[t\rangle$; it is *weakly live iff* for all $M \in R(\mathcal{N}, M_0)$, there exists $t \in T$ such that $M[t\rangle$; and it is *live iff* for all $t \in T$, t is live.

PN semiflows PN semiflows provide an analytical characterization of various concepts of *invariance* underlying the net dynamics. Generally, there are two types, p and t-semiflows, with a *p-semiflow* formally defined as a $|P|$-dimensional vector y satisfying $y^T\Theta = 0$ and $y \geq 0$, and a *t-semiflow* formally defined as a $|T|$-dimensional vector x satisfying $\Theta x = 0$ and $x \geq 0$. In the light of Equation 2, the invariance property expressed by a p-semiflow y is that $y^T M = y^T M_0$, for all $M \in R(\mathcal{N}, M_0)$. Similarly, Equation 2 implies that for any t-semiflow x, $M = M_0 + \Theta x = M_0$.

2.2 Petri net design through the Theory of Regions

In this section we overview an interpretation of the *theory of regions* provided in [2]. According to this interpretation, the problem addressed by the theory of regions can be stated as follows: Given a directed graph, $G = (N, E)$, with its edges labelled by elements from some set T, and containing a node $n_0 \in N$ such that there exists a path from n_0 to any other node $n \in N$, find a *pure* PN $\mathcal{N} = (P, T, W, M_0)$, such that its reachability graph $\mathcal{G}(\mathcal{N}, M_0)$ is expressed by G, when setting $M_0 \equiv n_0$. Since the net $\mathcal{N}$ is required to be pure, it can be fully defined by specifying the row $\Theta(p, \cdot)$ of the net flow matrix Θ and the initial marking $M_0(p)$, for each place $p \in P$. These parameters can be subsequently obtained through a system of equations derived from the structure of the target graph G and the logic underlying Equations 1–3.

In particular, Equation 1 implies that, for any undirected cycle, γ, in graph G:

$$\forall p \in P, \quad \Theta(p, \cdot) \cdot \bar{\gamma} = 0 \tag{4}$$

Equation 4 is known as the *"cycle"* equation of the theory of regions, and the parameter $\bar{\gamma}$ appearing in it is a vector of dimensionality $|T|$, and with component $\bar{\gamma}(t)$ denoting the difference between the number of times that t is encountered in γ labelling an edge pointing in the direction of the traversal of γ, and the number of times that t is encountered in γ labelling an edge pointing in the opposite direction.

Similarly, the reachability of a node $n \in N$ from node n_0 through some path $\xi(n)$, implies that

$$\forall p \in P, \quad M_0(p) + \Theta(p, \cdot) \cdot \bar{\xi}(n) \geq 0 \tag{5}$$

Equation 5 is known as the *"reachability condition"* associated with node n, and the parameter $\bar{\xi}(n)$ appearing in it is a vector of dimensionality $|T|$ and with component $\bar{\xi}(n;t)$ indicating the number of appearances of transition t in path $\xi(n)$. For nodes n reachable from n_0 through more than one paths, only one of the corresponding reachability conditions should be included in the considered system of equations, since the reachability conditions corresponding to the remaining paths can be derived from the included condition and the cycle equations discussed above.

On the other hand, for every node $n \in N$ and transition $t \in T$ such that there is no edge emanating from n that is labelled by t, there must exist a place $p \in P$ that disables the firing of transition t at the marking M corresponding to node n. This requirement is imposed by the following equation:

$$\exists p \in P, \quad M_0(p) + \Theta(p, \cdot) \cdot \bar{\xi}(n) + \Theta(p, t) \leq -1 \tag{6}$$

Equation 6 is known as the *"event separation condition"* associated with node-transition pair (n, t), and the parameter $\bar{\xi}(n)$ appearing in it is the same with that appearing in Equation 5. Also, the node-transition pairs, (n, t), such that there is no edge emanating from n that is labelled by t, are characterized as the *"event separation instances"*.

Finally, a last requirement is that the various nodes $n \in N$ of graph G correspond to different markings of the PN $\mathcal{N}$; i.e., for any given nodal pair (n, n'),

$$\exists p \in P, \quad \Theta(p, \cdot) \cdot \bar{\xi}(n) \neq \Theta(p, \cdot) \cdot \bar{\xi}(n') \tag{7}$$

Equation 7 in known as the *"state separation condition"*, and the parameter $\bar{\xi}(n)$ appearing in it is defined as in Equations 5 and 6.

In the light of the above characterizations, the theory of regions is epitomized by the following theorem:

Theorem 1. *[2] Consider a directed graph, $G = (N, E)$, with its edges labelled by elements from some set T, and containing a node $n_0 \in N$ such that there exists a path from n_0 to any other node $n \in N$. Then, there exists a pure PN $\mathcal{N} = (P, T, W, M_0)$ with graph G as its reachability graph and with node n_0 corresponding to its initial marking M_0, iff (i) for each place $p \in P$, the flow vector $\Theta(p, \cdot)$ satisfies (a) the cycle equation corresponding to each undirected cycle γ of G and (b) the reachability condition corresponding to each node n of G, where the latter is stated with respect to some arbitrary path from n_0 to n;*

(ii) the net flow matrix Θ satisfies the state separation condition for every nodal pair (n, n') with $n \neq n'$; and (iii) for every event separation instance in G, there exists a place $p \in P$ with its flow vector $\Theta(p, \cdot)$ satisfying the corresponding event separation condition.

In Section 3 we employ this result towards the development of a methodology that will support the design of reversibility-enforcing supervisors for bounded PN's.

