Synthesis of Deadlock Prevention Supervisors Using Petri Nets

Marian V. Iordache, John O. Moody, *Member, IEEE*, and Panos J. Antsaklis, *Fellow, IEEE*

***Abstract*—Given an arbitrary Petri net (PN) structure, which may have uncontrollable and unobservable transitions, the deadlock prevention procedure presented here determines a set of linear inequalities on the PN markings. When the PN is supervised so that its markings satisfy these inequalities, the supervised net is proved to be deadlock-free for all initial markings that satisfy the supervision constraints. Deadlock-freedom implies that there will always be at least one transition that is enabled in the closed-loop (supervised) system. The method is not guaranteed to ensure liveness, as it can be applied to systems that cannot be made live under any circumstances. However, for controllable and observable PNs, it is shown that, when the method ensures liveness as well, the liveness-ensuring supervisor is least restrictive. Moreover, it is shown that the method is not restrictive even for PNs in which not all transitions can be made live. The procedure allows automated synthesis of the supervisors.**

***Index Terms*—Deadlock prevention, Petri nets, supervisory control.**

I. Introduction

WE PRESENT a procedure for the automatic generation of deadlock prevention supervisors for arbitrary Petri net (PN) structures. These supervisors are specified independently of the initial marking, prevent deadlock, and are not restrictive. Deadlock prevention means that the closed-loop plant/supervisor system is deadlock-free, that is, all (total) deadlock states and all states from which (total) deadlock is unavoidably reached are avoided. The results presented in this paper are new and, to the authors' knowledge, are superior to related results in the literature.

The deadlock prevention method presented here uses PN models for the plant and results in a PN model of the supervisor, providing a unified formalism for representing the closed-loop system. The method presents the conditions necessary to ensure deadlock freedom as a set of linear integer inequalities. Such formulation is important because it can be used directly in optimization problems, e.g., determining the minimum number of resources a system requires using a linear integer program. The method is flexible enough to incorporate desired constraint specifications on the markings of the plant. This method is appropriate for use on nets that may not be structurally live, i.e., nonrepetitive systems for which liveness cannot be enforced under any circumstances. When the procedure is applied to repetitive PNs, liveness may be the result. We show that, in this case, under assumptions always satisfied by controllable and observable PNs, the resulting liveness-ensuring supervisor is least restrictive, i.e., no liveness-ensuring supervisor will ever allow a transition to fire that our supervisor would prevent from firing. (A controllable and observable PN is a PN without uncontrollable and unobservable transitions, that is, a PN in which a supervisor can directly inhibit/observe any transition firing.) The procedure we present can be computationally expensive, however, all computations are performed off-line. A supervisor resulting from our deadlock *prevention* method requires very little in terms of computational resources at run time. The method is an iterative approach that removes new potential deadlock situations at every iteration. When (and if) the procedure terminates, the control designer is presented with either a supervised net that is guaranteed to be deadlock-free or, in the case of controllable and observable systems, with an indication that the plant cannot be made deadlock-free under any circumstances.

The method we use defines the supervisor via linear marking inequalities. The supervisor is built using supervision based on place invariants [1]–[3]. Thus, the supervised PN in our approach can be represented as the original PN plus a number of additional places, as is in other approaches, such as [4]–[6]. It has been noticed that structural properties allow the synthesis of deadlock prevention supervisors to be carried out independently of the initial marking (e.g., [7]); similarly, in our approach, the supervisor is defined for all initial markings satisfying the marking inequalities generated by our procedure. It is well known that deadlock in PNs is related to *siphons* (e.g., [8]). As in other previous methodologies, e.g., [4], [6], [7], [9]–[11], we use control places to prevent the total marking in the siphons from becoming zero. In [4], it has been noticed that such siphon control is not enough to guarantee deadlock prevention, since new siphons may appear by adding control places. This problem has been solved in [4] for a subclass of bounded and conservative PNs by using more restrictive control policies, and liveness enforcement has thus been achieved. In [6], successive control of the siphons is used, until no new siphons appear. This is also one of the ideas of our procedure. One of the problems which appears by successively controlling the siphons in an ordinary PN is that the PN can stop being ordinary. Controlling siphons in a PN which is not ordinary is harder. A related result is given in

Manuscript received May 1, 2001. This paper was recommended for publication by Associate Editor S. Reveliotis and Editor N. Viswanadham upon evaluation of the reviewers' comments. This work was supported in part by the Army Research Office (DAAG55-98-1-0199), the National Science Foundation (NSF ECS99-12458), the DARPA/ITO-NEST Program (AF-F30602-01-2-0526), and the Lockheed Martin Corporation.

M. V. Iordache and P. J. Antsaklis are with the Department of Electrical Engineering, University of Notre Dame, Notre Dame, IN 46556 USA (e-mail: iordache.1@nd.edu; antsaklis.1@nd.edu).

J. O. Moody is with the Lockheed Martin Federal Systems, Owego, NY 13827-3998 USA (e-mail: john.moody@lmco.com).

Publisher Item Identifier S 1042-296X(02)01378-2.

[12], but we cannot use it as we desire our method to be as permissive as possible. Instead we transform the PN at the different stages of the procedure back into ordinary PNs, by adapting a technique from [6]. In order to have a method effective for nonrepetitive PNs, we define certain repetitive subnets of a PN as active subnets. Based on this idea, we then define a subclass of siphons, called active siphons, and prove new results which are fundamental for our method. These developments can be found in Section II. We give more details on active subnets and active siphons in [13]. The procedure that leads to deadlock-free PNs is described in Section III and is illustrated via an example in Section IV. The main theoretical results are given in Section V. Some important proofs and the computation of active subnets are included in the Appendix.

Finally, note that when the PNs are bounded and the initial marking is fixed, it is possible to transform the problem from the PN framework to finite automata and so to solve the problem by using finite automata methodologies, such as supervisory control techniques. However, the approach presented here makes no assumptions about the PN structure: the PN may be unbounded, nonrepetitive, generalized (i.e., with arc weights greater than one) and it may have uncontrollable and unobservable transitions. Furthermore, the usage of PNs in deadlock prevention may be preferable because deadlock often occurs in systems with concurrency, which are better described by PNs.

II. Preliminary Results

We consider PN structures of the form $\mathcal{N} = (P, T, F, W)$, where P is the set of places, T the set of transitions, F the set of transition arcs, and W the weight function. A PN with an initial marking μ_0 is denoted by $(\mathcal{N}, \mu_0)$. We consider the other usual PN notations [14].

We say that a PN can be made deadlock-free/live if there is a supervisor and an initial marking μ_0 such that the supervised PN is deadlock-free/live. In a PN, a transition t can be made live if there is a supervisor and an initial marking μ_0 such that t is live in the supervised PN.

A PN $\mathcal{N}$ is ordinary if $\forall f \in F: W(f) = 1$. We will refer to slightly more general PNs in which only the arcs from places to transitions have to have weights equal to one: $\forall p \in P, \forall t \in T$, if $(p,t) \in F$ then $W(p,t) = 1$. We call such (partially ordinary) PNs PT-ordinary. Our deadlock prevention procedure applies to arbitrary PN structures, not necessarily PT-ordinary; however it includes a transformation of general PNs to PT-ordinary PNs. A PN is said to be (partially) repetitive [14] if a marking μ_0 and an infinite firing sequence σ from μ_0 exist, such that every (some) transition occurs infinitely often in σ. A PN of incidence matrix[1] D is (partially) repetitive iff a vector $x \neq 0$ of positive (nonnegative) integers exists, such that $Dx \geq 0$ [14]. We prove the following theorem in [13].

Theorem 2.1: Consider a PN $\mathcal{N} = (P, T, F, W)$ which is not repetitive and let D be the incidence matrix. At least one transition exists such that for any given initial marking it cannot fire infinitely often. Let T_D be the set of all such transitions.

[1]In this paper the rows of the incidence matrix correspond to places and the columns to transitions.

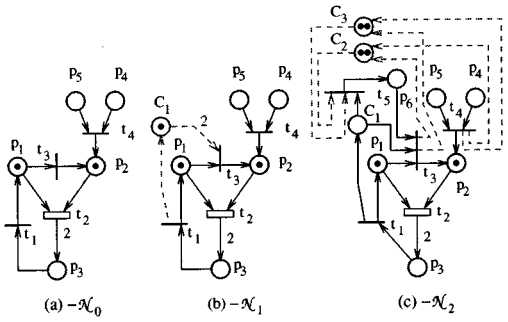

Fig. 1. Illustative example.

There is a nonnegative integer vector x such that $Dx \geq 0$, $x(i) \neq 0\ \forall t_i \in T \setminus T_D$ and $x(i) = 0\ \forall t_i \in T_D$.

Definition 2.1: Let $\mathcal{N} = (P, T, F, W)$ be a PN, D the incidence matrix, and $T_D \subseteq T$ the set defined in Theorem 2.1. $\mathcal{N}^A = (P^A, T^A, F^A, W^A)$ is an **active subnet** of $\mathcal{N}$ if[2] $P^A = T^A\bullet$, $F^A = F \cap \{(T^A \times P^A) \cup (P^A \times T^A)\}$, W^A is the restriction of W to F^A and T^A is the set of transitions with nonzero entry in some nonnegative vector x which satisfies $Dx \geq 0$. The maximal active subnet of $\mathcal{N}$ is the active subnet $\mathcal{N}^A$ such that $T^A = T \setminus T_D$.

The maximal active subnet of a PN can be computed using the algorithm given in the Appendix. A siphon is a set of places $S \neq \emptyset$ such that $\bullet S \subseteq S\bullet$. A siphon S is empty if it contains no tokens and controlled [9], [12] if

$$\sum_{p \in S} \mu(p) \geq 1 \tag{1}$$

is true for all markings μ reachable from the initial marking. Next we define a particular type of siphon.

Definition 2.2: Given an active subnet $\mathcal{N}^A$ of a PN $\mathcal{N}$, a siphon of $\mathcal{N}$ is said to be an **active siphon** with respect to $\mathcal{N}^A$ if it is, or includes, a siphon of $\mathcal{N}^A$. An active siphon is minimal if it does not include another active siphon with respect to the same active subnet.

As an example, the maximal active subnet $\mathcal{N}^A$ of the PN of Fig. 1(b) is given by $T^A = \{t_1, t_2, t_3\}$. The minimal active siphons w.r.t. $\mathcal{N}^A$ are $\{p_1, p_3\}$, $\{C_1, p_2, p_3, p_4\}$ and $\{C_1, p_2, p_3, p_5\}$.

Lemma 2.1: Let $\mathcal{N}^A = (P^A, T^A, F^A, W^A)$ be an active subnet of $\mathcal{N}$. Then $\bullet T^A \subseteq P^A$.

Proof: Let x be the vector defining T^A in Definition 2.1. Let σ be a sequence such that each transition t_i appears exactly $x(i)$ times in σ. There is an initial marking μ_0 enabling σ. Then $Dx \geq 0$ implies that the infinite sequence $\sigma_\infty = \sigma\sigma\sigma\ldots$ is enabled by μ_0. Since σ_∞ contains only transitions in T^A, the marking of the places $p \notin T^A\bullet$ is never increased while σ_∞ is fired. Furthermore, because the transitions of T^A appear infinitely often in σ_∞, $\bullet T^A \subseteq T^A\bullet$, q.e.d. ■

It is known that a deadlocked ordinary PN has an empty siphon [8]. We extend this result as follows.

Proposition 2.1: Let $\mathcal{N}^A$ be an arbitrary, nonempty, active subnet of a PT-ordinary PN $\mathcal{N}$. If μ is a deadlock marking of

[2]Given a set S, $\bullet S$ ($S\bullet$) denotes the preset (postset) of S evaluated in the total net, rather than in a subnet.

$\mathcal{N}$, then there is at least one empty minimal active siphon with respect to $\mathcal{N}^A$.