2.3 Petri-net supervisory control based on Generalized Mutual Exclusion constraints and "monitor" places

In many PN control applications, one seeks to impose a set of constraints on the marking, M, of a plant net, $\mathcal{N} = (P, T, W, M_0)$, that are expressed as a set of linear inequalities of the type

$$A \cdot M \leq b \tag{8}$$

where the elements of matrix A and the right-hand-side (rhs) vector b are non-negative integers. Marking constraints of the type expressed by Equation 8 are known as *Generalized Mutual Exclusion (GME)* constraints. Consider the GME constraint of Equation 8 that is defined by the row $A(i, \cdot)$ of matrix A and the component $b(i)$ of the rhs vector b. Then, according to the theory of [4], this constraint can be imposed on the plant net $\mathcal{N}$ by super-imposing on it a single *"monitor"* place $p_c(i)$; this place must be connected to the rest of the network according to the flow vector:

$$\Theta(p_c(i), \cdot) = -A(i, \cdot) \cdot \Theta \tag{9}$$

and its initial marking must be set to:

$$M_0(p_c(i)) = b(i) \tag{10}$$

Under the aforementioned configuration, $p_c(i)$ enforces the constraint

$$A(i, \cdot) \cdot M \leq b(i) \tag{11}$$

on the markings, M, of the original net, by essentially establishing the invariant

$$A(i, \cdot) \cdot M + M(p_c(i)) = b(i) \tag{12}$$

Equation 12 indicates that the token content, $M(p_c(i))$, of place $p_c(i)$ expresses the "slack" of Constraint 11 under marking M, and justifies the characterization of the control place $p_c(i)$ as a *"monitor"* place.

We conclude this brief discussion on GME constraints and their enforcing monitor places, by establishing the following result, that will be useful in the developments of Section 3:

Lemma 1. *Consider a monitor place $p_c(i)$ that enforces the GME constraint of Equation 11 on a plant net $\mathcal{N}$. Then, every t-semiflow, x, of $\mathcal{N}$ is also a t-semiflow for place $p_c(i)$.*

Proof: We need to show that $\Theta(p_c(i), \cdot) \cdot x = 0$. But this is an immediate implication of Equation 9 and the fact that x is a t-semiflow of the original net $\mathcal{N}$. □

3 A Formal Statement of the Considered Problem and the Proposed Supervisor Design Methodology

Having established in the previous section all the concepts and results that are necessary for the formal development of this work, we can now proceed to the detailed statement of the undertaken problem and the systematic exposition of the methodology proposed for its solution. We start with the formal problem statement.

A formal statement of the problem considered in this work The problem considered in this work can be formally defined as follows: Given a non-reversible, bounded PN $\mathcal{N}$, identify a set of GME constraints

$$A \cdot M \leq b \tag{13}$$

such that

i. when imposed on the plant net $\mathcal{N}$, will incur the reversibility of the controlled system.
ii. Furthermore, the cardinality of the imposed constraint set must not exceed a pre-specified parameter K.
iii. In addition,

$$\forall i, j,\ A(i,j) \in \{0, 1, \ldots, \bar{A}(i,j)\} \text{ and } \forall i,\ b(i) \in \{0, 1, \ldots, \bar{b}(i)\}, \tag{14}$$

where $\bar{A}(i,j)$ and $\bar{b}(i)$ are *finitely* valued, externally provided parameters.
iv. Finally, assuming that every reachable marking $M_i \in R(\mathcal{N}, M_0)$ of $\mathcal{N}$ is associated with some value w_i, the developed supervisor must maximize the total value of the admissible markings, over the set of supervisors satisfying the aforementioned requirements.

In the sequel, a PN supervisor that is defined by Equation 13 for some pricing of matrix A and vector b, will be referred to as the supervisor $\mathcal{S}(A, b)$.

Overview of the proposed solution Next, we provide a *Mixed Integer Programming (MIP)* formulation for the aforestated problem. The objective function of this formulation will express the optimality requirement stated in item (iv) above. Requirement (ii) will be captured by the structure of the decision variables of the presented formulation, while requirements (i) and (iii) will be explicitly encoded in its constraints. More specifically, given a pricing of the matrix A and the rhs vector b, the constraint set must check whether this pricing abides to requirement (iii) and it must also assess the ability of this pricing to satisfy requirement (i), i.e., establish the reversibility of the controlled system. This last requirement further implies that all the markings $M \in R(\mathcal{N}, M_0)$ that remain reachable under the considered GME constraints, are also co-reachable under these constraints. Hence, the constraint set of the proposed formulation must be able to assess the reachability and co-reachability of the markings $M \in R(\mathcal{N}, M_0)$ under the net

supervision by any tentative GME constraint set, $A \cdot M \leq b$, and it must also be able to validate that all reachable markings are also co-reachable. The rest of this section proceeds to the detailed derivation of a formulation that possesses the aforementioned qualities.

Characterizing the net transition firing under supervision by a GME constraint-based supervisor $\mathcal{S}(A, b)$ In order to be able to assess the reachability and co-reachability of the various markings $M \in R(\mathcal{N}, M_0)$ under supervision by a supervisor $\mathcal{S}(A, b)$, it is necessary to characterize how the various transitions, $t \in T$, of the plant net $\mathcal{N}$, retain their fireability in the controlled system. Next, we introduce a set of variables and constraints that will achieve this purpose. The main issue to be addressed is whether a transition t that was fireable in some marking $M_i \in R(\mathcal{N}, M_0)$, leading to another marking $M_j \in R(\mathcal{N}, M_0)$, will remain fireable under supervision by $\mathcal{S}(A, b)$. For this to be true, t must be enabled at M_i by all the monitor places, $p_c(k)$, $k = 1, \ldots, K$, that implement the supervisor $\mathcal{S}(A, b)$. Testing whether transition t is enabled at marking M_i by a monitor place $p_c(k)$ can be done through the employment of a *binary* variable z_{ij}^k, that will be priced to one, if this condition is true, and to zero, otherwise. A set of constraints that will enforce the pricing of z_{ij}^k according to the aforementioned scheme is the following:

$$M_0(p_c(k)) + \sum_{(u,v)\in\xi(i)} \Theta(p_c(k), t(u,v)) + \Theta(p_c(k), t(i,j)) + (z_{ij}^k - 1)L_{ij}^k \geq 0 \quad (15)$$

$$M_0(p_c(k)) + \sum_{(u,v)\in\xi(i)} \Theta(p_c(k), t(u,v)) + \Theta(p_c(k), t(i,j)) - z_{ij}^k U_{ij}^k \leq -1 \quad (16)$$

The parameter $\xi(i)$ appearing in Equations 15 and 16 denotes any path in $R(\mathcal{N}, M_0)$ leading from M_0 to M_i. (u, v) denotes an edge of $\xi(i)$ leading from node M_u to node M_v, and $t(u, v)$ denotes its labelling transition. L_{ij}^k denotes a lower bound for the quantity $M_0(p_c(k)) + \sum_{(u,v)\in\xi(i)} \Theta(p_c(k), t(u,v)) + \Theta(p_c(k), t(i,j))$, and U_{ij}^k denotes an upper bound for the quantity $M_0(p_c(k)) + \sum_{(u,v)\in\xi(i)} \Theta(p_c(k), t(u,v)) + \Theta(p_c(k), t(i,j)) + 1$. Then, it is clear, that, when $M_0(p_c(k)) + \sum_{(u,v)\in\xi(i)} \Theta(p_c(k), t(u,v)) + \Theta(p_c(k), t(i,j)) \geq 0$ – i.e., when transition $t(i, j)$ is enabled by monitor place $p_c(k)$ in marking M_i – the above set of constraints is satisfied by setting $z_{ij}^k = 1$. On the other hand, when $M_0(p_c(k)) + \sum_{(u,v)\in\xi(i)} \Theta(p_c(k), t(u,v)) + \Theta(p_c(k), t(i,j)) < 0$ the above constraint set is satisfied by setting $z_{ij}^k = 0$.

It remains to connect the variables $M_0(p_c(k))$ and $\Theta(p_c(k), \cdot)$ to the primary problem variables, A, b, and explain how to compute the bounds L_{ij}^k and U_{ij}^k employed in the above equations. Connecting $M_0(p_c(k))$ and $\Theta(p_c(k), \cdot)$ to the variables A, b can be done straightforwardly through Equations 9 and 10; the corresponding substitutions respectively transform Equations 15 and 16 to:

$$b(k) - \sum_{(u,v)\in\xi(i)} A(k,\cdot)\cdot\Theta(\cdot,t(u,v)) - A(k,\cdot)\cdot\Theta(\cdot,t(i,j)) + (z_{ij}^k - 1)L_{ij}^k \geq 0 \quad (17)$$

$$b(k) - \sum_{(u,v)\in\xi(i)} A(k,\cdot)\cdot\Theta(\cdot,t(u,v)) - A(k,\cdot)\cdot\Theta(\cdot,t(i,j)) - z_{ij}^k U_{ij}^k \leq -1 \quad (18)$$

Finally, it should be clear from the structure of Constraints 17 and 18 that the bound L_{ij}^k (resp., U_{ij}^k), defined above, can be obtained by minimizing (resp., maximizing) the quantity $b(k) - \sum_{(u,v)\in\xi(i)} A(k,\cdot)\cdot\Theta(\cdot,t(u,v)) - A(k,\cdot)\cdot\Theta(\cdot,t(i,j))$ over the space defined by the admissible ranges of the involved variables $A(k,\cdot)$ and $b(k)$ (c.f., item (iii) in the formal problem statement provided at the beginning of this section).

Once variables z_{ij}^k have been properly priced for all k, the feasibility of $M_i[t(i,j)\rangle M_j$ can be assessed by introducing another *real* variable, z_{ij}, that is priced according to the following constraints:

$$z_{ij} \leq z_{ij}^k, \quad \forall k \in \{1,\ldots,K\} \quad (19)$$

$$z_{ij} \geq \sum_{k=1}^{K} z_{ij}^k - K + 1 \quad (20)$$

$$0 \leq z_{ij} \leq 1 \quad (21)$$

To understand the pricing logic behind Constraints 19–21, first notice that Constraint 21 restricts the variable z_{ij} within the interval $[0,1]$. Then, Constraint 19 sets it to zero, as long as any of the variables z_{ij}^k is priced to zero – and therefore, the corresponding monitor place $p_c(k)$ disables $t(i,j)$. On the other hand, when all variables z_{ij}^k are priced to one, Constraint 20 forces variable z_{ij} to its extreme value of one.

Characterizing the reachability of the markings $M_i \in R(\mathcal{N},M_0)$ under supervision by a GME constraint-based supervisor $\mathcal{S}(A,b)$ The availability of the variables z_{ij}, defined above, subsequently enables the characterization of the reachability of the various markings $M_i \in R(\mathcal{N},M_0)$ under supervision by the GME constraint-based supervisor $\mathcal{S}(A,b)$. This can be done by introducing the *real* variables $y_i^l,\ 0 \leq i \leq |R(\mathcal{N},M_0)|,\ 0 \leq l \leq \bar{l}$, and pricing them so that $y_i^l = 1$ indicates that marking M_i is reachable from the initial marking M_0 under supervision by $\mathcal{S}(A,b)$ and the minimum length of any transition sequence leading from M_0 to M_i is l; if M_i is not reachable from M_0 under supervision by $\mathcal{S}(A,b)$, y_i^l should be set to zero, for all l. Clearly, in order to satisfy this definition of y_i^l, $\bar{l}$ must be set to the length of the maximum path in $\mathcal{G}(\mathcal{N},M_0)$ that starts from M_0 and contains no cycles. Then, a set of constraints that achieves the pricing of y_i^l described above, is as follows:

$$y_i^0 = \begin{cases} 1, i = 0 \\ 0, i \neq 0 \end{cases} \tag{22}$$

$$0 \leq y_i^l, \quad \forall i \in \{1, \ldots, |R(\mathcal{N}, M_0)|\}, \; l \in \{1, \ldots, \bar{l}\} \tag{23}$$

$$\sum_{l=0}^{\bar{l}} y_i^l \leq 1 \tag{24}$$

$$\delta_{ji}^l \leq y_j^{l-1}, \quad \forall j : (M_j, M_i) \in \mathcal{G}(\mathcal{N}, M_0) \tag{25}$$

$$\delta_{ji}^l \leq z_{ji}, \quad \forall j : (M_j, M_i) \in \mathcal{G}(\mathcal{N}, M_0) \tag{26}$$

$$y_i^l \leq \sum_j \delta_{ji}^l \tag{27}$$

$$y_i^l \geq y_j^{l-1} + z_{ji} - 1 - \sum_{q=0}^{l-1} y_i^q, \quad \forall j : (M_j, M_i) \in \mathcal{G}(\mathcal{N}, M_0) \tag{28}$$