Proof: Since μ is a deadlock marking and $\mathcal{N} = (P, T, F, W)$ is PT-ordinary, $\forall t \in T\ \exists p \in {\bullet}t$: $\mu(p) = 0$. By Lemma 2.1, a marking μ restricted to the active subnet enables a transition t iff μ enables t in the total net. Therefore, because the total net $(\mathcal{N}, \mu)$ is in deadlock, the active subnet is deadlocked and so there is an empty minimal siphon s of the active subnet. Consider s in the total net. If s is a siphon of the total net, then s is also a minimal active siphon; therefore the net has a minimal active siphon which is empty. If s is not a siphon of the total net, ${\bullet}s \backslash T^A \neq \emptyset$. Let S be the set recursively constructed as follows: $S_0 = s$, $S_i = S_{i-1} \cup \{p \in {\bullet}({\bullet}S_{i-1} \setminus S_{i-1}{\bullet}) : \mu(p) = 0\}$, where μ is the (deadlock) marking of the net. In other words S is a completion of s with places with null marking such that S is a siphon. By construction S is an active siphon and is empty for the marking μ, q.e.d. ■

The significance of Proposition 2.1 is that it provides a way to do deadlock prevention, since deadlock is impossible when all active siphons with respect to a nonempty active subnet cannot become empty. Our usage of active subnets is as follows. In a PN $\mathcal{N}$, we may not be able to prevent reaching a marking such that the transitions in a set X are dead. The set X may contain more than just the transitions of the set T_D of Theorem 2.1 when the PN has uncontrollable and/or unobservable transitions, or when the PN markings are restricted to satisfy some given set of constraints (which beginning with Section III-A are called *initial constraints*). Then, we can still prevent deadlock in $\mathcal{N}$ only if there is a nonempty active subnet $\mathcal{N}^A$ which does not contain any of the transitions in X. Furthermore, out of all siphons, only the active siphons are to be controlled, by Proposition 2.1. When $X = \emptyset$ and $\mathcal{N}$ is repetitive, the maximal active subnet $\mathcal{N}^A$ equals $\mathcal{N}$ and so all siphons are active with respect to $\mathcal{N}^A$. We use active subnets in order to be able to effectively deal with the cases when $X \neq \emptyset$ or $\mathcal{N}$ is nonrepetitive. For instance, in Fig. 1(c), we know that we do not need to control the siphons $\{p_4\}$ and $\{p_5\}$, since they are not active siphons.

III. The Deadlock Prevention Approach

A. Introduction to the Method

Given a PN $\mathcal{N}_0$, the deadlock prevention procedure generates a sequence of PT-ordinary PNs, $\mathcal{N}_1, \mathcal{N}_2, \ldots \mathcal{N}_k$, increasingly improved with respect to deadlock prevention. $\mathcal{N}_1$ is $\mathcal{N}_0$ transformed into a PT-ordinary net. The other PNs are obtained as follows: at each iteration i the new minimal active siphons of $\mathcal{N}_i$ are *controlled* and then, if needed, the net is transformed to be PT-ordinary; the resulting PT-ordinary net is $\mathcal{N}_{i+1}$. The active siphons of each $\mathcal{N}_i$ are taken with respect to an active subnet $\mathcal{N}_i^A$ computed for every iteration i. To control a siphon, a linear marking inequality is enforced. Let $L_i\mu \geq b_i$ be the total set of constraints enforced in $\mathcal{N}_i$. Because $\mathcal{N}_k$ is the last PN in the sequence, it has no uncontrolled active siphons. Therefore $\mathcal{N}_k$ is deadlock free for all initial markings which satisfy $L_k\mu \geq b_k$. Finally, the constraints $L_k\mu \geq b_k$ can easily be translated to constraints in terms of the markings of $\mathcal{N}_0$, which define the supervisor for deadlock prevention in $\mathcal{N}_0$.

When the markings of the net are restricted due to additional specifications, a set of inequalities $L_I\mu \geq b_I$ describing how the markings are restricted can be passed as input to the procedure. They are called **initial constraints**. The usage of initial constraints $L_I\mu \geq b_I$ may result in less complex supervisors, may enhance convergence and guarantees that the procedure will not generate constraints requiring $L_I\mu \not\geq b_I$.

Uncontrollable and unobservable transitions, as well as initial constraints which are too restrictive, may cause the procedure to fail to control some siphons. When this happens, rather than leaving some active siphons uncontrolled, the procedure shrinks the active subnet such that those uncontrolled siphons are no longer active. To this end, the procedure places into an internal variable X the transitions in the postset of such siphons and then recomputes the active subnet to exclude the transitions in X.

B. Transforming Petri Nets to PT-Ordinary Petri Nets

The transformation we use is a modification of a similar operation in [6]. This modification allows us to reduce the number of siphons and to ensure that the siphons controlled in an iteration of our deadlock prevention procedure remain controlled after the transformation is applied. Let $\mathcal{N} = (P, T, F, W)$ be a PN. Transitions $t_j \in T$ such that $\exists p \in {\bullet}t_j$: $W(p, t_j) > 1$ are **split**. Given t_j, let $m = \max\{W(p, t_j)$: $p \in {\bullet}t_j\}$. When t_j is split, $m-1$ new transitions and $m-1$ new places are generated: $t_{j,1}, t_{j,2}, \ldots t_{j,m-1}$ and $p_{j,1}, p_{j,2}, \ldots p_{j,m-1}$. The connections are as follows, where the preset/postset operator is denoted by $\bullet$ for evaluations in $\mathcal{N}$ and by ${\bullet}'$ in the PN obtained by splitting t_j, which is denoted by $\mathcal{N}' = (P', T', F', W')$.

1) ${\bullet}'p_{j,i} = t_{j,i}$ and $t_{j,i}{\bullet}' = p_{j,i}$ for $i = 1 \ldots m-1$, $p_{j,i}{\bullet}' = t_{j,i-1}$ for $i = 2 \ldots m-1$ and $p_{j,1}{\bullet}' = t_j$.
2) ${\bullet}'t_{j,i} = \{p \in {\bullet}t_j : W(p, t_j) > i\} \cup Y$ for $i = 1 \ldots m-1$, where $Y = \emptyset$ for $i = m-1$ and $Y = \{p_{j,i+1}\}$ otherwise.
3) ${\bullet}'t_j = {\bullet}t_j \cup \{p_{j,1}\}$ and $t_j{\bullet}' = t_j{\bullet}$.
4) $\forall p \in {\bullet}'t_{j,i}$: $W'(p, t_{j,i}) = 1$ and $W'(t_{j,i}, p_{j,i}) = 1$, for $i = 1 \ldots m-1$.
5) $\forall p \in {\bullet}'t_j$: $W'(p, t_j) = 1$ and $\forall p \in t_j{\bullet}'$: $W'(t_j, p) = W(t_j, p)$.

Note that the connections of t_j in $\mathcal{N}'$ are the same as in $\mathcal{N}$, except for an additional transition arc and for the weights of the input arcs. Firing t_j in $\mathcal{N}$ corresponds to firing the sequence $t_{j,m-1} \ldots t_{j,1}, t_j$ in $\mathcal{N}'$. The transformation is illustrated in Fig. 2.

C. Enforcing Linear Marking Constraints

Linear constraints on the marking vector have the form $L\mu \geq b$, where L is matrix and b is vector. Enforcing such constraints is done according to the supervision based on place invariants in [1] and [2]. However, this requires admissible constraints. A constraint is admissible [1] if the supervisor enforcing the constraint does not inhibit an enabled uncontrollable transition and does not observe an unobservable transition. Then $L\mu \geq b$ are admissible if the following conditions of [1] are satisfied: $LD_{\text{uc}} \geq 0$ and $LD_{\text{uo}} = 0$, where the columns of D_{uc} and D_{uo} are the columns of the incidence matrix D that correspond to uncontrollable (D_{uc}) and unobservable (D_{uo}) transitions.

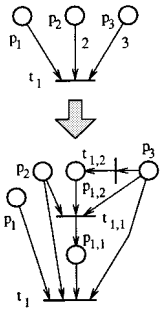

Fig. 2. The transformation to PT-ordinary PNs.

Controlling siphons involves enforcing (1). Enforcing (1) via the method of [1], [2] is equivalent to the approach used for siphon control in [4], [6], and [9]; note that the latter references consider controllable and observable transitions only. In the case of uncontrollable and unobservable transitions, in order to have a valid supervisor, the final constraints $L\mu \geq b$ generated by the procedure need to be admissible. Recall that the procedure adds constraints to the intermediary PN's $\mathcal{N}_i$ and when it terminates, the final set of constraints is written in terms of the target net $\mathcal{N}_0$. So we say that (1) is *admissible* if (1), when written in terms of $\mathcal{N}_0$, is admissible in $\mathcal{N}_0$. As we want all constraints to be admissible, when (1) is not admissible it is transformed to an admissible constraint

$$\sum_{p\in S} \alpha_p \mu(p) \geq 1 \tag{2}$$

such that $\alpha_p \in \mathbb{N}$ and at least two[3] coefficients α_p are nonzero. Enforcing (2) requires an additional place, which is called the **control place**. The control place C of a siphon S introduces the place invariant described by

$$\mu(C) = \sum_{p\in S} \alpha_p \mu(p) - 1. \tag{3}$$

Example: The PN of Fig. 1(a) has a single minimal active siphon with respect to the maximal active subnet: $S = \{p_1, p_3\}$. Assume that t_2 is unobservable. Then (1) is not admissible. However, (1) can be transformed to the following admissible constraint of the form (2):

$$2\mu(p_1) + \mu(p_3) \geq 1 \tag{4}$$

In Fig. 1(b), the control place C_1 enforces (4). □

In order to prove Theorem 5.1, we need to impose an additional requirement on (2). Let T_R be the set of all transitions created by the transformations to PT-ordinary nets during the iterations of the deadlock prevention procedure; for instance, in Fig. 2, the transitions created by splitting t are $t_{1,1}$ and $t_{1,2}$. We impose the requirement that (2) is such that $\forall t \in T_R$: $C \notin t\bullet$. We provide in [15] an algorithm to transform (1) to (2) such that all requirements are satisfied.

For all $t \in C\bullet$ such that $W(C,t) > 1$, t is split. The inequality (2) is still true after the split, but the place invariant is changed to include the markings of the new places resulting through the split, and (3) is changed accordingly. Consider an enforced inequality $l^T\mu \geq b$ or an invariant $l^T\mu = b$, where $l \in \mathbb{N}^{n\times 1}$, $b \in \mathbb{N}$, and n is the number of places. If a transition t_i is split, using the notations of Section III-B, the inequality or invariant is modified as follows:

$$\sum_p l_p \mu(p) \longrightarrow \sum_p l_p \mu(p) + \sum_{p:m_p>1} \left(l_p \sum_{j=1}^{m_p-1} j\mu(p_{i,m_p-j}) \right) \tag{5}$$

where $m_p = W(p,t_i)$ if $p \in \bullet t_i$ and else $m_p = 0$. ($W(p,t_i)$ is evaluated before splitting t_i.)

Example: In the example of Fig. 1, C_1 results with $W(C_1,t_3) = 2$. By splitting t_3, a new place p_6 and a new transition t_5 are generated. The place invariant (3) becomes

$$\mu(C_1) = 2\mu(p_1) + \mu(p_3) - \mu(p_6) - 1 \tag{6}$$

However, note that (4) is still valid, since (4) is implied by $2\mu(p_1) + \mu(p_3) - \mu(p_6) \geq 1$. □

In an intermediary PN $\mathcal{N}_i$, the marking of the control places μ_c can be expressed in terms of the marking of the other places μ_π by

$$\mu_c = L_i \mu_\pi - b_i. \tag{7}$$

The matrices L_i and b_i are recursively obtained as follows: if a control place C has been added in iteration i with regard to a siphon S, replace in (3) the markings of all control places $C' \in S$ added in the previous iterations with their expressions available from L_{i-1} and b_{i-1}. Thus the new form of (3) is $\mu(C) = l^T\mu_\pi - b$ and the new form of (2) is

$$l^T\mu_\pi \geq b \tag{8}$$

Example: In Fig. 1(c), $\mathcal{N}_2$ has two uncontrolled siphons: $S_1 = \{C_1, p_2, p_3, p_4\}$ and $S_2 = \{C_1, p_2, p_3, p_5\}$. For both $S = S_1$ and $S = S_2$, (1) is inadmissible. However, (1) for S_1 and S_2 can be transformed to the forms (2) as follows:

$$\mu(C_1) + 2\mu(p_2) + \mu(p_3) + \mu(p_4) \geq 1 \tag{9}$$
$$\mu(C_1) + 2\mu(p_2) + \mu(p_3) + \mu(p_5) \geq 1. \tag{10}$$

By enforcing (9) and (10), we obtain the control places C_2 and C_3. By substituting (6) into (9) and (10), we obtain

$$2\mu(p_1) + 2\mu(p_2) + 2\mu(p_3) + \mu(p_4) - \mu(p_6) \geq 2 \tag{11}$$
$$2\mu(p_1) + 2\mu(p_2) + 2\mu(p_3) + \mu(p_5) - \mu(p_6) \geq 2, \tag{12}$$