Constraint 22 expresses the fact that marking M_0 is reachable from itself in zero steps, under supervision by $\mathcal{S}(A, b)$, and this is the only marking in $R(\mathcal{N}, M_0)$ possessing this property. Constraint 23 states the nonnegative real nature of variables $y_i^l, \; i > 0, \; l > 0$, while Constraint 24 expresses the fact that, according to the pricing scheme discussed above, only one of the variables $y_i^l, \; 0 \leq l \leq \bar{l}$, can be priced to one. Constraints 25, 26 and 27 express the fact that, under supervision by $\mathcal{S}(A, b)$, there is a minimal path from marking M_0 to marking M_i of length l, only if there is a minimal path of length $l-1$ from M_0 to some marking M_j such that (i) $(M_j, M_i) \in \mathcal{G}(\mathcal{N}, M_0)$ and (ii) this transition remains feasible under $\mathcal{S}(A, b)$. In particular, variables δ_{ji}^l is a set of auxiliary *real* variables that are used to force y_i^l to zero every time that the aforestated condition is violated for all the markings $M_j \in R(\mathcal{N}, M_0)$ such that $(M_j, M_i) \in \mathcal{G}(\mathcal{N}, M_0)$. On the other hand, Constraint 28 tends to price variable y_i^l to one every time that there exists a marking M_j such that (i) $(M_j, M_i) \in \mathcal{G}(\mathcal{N}, M_0)$, (ii) this transition remains feasible under $\mathcal{S}(A, b)$, and (iii) M_j is reachable from M_0 under supervision by $\mathcal{S}(A, b)$ through a minimal path of length $l-1$; however, this pricing is enforced only when the quantity $\sum_{q=0}^{l-1} y_i^q$ appearing in the right-hand-side of this constraint is equal to zero – i.e., only when the marking M_i cannot be reached from the initial marking M_0 through a path of smaller length.

Characterizing the co-reachability of the markings $M_i \in R(\mathcal{N}, M_0)$ under supervision by a GME constraint-based supervisor $\mathcal{S}(A, b)$ It is well-known that the co-reachability of a marking $M_i \in R(\mathcal{N}, M_0)$ is equivalent to the reachability of the same marking in the graph $\mathcal{G}^R(\mathcal{N}, M_0)$, obtained from $\mathcal{G}(\mathcal{N}, M_0)$ by reversing all its arcs. In the light of this observation, the set of constraints

characterizing the co-reachability of the markings $M_i \in R(\mathcal{N}, M_0)$, under supervision by a GME constraint-based supervisor $\mathcal{S}(A, b)$, can be obtained through a straightforward modification of the constraint set 22–28, characterizing the reachability of these markings. More specifically, let ψ_i^l be a *real* variable that will be priced to one, if $M_i \in R(\mathcal{N}, M_0)$ is co-reachable under supervision by $\mathcal{S}(A, b)$, and a minimal transition sequence leading from M_i to M_0 has a length equal to l; otherwise, ψ_i^l should be priced to zero. By following a logic similar to that employed in the previous paragraph for the pricing of variables y_i^l, we obtain the following set of constraints for the pricing of variables ψ_i^l:

$$\psi_i^0 = \begin{cases} 1, i = 0 \\ 0, i \neq 0 \end{cases} \tag{29}$$

$$0 \leq \psi_i^l, \quad \forall i \in \{1, \ldots, |R(\mathcal{N}, M_0)|\}, \; l \in \{1, \ldots, \tilde{l}\} \tag{30}$$

$$\sum_{l=0}^{\tilde{l}} \psi_i^l \leq 1 \tag{31}$$

$$\eta_{ij}^l \leq \psi_j^{l-1}, \quad \forall j : (M_i, M_j) \in \mathcal{G}(\mathcal{N}, M_0) \tag{32}$$

$$\eta_{ij}^l \leq z_{ij}, \quad \forall j : (M_i, M_j) \in \mathcal{G}(\mathcal{N}, M_0) \tag{33}$$

$$\psi_i^l \leq \sum_j \eta_{ij}^l \tag{34}$$

$$\psi_i^l \geq \psi_j^{l-1} + z_{ij} - 1 - \sum_{q=0}^{l-1} \psi_i^q, \quad \forall j : (M_i, M_j) \in \mathcal{G}(\mathcal{N}, M_0) \tag{35}$$

The parameter $\tilde{l}$, appearing in Equations 30 and 31, denotes the length of the maximum path in $\mathcal{G}^R(\mathcal{N}, M_0)$ that leads from node M_0 to node M_i and contains no cycles, and the auxiliary variables η_{ij}^l, that appear in Constraints 32 and 33, play a role identical to that played by variables δ_{ji}^l in Constraints 25 and 26.

Characterizing the closure of the sub-space that is reachable and co-reachable under supervision by a GME constraint-based supervisor $\mathcal{S}(A, b)$ Let x_i be a *real* variable that will be priced to one when the marking $M_i \in R(\mathcal{N}, M_0)$ is reachable and co-reachable under supervision by $\mathcal{S}(A, b)$, and it will be priced to zero, otherwise. Then, in the light of the above characterizations of reachability and co-reachability, the desired pricing of x_i can be enforced by the following constraints:

$$x_i \leq \sum_{l=0}^{\bar{l}} y_i^l \tag{36}$$

$$x_i \leq \sum_{l=0}^{\tilde{l}} \psi_i^l \tag{37}$$

$$x_i \geq \sum_{l=0}^{\bar{l}} y_i^l + \sum_{l=0}^{\tilde{l}} \psi_i^l - 1 \tag{38}$$

$$0 \leq x_i \leq 1 \tag{39}$$

Constraint 39 restricts x_i in the interval $[0, 1]$. Then, Constraints 36 and 37 force it to zero, when marking M_i is not reachable or co-reachable. On the other hand, if M_i is both reachable and co-reachable, Constraint 38 forces x_i to its extreme value of one.

Finally, the availability of variables x_i allows us to express the requirement for closure of the sub-space of $R(\mathcal{N}, M_0)$ that is reachable and co-reachable under supervision by $\mathcal{S}(A, b)$, through the following constraint:

$$(1 - x_i) + x_j \geq z_{ij}, \quad \forall i, j : (M_i, M_j) \in \mathcal{G}(\mathcal{N}, M_0) \tag{40}$$

When $x_i = 1$ and $x_j = 0$ – i.e., when x_i belongs to the target space of markings that are reachable and co-reachable under supervision by $\mathcal{S}(A, b)$, but x_j does not belong to this set – Constraint 40 forces variable z_{ij} to zero – i.e., it requires that the corresponding transition $M_i[t(i, j)\rangle M_j$ is disabled by $\mathcal{S}(A, b)$. In any other case, the left-hand-side of Constraint 40 is greater than or equal to one, and therefore, the constraint becomes inactive.

The objective function of the proposed formulation The objective function of the considered formulation is straightforwardly expressed as follows:

$$\max \sum_i w_i x_i \tag{41}$$

Proving the correctness of the proposed formulation Next, we state and prove the correctness of the derived formulation.

Theorem 2. *The formulation of Equations 14,17–41 returns an optimal solution to the problem stated at the beginning of this section, provided that such a solution exists; otherwise, this formulation will be infeasible.*

Proof: First, let us suppose that the aforementioned formulation returns a feasible solution. Then, it is clear from the earlier discussion of the various constraints of the considered formulation, that the set of markings $M_i \in R(\mathcal{N}, M_0)$ with $x_i = 1$ and the edges (M_i, M_j) of $\mathcal{G}(\mathcal{N}, M_0)$ with $z_{ij} = 1$, in the returned solution, define a strongly connected subgraph of $\mathcal{G}(\mathcal{N}, M_0)$ containing the initial marking M_0; let us denote this subgraph by $\mathcal{G}^C(\mathcal{N}, M_0)$. Next we show that $\mathcal{G}^C(\mathcal{N}, M_0)$ is the reachability graph of the net $\mathcal{N}^C$, that is obtained from the plant net $\mathcal{N}$ by super-imposing on it the monitor places that implement

the GME constraint set $A \cdot M \leq b$, where A, b have the values returned by the considered formulation. To establish this result, it is sufficient to show that the net $\mathcal{N}^C$ satisfies the conditions of Theorem 1 with respect to $\mathcal{G}^C(\mathcal{N}, M_0)$. This can be shown as follows: First notice that the state separation condition over $\mathcal{G}^C(\mathcal{N}, M_0)$ is immediately satisfied by $\mathcal{N}^C$, since its marking subsumes the marking of the original net $\mathcal{N}$. The satisfaction of the reachability condition for every node of $\mathcal{G}^C(\mathcal{N}, M_0)$ is a consequence of Constraints 17–28. For event separation instances of $\mathcal{G}^C(\mathcal{N}, M_0)$ that were already present in $\mathcal{G}(\mathcal{N}, M_0)$, there must be a place p in the original net $\mathcal{N}$ that satisfies the corresponding event separation condition. For the remaining event separation instances, the definition of $\mathcal{G}^C(\mathcal{N}, M_0)$ implies that the relevant variables z_{ij} were priced to zero in the returned solution, and therefore, for each of them, there exists a place $p_c(k),\ k \in \{1, \ldots, K\}$, that satisfies the corresponding event separation condition. For places p in the original net $\mathcal{N}$, the cycle equations for the various cycles of $\mathcal{G}^C(\mathcal{N}, M_0)$ are immediately satisfied by the fact that $\mathcal{G}^C(\mathcal{N}, M_0)$ is a subgraph of $\mathcal{G}(\mathcal{N}, M_0)$. For the monitor places $p_c(k),\ k \in \{1, \ldots, K\}$, the satisfaction of the cycle equations for the cycles of $\mathcal{G}^C(\mathcal{N}, M_0)$ is guaranteed by Lemma 1. Finally, the optimality of the supervisor $\mathcal{S}(A, b)$, that is returned by the considered formulation, is guaranteed by the specification of the objective function (c.f. Equation 41).

On the other hand, if the considered formulation is infeasible, then it is impossible to identify a strongly connected subgraph of $\mathcal{G}(\mathcal{N}, M_0)$ that contains the initial marking M_0 and can be separated from $\mathcal{G}(\mathcal{N}, M_0)$ by using K GME constraints with the corresponding matrix A and rhs vector b priced in the prespecified ranges. Hence, it can be concluded that the supervisor design problem defined at the beginning of this section, is infeasible. □

4 Example

In this section, we demonstrate the implementation and the efficacy of the design methodology developed in Section 3, by applying it to the design of a liveness-enforcing supervisor for the PN depicted in Figure 1.