So L_3 and b_3 are

$$L_3 = \begin{bmatrix} 2 & 0 & 1 & 0 & 0 & -1 \\ 2 & 2 & 2 & 1 & 0 & -1 \\ 2 & 2 & 2 & 0 & 1 & -1 \end{bmatrix} \quad b_3 = \begin{bmatrix} 1 \\ 2 \\ 2 \end{bmatrix}. \tag{13}$$

□

Some siphons may not need enforcing (1) with a control place. When a siphon S has this property, the control place C which results by enforcing (1) satisfies $C\bullet \subseteq \bullet S$. (This identifies the case when S also is a *trap*.) Then we only need to ensure that the initial markings satisfy (1). So, after bringing (1) to a form $l^T\mu \geq b$ by replacing the control place markings with their expressions in (7), we add the constraint $l^T\mu \geq b$ to

[3]Except when S is a single place, allowing a single nonzero coefficient constraint makes the deadlock prevention procedure diverge.

a set of constraints $L_{0,i}\mu \geq b_{0,i}$ rather than $L_i\mu \geq b_i$, since $l^T\mu \geq b$ is only a constraint on the initial marking.

Example: In Fig. 1(c), after adding C_2 and C_3, there are four new active siphons: $\{p_4, p_6, C_2\}$, $\{p_5, p_6, C_2\}$, $\{p_4, p_6, C_3\}$ and $\{p_5, p_6, C_3\}$. They are controlled if initially marked. For instance, the constraints $\mu(p_4)+\mu(p_6)+\mu(C_2) \geq 1$ and $\mu(p_5)+\mu(p_6)+\mu(C_3) \geq 1$ are written as

$$2\mu(p_1)+2\mu(p_2)+2\mu(p_3)+2\mu(p_4) \geq 3 \quad (14)$$

$$2\mu(p_1)+2\mu(p_2)+2\mu(p_3)+2\mu(p_5) \geq 3 \quad (15)$$

and are included in $L_{0,4}\mu \geq b_{0,4}$. Note that no control places are added and so no new active siphons appear after ensuring that these four siphons are controlled. □

D. The Deadlock Prevention Procedure

Input: The target PN $\mathcal{N}_0 = (P_0, T_0, F_0, W_0)$ and optionally a set of initial constraints (L_I, b_I), by default empty.

Output: Two sets of constraints (L, b) and (L_0, b_0). (Deadlock is prevented for all initial markings μ_0 such that $L\mu_0 \geq b$ and $L_0\mu_0 \geq b_0$, when $(\mathcal{N}_0, \mu_0)$ is supervised according to $L\mu \geq b$, where the constraints $L\mu \geq b$ are by construction admissible.)

Procedure:

1) (L_0, b_0) is initialized to (L_I, b_I) and (L, b) to be empty. $\mathcal{N}_0$ is transformed into a PT-ordinary net; the new PN is $\mathcal{N}_1$ and (L_0, b_0) is updated accordingly [see (5)]. Let $i = 1$. If not previously defined, let $X = \emptyset$.
2) The largest active subnets of $\mathcal{N}_0$ and $\mathcal{N}_1$ which do not contain the transitions of X are computed.[4] Let them be $\mathcal{N}_0^A$ and $\mathcal{N}_1^A$. If $\mathcal{N}_0^A$ is empty, the procedure cannot generate a deadlock prevention supervisor and so it terminates.
3) **For $i \geq 1$ do** (the initial PN of the iteration i is denoted $\mathcal{N}_i = (P_i, T_i, F_i, W_i)$ and the active subnet $\mathcal{N}_i^A$.)
   a) If no uncontrolled minimal active siphon[5] is found, the next step is step 4). (The active siphons are taken with respect to the current active subnet $\mathcal{N}_i^A$. A siphon S is *uncontrolled* if (1) is not implied by the current $L\mu \geq b$ and $L_0\mu \geq b_0$)
   b) **For** all uncontrolled minimal active siphons S do
      i) Let χ be the constraint (1). Verify whether enforcing χ produces a control place C such that $C\bullet \subseteq \bullet S$. If so, S does not need control, C is not added to $\mathcal{N}_i$ and the next step is step 3.b.iv below.
      ii) If χ is an inadmissible constraint, χ is transformed[6] so that it is an admissible constraint of the form (2). If this is not possible, $X \rightarrow X \cup S\bullet$ and the *for* loop continues with the next active siphon.
      iii) The constraint χ is enforced using the invariant based supervision [1], [2].
      iv) Let $l^T\mu \geq c$ be the constraint χ written in the form (8) and let P_R be the set of all places generated by transition splits until the current iteration. Check whether the system $l^T\mu \geq c$, $L\mu \geq b$, $L_0\mu \geq b_0$, $\mu(p) = 0\ \forall p \in P_R$ is feasible. If the system is infeasible, $X \rightarrow X \cup S\bullet$. Else, $l^T\mu \geq c$ is added to (L_0, b_0) if the previous step was 3.b.i, or to (L, b) if the previous step was 3.b.ii.
   c) If the PN is no longer PT-ordinary, the transitions which do not comply with this requirement are *split* (Section III-B) and the matrix L is updated according to (5).
   d) The active subnet is updated as the largest active subnet which does not contain the transitions in X.
   e) Let T^A be the set of transitions of the active subnet. If the active subnet is empty ($T^A = \emptyset$), the procedure cannot generate a deadlock prevention supervisor and so it terminates. Else if an infeasibility occurred at a step 3.b.iv of the current iteration, $X \rightarrow T_0 \setminus T^A$ and the procedure is restarted at step 1 with this value of X.
   f) The final nets of the iteration i are denoted by $\mathcal{N}_{i+1}$ and $\mathcal{N}_{i+1}^A$. Let $i \rightarrow i+1$.
4) The constraints (L, b) and (L_0, b_0) are modified to be written only in terms of the marking of the target net $\mathcal{N}_0$. This is done by removing the columns of L and L_0 corresponding to places not in $\mathcal{N}_0$.
5) Redundant constraints of (L, b) and (L_0, b_0) are removed.[7]