Interpreting the PN of Figure 1 as a process-resource net The PN in Figure 1 models a Resource Allocation System (RAS), consisting of three resource types, R_1, R_2, and R_3, with respective capacities $C_1 = C_3 = 1$, and $C_2 = 2$, and supporting two process types, JT_1 and JT_2. The process plans of these two process types are respectively modelled by the paths $< t_{10}p_{11}t_{11}p_{12}t_{12}p_{13}t_{13} >$ and $< t_{20}p_{21}t_{21}p_{22}t_{22}p_{23}t_{23} >$; thus, it can be seen that (i) each process consists of three consecutive stages, (ii) the execution of each processing stage by some process instance requires the exclusive allocation of a single unit from a certain resource type, and (iii) a process instance can release the resource currently allocated to it and advance to the next processing stage only when it has secured the allocation of the next required resource. Finally, the places p_{10} and p_{20} are characterized as the "idle places" of the corresponding processes, and their

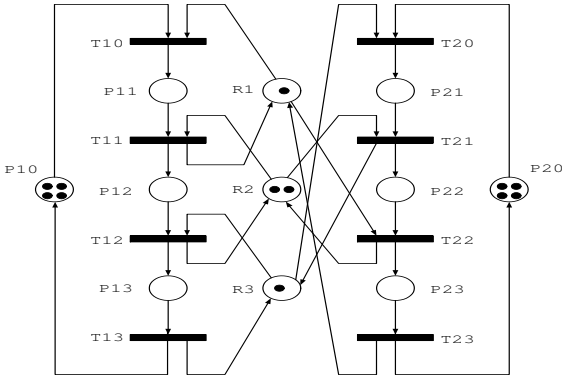

Fig. 1. The process-resource net considered in the example of Section 4

initial marking, $M_0(p_{i0})$, $i = 1, 2$, establishes an upper bound to the number of instances of process type JT_i that can be simultaneously loaded into the system.

Liveness-enforcing supervision of process-resource nets based on Generalized Mutual Exclusion constraints and monitor places [5] The reachability space, $R(\mathcal{N}, M_0)$, for the PN depicted in Figure 1 is provided in Figure 2, while the detailed characterization of the markings corresponding to the various nodes of the graph of Figure 2 can be found in Table 1.[6] It can be seen in Figure 2 that the considered net is not reversible. In particular, there is a class of states depicted by the darker-shaded nodes in Figure 2 such that every time that the net transitions to one of these states, there is no path to the initial state M_0; for further reference, this class of markings will be characterized as *unsafe*. From a more conceptual standpoint, the net non-reversibility can be interpreted by the development of a RAS deadlock, i.e., the entanglement of a subset of the running processes in a circular waiting pattern, where each process in this subset waits upon some other process of this set to release its currently allocated resource. Furthermore, the net non-reversibility implies that the underlying RAS might not be able to complete the currently loaded processes, under normal operation.[7]

The last fifteen years have seen the development of an extensive body of research seeking to develop supervisors that will enforce the reversibility of the considered class of process-resource nets. Generally speaking, these supervisors

[5] We remind the reader that, in the considered class of process-resource nets, reversibility and liveness are equivalent concepts, and that the term "liveness-enforcing supervision (LES)" has prevailed over the term "reversibility-enforcing supervision".

[6] Table 1 provides only the markings of the places corresponding to the various processing stages, since the markings of the remaining places can be easily obtained from the net invariants corresponding to (i) the reusability of the system resources and (ii) the circuits established by the introduction of the process idle places.

[7] i.e., without external intervention to resolve the developed deadlock.

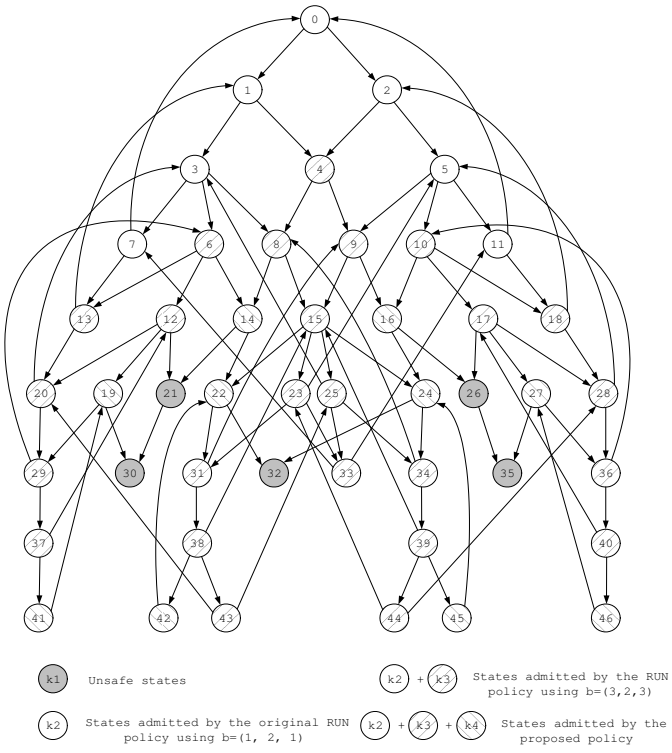

Fig. 2. The reachability graph of the process-resource net of Figure 1, and a comparison of the sub-spaces admitted by the supervisors of Equations 42, 43 and 45

seek to constrain the behavior of the underlying process-resource net within a strongly connected component of its safe sub-space, that also contains the initial marking M_0.[8] Furthermore, one particular class of such supervisors, known as *algebraic*, seeks to establish the reversibility of the underlying process-resource net by restricting the number of process instances that can execute simultaneously various subsets of processing stages; hence, each such restriction can be expressed as a GME constraint involving the subset of places in the underlying process-resource net that correspond to the repsective processing stages. For the process-resource net of Figure 1, two such algebraic reversibility-enforcing supervisors have been developed in [6]. They are respectively expressed by the

[8] Ideally, one would like to obtain the *maximally permissive* supervisor that will admit the entire safe sub-space of the underlying process-resource net, but it has been shown that, for the considered PN sub-class, the recognition of the net safe markings is an NP-complete problem [8].

Table 1. The markings of the reachability space depicted in Figure 2

State	p_{11}	p_{12}	p_{13}	p_{21}	p_{22}	p_{23}	State	p_{11}	p_{12}	p_{13}	p_{21}	p_{22}	p_{23}
0	0	0	0	0	0	0	24	0	1	0	1	1	0
1	1	0	0	0	0	0	25	0	1	0	0	0	1
2	0	0	0	1	0	0	26	1	0	0	0	2	0
3	0	1	0	0	0	0	27	0	0	0	1	2	0
4	1	0	0	1	0	0	28	0	0	0	0	1	1
5	0	0	0	0	1	0	29	1	1	1	0	0	0
6	1	1	0	0	0	0	30	1	2	0	1	0	0
7	0	0	1	0	0	0	31	1	0	1	0	1	0
8	0	1	0	1	0	0	32	1	1	0	1	1	0
9	1	0	0	0	1	0	33	0	0	1	0	0	1
10	0	0	0	1	1	0	34	0	1	0	1	0	1
11	0	0	0	0	0	1	35	1	0	0	1	2	0
12	0	2	0	0	0	0	36	0	0	0	1	1	1
13	1	0	1	0	0	0	37	0	2	1	0	0	0
14	1	1	0	1	0	0	38	0	1	1	0	1	0
15	0	1	0	0	1	0	39	0	1	0	0	1	1
16	1	0	0	1	1	0	40	0	0	0	0	2	1
17	0	0	0	0	2	0	41	1	2	1	0	0	0
18	0	0	0	1	0	1	42	1	1	1	0	1	0
19	1	2	0	0	0	0	43	0	1	1	0	0	1
20	0	1	1	0	0	0	44	0	0	1	0	1	1
21	0	2	0	1	0	0	45	0	1	0	1	1	1
22	1	1	0	0	1	0	46	0	0	0	1	2	1
23	0	0	1	0	1	0							

following GME constraint sets:

$$\begin{bmatrix} 1 & & 1\,1\,1 \\ 1\,1 & & 1\,1 \\ 1\,1\,1\,1 & & \end{bmatrix} \cdot \hat{M} \leq \begin{bmatrix} 1 \\ 2 \\ 1 \end{bmatrix}. \tag{42}$$

$$\begin{bmatrix} 1 & & 1\,1\,1 \\ 1\,1 & & 1\,1 \\ 1\,1\,1\,1 & & \end{bmatrix} \cdot \hat{M} \leq \begin{bmatrix} 3 \\ 2 \\ 3 \end{bmatrix}. \tag{43}$$

where $\hat{M} = (M(p_{11}), M(p_{12}), M(p_{13}),\ M(p_{21}), M(p_{22}), M(p_{23}))^T$. Notice that the Constraint set 43 is a relaxation of the Constraint set 42 since $A_1 = A_2$ and $b_1 \leq b_2$. Therefore, the supervisor established by the Constraint set 43 is expected to be more permissive than the supervisor established by the Constraint set 42, and this is indeed reflected in Figure 2 that also depicts the sub-spaces admitted by each of these two supervisors.

Obtaining a more permissive supervisor for the net of Figure 1 In this work, we employed the formulation of Equations 14,17–41 in order to compute an algebraic reversibility-enforcing supervisor for the net of Figure 1 that possesses the same computational complexity with the supervisors of Equations 42 and 43, but it is *maximally permissive*. In other words, we sought to obtain a pair

$$A = \begin{bmatrix} a_{11}\ a_{12}\ a_{13}\ a_{14}\ a_{15}\ a_{16} \\ a_{21}\ a_{22}\ a_{23}\ a_{24}\ a_{25}\ a_{26} \\ a_{31}\ a_{32}\ a_{33}\ a_{34}\ a_{35}\ a_{36} \end{bmatrix}, b = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix}, \tag{44}$$

such that (i) the supervisor $\mathcal{S}(A,b) \equiv A \cdot \hat{M} \leq b$ will accept a strongly connected component of the safe sub-space depicted in Figure 2 containing the initial marking M_0, and furthermore, (ii) the number of markings accepted by this supervisor is the maximal possible that can be accepted by any algebraic supervisor possessing the aforementioned structure.

Foregoing the straightforward implementational details, for the sake of brevity, we proceed to the presentation of the results of our computation. The supervisor returned by the proposed formulation is:

$$\mathcal{S}(A,b) = \mathcal{S}\left(\begin{bmatrix} 1\,0\,0\,0\,3\,0 \\ 0\,1\,0\,2\,0\,0 \\ 2\,2\,0\,2\,3\,0 \end{bmatrix}, \begin{bmatrix} 6 \\ 3 \\ 8 \end{bmatrix}\right). \tag{45}$$

The sub-space admitted by the supervisor of Equation 45 is also depicted in Figure 2. As it can be seen in this figure, the obtained supervisor manages to recognize the entire safe space of the considered process-resource net, and therefore, it is optimal. Hence, this example corroborates the efficacy and analytical power of the proposed methodology.

5 Enhancements and Extensions of the Proposed Approach

In this section we consider some enhancements of the basic formulation developed in Section 3, and some modifications of the underlying supervisor design methodology, that will allow the accommodation of additional considerations, like the uncontrollability of certain transitions of the plant net $\mathcal{N}$, the potentially prohibitive computational cost resulting from the very large size of the underlying reachability space $\mathcal{G}(\mathcal{N}, M_0)$, and the imposition of additional costs and/or restrictions on the elements of the matrix A. We deal with each of these issues in a separate paragraph.

Accommodating the uncontrollability of the plant transitions In certain cases it is possible that some of the plant transitions $t \in T$ cannot have their firing controlled by an external supervisor, but they will fire spontaneously any time that they are enabled by the plant. Their presence partitions the transition set T to the subset T^U of *uncontrollable* transitions, and its complement T^C of the *controllable* ones. Clearly, a reversibility-enforcing supervisor $\mathcal{S}(A,b)$ should not try to disable the fireability of any transition $t \in T^U$, whenever such a transition is enabled by the places of the original plant net $\mathcal{N}$. This requirement can be easily introduced in the MIP formulation of Equations 14,17–41, by adding to it the following constraint:

$$\forall i,j : (M_i, M_j) \in \mathcal{G}(\mathcal{N}, M_0) \ \wedge \ t(i,j) \in T^U, \quad z_{ij} \geq x_i \tag{46}$$

Constraint 46 essentially requests that the uncontrollable transition between markings M_i and M_j is enabled by all the monitor places implementing the

supervisor $\mathcal{S}(A, b)$, but this request is enforced only for the transitions emanating from markings $M_i \in R(\mathcal{N}, M_0)$ that remain accessible during the operation of the controlled net.