IV. Example

Consider the target PN structure of Fig. 3(a), which we use to illustrate our procedure. The PN may be seen as the representation of a manufacturing system shown in Fig. 3(d), which we describe next. We have four work areas, $WA1 \ldots WA4$ and three machine areas $MA1$, $MA2$ and $MA3$. In $WA1$ two parts are assembled and this operation involves two machines from $MA1$ and one from $MA2$; upon completion, all three machines should be in $MA2$. Work in $WA2$ involves one part, one machine from $MA2$, and one from $MA1$; upon completion, both machines should be in $MA1$. Work in $WA3$ involves one part which may be of two different types and one machine from $MA3$; upon completion, the machine returns to $MA3$. Optionally, the operation in $WA3$ is continued with an additional operation in $WA4$; when this is the case, the machine of $MA3$ is released when the process in $WA4$ is completed. If no failure occurs in $WA4$, the machine returns to $MA3$. When a failure occurs, the machine no longer may be used in $MA3$, but it can still be used in $MA1$ or $MA2$ and is moved to $MA2$. The marking of the places p_3, p_6, and p_7 corresponds to available machines. The marking of the places p_5, p_1, p_4, and p_8 corresponds to the number of working processes in $WA1 \ldots WA4$.

[4]This involves linear programming and can be carried out with polynomial complexity; see the Appendix.

[5]Experience shows that the computation of the minimal siphons may be slow; verifying whether a siphon is uncontrolled usually involves solving an integer program.

[6]In [15] we implement the transformation using integer programming; however, integer programming may be avoided.

[7]This operation is optional and it usually involves integer programming.

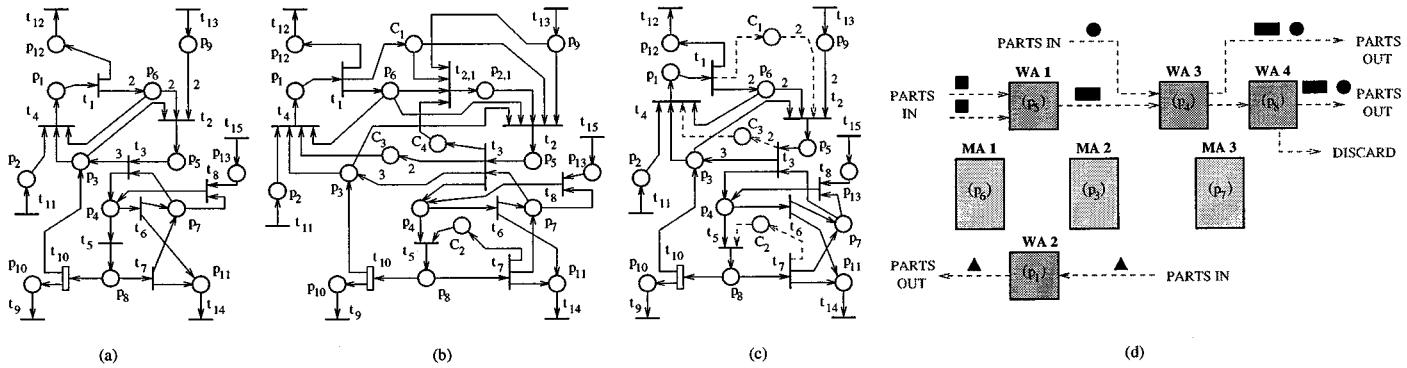

Fig. 3. (a) Target PN. (b) The PN after five iterations. (c) The supervised PN. (d) Manufacturing system.

The markings of p_2, p_9, p_{10}, p_{11}, p_{12}, and p_{13} represent the number of parts in buffer areas. The uncontrollable transition t_{10} models the failure in $WA4$.

In the first iteration, the PN structure $\mathcal{N}_1 = (P_1, T_1, F_1, W_1)$ is that of Fig. 3(b), but without the control places $C_1, \ldots C_4$ and their transition arcs. The place $p_{2,1}$ and the transition $t_{2,1}$ appear by splitting t_1. The maximal active subnet has the transitions in $T_1 \setminus \{t_9, t_{10}\}$. There are two minimal active siphons: $\{p_1, p_6\}$ and $\{p_4, p_7, p_8\}$. They are controlled with two new control places: C_1 and C_2, respectively, where the constraint of C_2 is transformed to (17), which is admissible.

In the second iteration, the maximal active subnet still has the transitions $T_1 \setminus \{t_9, t_{10}\}$ and the only uncontrolled minimal active siphon is $\{C_2, p_8\}$. There is no admissible constraint of the form (2) for the control of $\{C_2, p_8\}$. ($\mu(C_2) \geq 1$ is not of the form (2), as (2) requires at least two nonzero coefficients α_p.) Therefore X, the set of transitions which should not appear in the active subnets of the following iterations, is set to $X = \{t_5, t_7, t_{10}\}$.

In the third iteration and the remaining iterations, the active subnet has the set of transitions $T_1 \setminus \{t_5, t_7, t_9, t_{10}\}$. The only uncontrolled minimal active siphon is $S = \{C_2, p_8, p_3, p_5\}$. The control place which results is C_3, enforcing the constraint (1), which is admissible in this case.

In the fourth iteration, the only uncontrolled minimal active siphon is $\{p_1, C_1, p_5, C_3\}$, and so the control place C_4 is added.

In the fifth iteration, the only uncontrolled minimal active siphon is $S = \{C_4, p_{2,1}, p_5\}$. Since the control place which would control this siphon satisfies $C\bullet \subseteq \bullet S$, no control place is added and the constraint (1) is included in (L_0, b_0).

The procedure terminates at the sixth iteration, as there is no uncontrolled minimal active siphon left. The constraints after the step 4) of the procedure are

$$\mu_1 + \mu_6 \geq 1 \qquad (16)$$

$$\mu_4 + \mu_7 \geq 1 \qquad (17)$$

$$\mu_3 + \mu_4 + \mu_5 + \mu_7 + \mu_8 \geq 2 \qquad (18)$$

$$2\mu_1 + \mu_3 + \mu_4 + 2\mu_5 + \mu_6 + \mu_7 + \mu_8 \geq 4 \qquad (19)$$

$$2\mu_1 + \mu_3 + \mu_4 + 3\mu_5 + \mu_6 + \mu_7 + \mu_8 \geq 5 \qquad (20)$$

where $\mu_i = \mu(p_i)$. The inequalities (16)–(19) are included in $L\mu \geq b$ and correspond to $C_1 \ \ldots \ C_4$ in this order, while the inequality (20) is written as $L_0\mu \geq b_0$. The inequality (19) is redundant and so it can be omitted. The PN supervised for deadlock freedom is obtained by enforcing the constraints (L, b) on the target net [Fig. 3(c)].

V. Main Results

In this section, we prove that our deadlock prevention method produces supervisors which prevent deadlock. Then we prove that the supervisors are not restrictive. Finally, we show how initial constraints can help the procedure terminate.

Definition 5.1: A marking μ of an intermediary PN $\mathcal{N}_i$ is said to be **valid** if its restrictions to the control places (μ_c) and the rest of the places (μ_π) satisfy (7) and $\mu(p) \neq 0$ only if p is a place of $\mathcal{N}_0$ or a control place. The markings μ_i of $\mathcal{N}_i$ and μ_j of $\mathcal{N}_j$ are **equivalent** if both are valid and $\mu_i(p) = \mu_j(p)$ for all places p common to $\mathcal{N}_i$ and $\mathcal{N}_j$.

In order to prove the results of this section, we need to introduce some notations. When a transition t_i of $\mathcal{N}$ is split in $t_{i,1}, \ldots, t_{i,m-1}$, thus a new net $\mathcal{N}'$ resulting, firing the sequence $t_{i,m-1}, \ldots \ t_{i,1}, \ t_i$ has the same effect as firing t_i in $\mathcal{N}$. In our procedure, a transition t_i may be split in some iteration, then some $t_{i,k}$ (where $t_{i,k}$ resulted by splitting t_i) can be split in a subsequent iteration and so on. We denote by $\sigma_{0,j}(t)$ an arbitrary transition sequence of $\mathcal{N}_j$ such that: 1) $\sigma_{0,j}(t)$ enumerates the transitions (including t itself) in which t of $\mathcal{N}_0$ is successively split until (and including) the iteration $j-1$ and 2) markings μ of $\mathcal{N}_j$ exist such that $\mu(p) \neq 0$ only if p is a place of $\mathcal{N}_0$ or a control place and μ enables $\sigma_{0,j}(t)$. In this way, firing the sequence $\sigma_{0,j}(t)$ in $\mathcal{N}_j$ corresponds to firing t in $\mathcal{N}_0$. If t is not split, we let $\sigma_{0,j}(t) = t$. The notation $\sigma_{i,j}(t)$ for $i < j$ and t in $\mathcal{N}_i$ is similarly defined by taking $\mathcal{N}_i$ instead of $\mathcal{N}_0$. Also, if $\sigma = t_1 t_2 t_3 \ldots$, we let $\sigma_{i,j}(\sigma) = \sigma_{i,j}(t_1)\sigma_{i,j}(t_2)\sigma_{i,j}(t_3)\ldots$. For all $i \geq 0$, we use the notation $\mathcal{N}_i = (P_i, T_i, F_i, W_i)$.

Theorem 5.1: Assume that the procedure terminates at step 5). The target net $\mathcal{N}_0$ supervised by enforcing $L\mu \geq b$ is deadlock-free for all initial markings μ_0 such that $L\mu_0 \geq b$ and $L_0\mu_0 \geq b_0$.

Proof: Let $\mathcal{N}_k$ be the PN of the last iteration, μ_0 an arbitrary initial marking of $\mathcal{N}_0$ satisfying $L\mu_0 \geq b$ and $L_0\mu_0 \geq b_0$, $\mu_{0,k}$ be the equivalent initial marking in $\mathcal{N}_k$ and $\mathcal{C}$ the set of control places in $\mathcal{N}_k$. By construction, all active siphons of $(\mathcal{N}_k, \mu_{0,k})$ are controlled. Hence, in view of Proposition 2.1, $(\mathcal{N}_k, \mu_{0,k})$ is deadlock-free.

We prove by contradiction that $(\mathcal{N}_0, \mu_0)$ in closed loop with the supervisor defined by $L\mu \geq b$ cannot reach a marking μ such

that all possible firings in $\mathcal{N}_0$ lead either to deadlock markings or to markings which do not satisfy $L\mu \geq b$. Assume the contrary, that such a marking μ can be reached. Let μ_k be the equivalent marking of μ in $\mathcal{N}_k$. Because $(\mathcal{N}_k, \mu_k)$ is deadlock-free, μ_k enables an infinite transition sequence σ in $\mathcal{N}_k$. Let T_R be the set of transitions created by split transition operations. Enforcing (2) on a siphon S yields $C \notin T_R\bullet$ by the way we construct (2) [15]; we also prove in [15] that enforcing (1) yields $C \notin T_R\bullet$. Therefore, firing any $t \in T_R$ always reduces the marking of some places in $P_0 \cup \mathcal{C}$ and only firing $t \in T_0$ (note that $T_0 = T_k \setminus T_R$) may increase the marking of some places in $P_0 \cup \mathcal{C}$. Because the total marking of $P_0 \cup \mathcal{C}$ is finite, σ must include transitions $t \in T_0$. Let t_1 be the first transition in T_0 that appears in σ. Since all transitions of σ before t_1 are in T_R, firing them only decreases markings of $P_0 \cup \mathcal{C}$ and t_1 cannot fire unless all other transitions of $\sigma_{0,k}(t_1)$ fired before (as μ_k is valid), it follows that $\sigma_{0,k}(t_1)$ is enabled by μ_k. But this implies that t_1 also is enabled by μ in $\mathcal{N}_0$ supervised with $L\mu \geq b$, which is a contradiction. ∎

The assumptions of Theorem 5.1 are that the procedure terminates and that it terminates at step 5) rather than 2) or 3.e. Termination at 2) occurs when the structure of the PN does not allow deadlock prevention, where this corresponds to an empty active subnet of $\mathcal{N}_0$ (cf. Theorem 2.1). Termination at 3.e occurs when there are enough many siphon control failures, where siphon control failures are the instances in which uncontrollable and unobservable transitions prevent the transformation of the constraint χ to an admissible constraint at step 3.b.ii and the instances in which the system at the step 3.b.iv is infeasible (infeasibilities occur for restrictive enough initial constraints).

Given a set of transitions T, we say that a supervisor enforces **T-liveness** if all transitions $t \in T$ are live in the supervised PN. We include in the Appendix the proof of the next result, as it is more involved.