Dealing with complexity considerations It is clear from the structure of the formulation of Equations 14,17–41, that it involves a number of variables and constraints that is polynomially related to the size of the original reachability graph $\mathcal{G}(\mathcal{N}, M_0)$ of the underlying plant net $\mathcal{N}$. It is well-known, though, that, in general, the size of $\mathcal{G}(\mathcal{N}, M_0)$ is a super-polynomial function of the size of the net $\mathcal{N}$, and therefore, there might be cases where the generation and solution of the proposed formulation will be a computationally intractable task. In these cases, the approach proposed in Section 3 can still be pursued on a judiciously selected subspace of $\mathcal{G}(\mathcal{N}, M_0)$. The main requirements imposed on this subspace are that (i) it includes a strongly connected component containing the initial state M_0, and (ii) there are no uncontrollable transitions leading from this subspace to the rest of the graph $\mathcal{G}(\mathcal{N}, M_0)$; otherwise, its selection is left to the jurisdiction of the designer. Let $\mathcal{C}$ denote the *cut* from the subspace of $\mathcal{G}(\mathcal{N}, M_0)$ to be considered during the application of the proposed methodology, to the rest of $\mathcal{G}(\mathcal{N}, M_0)$. Then, the only modification required in the formulation of Equations 14,17–41 so that it effectively applies on the subgraph of $\mathcal{G}(\mathcal{N}, M_0)$ mentioned above, is that it must also contain a set of variables $z_{ij}^k,\ k \in \{1, \ldots, K\},\ z_{ij}$, for every edge $(M_i, M_j) \in \mathcal{C}$, priced according to the constraint set 17–21, and the additional constraint:

$$\forall i, j : (M_i, M_j) \in \mathcal{C}, \quad z_{ij} \leq 1 - x_i \tag{47}$$

Constraint 47 requests the disabling by the developed supervisor of the transitions in the cut $\mathcal{C}$, but, similar to Constraint 46, it enforces this requirement only for those transitions of $\mathcal{C}$ that emanate from markings that remain reachable in the operation of the controlled net.

Restricting the elements of matrix A In certain cases, it might be pertinent, for computational or more general implementational purposes, to put a cost structure on the elements of matrix A. As a case in point, it will be generally desirable to keep the elements of matrix A as small as possible. Of course, this additional requirement should not compromise the primary objective of the design process, which is stated in Equation 41. Hence, these additional concerns can be addressed through a *hierarchical goal programming* [10] approach. According to this approach, the formulation of Equations 14,17–41 is initially solved to optimality without any consideration of the extra concerns, in order to obtain the value of an optimal solution to the original problem stated in Section 3. Subsequently, the entire formulation is resolved with a new objective that expresses the cost criterion imposed on the elements of matrix A, while the desired value of the derived solution with respect to the original objective is fixed to the earlier computed optimal value and it is communicated to the new problem as an additional constraint; we leave to the reader the implementational details of this idea.

6 Conclusions

This paper proposed an analytical method for the synthesis of reversibility-enforcing supervisors for bounded Petri nets. The proposed method was based upon recent developments from (i) the theory of regions, that enables the design of Petri nets with pre-specified behavioral requirements, and (ii) the theory concerning the imposition of generalized mutual exclusion constraints on the net behavior through monitor places. The derived methodology takes the form of a Mixed Integer Programming formulation, which is readily solvable through canned optimization software. A small example borrowed from the theory of liveness-enforcing supervision for process-resource nets demonstrated the efficacy of the proposed approach, while the last part of the paper discussed extensions of the presented method so that it accommodates uncontrollable behavior and any potential complications arising from the large-scale nature of the underlying plant nets and their behavioral spaces.

References

1. E. Badouel and P. Darondeau. Theory of regions. In W. Reisig and G. Rozenberg, editors, *LNCS 1491 – Advances in Petri Nets: Basic Models*, pages 529–586. Springer-Verlag, 1998.
2. A. Ghaffari, N. Rezg, and X. Xie. Design of a live and maximally permissive petri net controller using the theory of regions. *IEEE Trans. on Robotics & Automation*, 19:137–141, 2003.
3. A. Giua, F. DiCesare, and M. Silva. Generalized mutual exclusion constraints on nets with uncontrollable transitions. In *Proceedings of the 1992 IEEE Intl. Conference on Systems, Man and Cybernetics*, pages 974–979. IEEE, 1992.
4. J. O. Moody and P. J. Antsaklis. *Supervisory Control of Discrete Event Systems using Petri nets.* Kluwer Academic Pub., Boston, MA, 1998.
5. T. Murata. Petri nets: Properties, analysis and applications. *Proceedings of the IEEE*, 77:541–580, 1989.
6. J. Park and S. Reveliotis. Algebraic synthesis of efficient deadlock avoidance policies for sequential resource allocation systems. *IEEE Trans. on R&A*, 16:190–195, 2000.
7. P. J. G. Ramadge and W. M. Wonham. The control of discrete event systems. *Proceedings of the IEEE*, 77:81–98, 1989.
8. S. A. Reveliotis. *Real-time Management of Resource Allocation Systems: A Discrete Event Systems Approach.* Springer, NY, NY, 2005.
9. M. Uzam. An optimal deadlock prevention policy for flexible manufacturing systems using petri net models with resources and the theory of regions. *Intl. Jrnl of Advanced Manufacturing Technology*, 19:192–208, 2002.
10. W. L. Winston. *Introduction To Mathematical Programming: Applications and Algorithms, 2nd ed.* Duxbury Press, Belmont, CA, 1995.
11. M. Zhou and M. P. Fanti (editors). *Deadlock Resolution in Computer-Integrated Systems.* Marcel Dekker, Inc., Singapore, 2004.