Theorem 5.2: Let T_0^A be the set of transitions of the maximal active subnet of the target PN. Assume that: 1) a T_0^A-liveness supervisor subject to the same initial constraints (if any initial constraints are given) exists; 2) the deadlock prevention procedure terminates; 3) no failure to transform a constraint to the admissible form (2) occurs at any step 3.b.ii; and 4) for all constraints transformed to (2) all α_p are nonzero. Then the supervisor generated by the deadlock prevention procedure is no more restrictive than the least restrictive supervisor which enforces T_0^A-liveness and is subject to the same initial constraints (if any).

Theorem 5.2 gives sufficient conditions for the supervisor provided by the procedure (let it be Ξ) to be at least as permissive as any supervisor Ξ_L which enforces all transitions of the maximal active subnet to be live in the target net. Note that this does not mean that our supervisor enforces that the transitions of the maximal active subnet are live. However, this means that: a) if Ξ_L enforces T_0^A-liveness for some initial marking μ_0, then Ξ is defined for μ_0, that is, it prevents deadlock for μ_0 (where T_0^A is the set of transitions of the maximal active subnet) and b) any firing sequence σ enabled by Ξ_L from such a μ_0 is also enabled by Ξ from μ_0. Note also that the assumptions 3) and 4) are always satisfied when the target PN has no uncontrollable and no unobservable transitions. In the following corollary, note that when liveness enforcing supervisors exist, the target PN is repetitive.

Corollary 5.1: Assume that a liveness enforcing supervisor (subject to the same initial constraints, if any) exists and the assumptions 2)-4) of Theorem 5.2 hold true. Then the deadlock prevention procedure provides a supervisor no more restrictive than the least restrictive supervisor which enforces liveness and is subject to the same initial constraints (if any).

Theorem 5.1 shows that the procedure is guaranteed to prevent (total) deadlock if it terminates normally (at step 5). Our experience shows that the procedure tends to enforce liveness (when this is possible). However it seems to be hard to characterize the PNs for which the procedure is guaranteed to enforce liveness upon termination at step 5. In particular, we have shown that deadlock prevention is equivalent to liveness enforcement when the incidence matrix D satisfies that for all vectors $x \geq 0$ if $Dx \geq 0$ then x has all entries nonzero [13]. Under the assumptions of Corollary 5.1, when a supervisor generated by the procedure enforces liveness, it is the least restrictive liveness enforcing supervisor. In particular, when no uncontrollable and unobservable transitions exist, a liveness enforcement supervisor (subject to the same initial constraints, if any are given) exists and the procedure terminates, it terminates at step 5. Furthermore, if the generated supervisor enforces liveness, by Corollary 5.1, it is the least restrictive liveness enforcing supervisor. Corollary 5.1 may be seen as the particularization of Theorem 5.2 for repetitive PNs.

To illustrate the application of our results, we first refer to the example of Section III-C, involving the PN of Fig. 1. When the procedure is applied, the remaining constraints after step 5 are (4) in (L, b) and (14) and (15) in (L_0, b_0). The supervised PN corresponds to Fig. 1(b). Theorem 5.2 applies, since (4), (9), and (10) are of the form (2), with nonzero coefficients α_p. So, by Theorem 5.1 deadlock is prevented and by Theorem 5.2 the supervisor is no more restrictive than the least restrictive $\{t_1, t_2, t_3\}$-liveness enforcing supervisor; it can easily be seen that the supervisor enforces $\{t_1, t_2, t_3\}$-liveness, so it is the least restrictive $\{t_1, t_2, t_3\}$-liveness enforcing supervisor. However, Theorem 5.2 does not apply to the example of Section IV, since the assumptions 3) and 4) are violated: the constraint (17) controlling $\{p_4, p_7, p_8\}$ has $\alpha_{p_8} = 0$ and the procedure cannot generate an admissible constraint when it attempts to control $\{C_2, p_8\}$.

The procedure does not have guaranteed termination; however, it can be helped to terminate by using initial constraints. A particular case is when we are only interested in a finite set of initial markings and the target PN is bounded. Then initial constraints can be chosen to define a bounded set including all markings reachable from the initial markings of interest. Then, if the procedure is started with these initial constraints, assuming that no transition splits occur during the iterations (which in practice is often the case), the procedure terminates. Termination occurs because each time a new constraint is added to (L, b) or (L_0, b_0) in the procedure, at least one new marking is forbidden and the number of markings which can be forbidden is finite due to the initial constraints. To summarize, given a target PN $\mathcal{N}$:

- Find a set of constraints $L_I\mu \geq b_I$ with bounded feasible set $\mathcal{F}$ such that for all initial markings μ_0 of interest for $\mathcal{N}$: $\mathcal{R}(\mathcal{N}, \mu_0) \subseteq \mathcal{F}$ (a possible approach to generate $L_I\mu \geq$

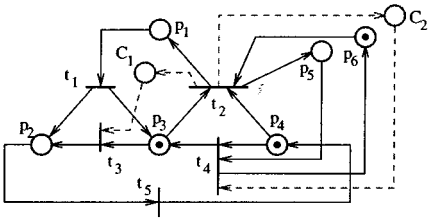

Fig. 4. Divergence example.

b_I is given in [15, Appendix]). Let $\mathcal{M}_I$ be the set of initial markings of interest.

- Apply the procedure on $\mathcal{N}$ with initial constraints (L_I, b_I).
- The resulting supervisor can be used for the initial markings $\mu_0 \in \mathcal{M}_I$ satisfying $L\mu_0 \geq b$ and $L_0\mu_0 \geq b_0$, where (L, b) and (L_0, b_0) are the two sets of constraints generated by the procedure.

Example: Consider the PN of Fig. 4. At the first iteration, the uncontrolled siphons are: $S_1 = \{p_1, p_3, p_5\}$, $S_2 = \{p_1, p_2, p_3, p_4\}$ and $S_3 = \{p_5, p_6\}$. The control place C_1 is added to control S_1; the inequality $\mu_1 + \mu_3 + \mu_5 \geq 1$ is added to (L, b), where $\mu_i = \mu(p_i)$. However, S_2 and S_3 do not need a control place (refer to step 3.b.i of the procedure), so $\mu_1 + \mu_2 + \mu_3 + \mu_4 \geq 1$ and $\mu_5 + \mu_6 \geq 1$ are added to (L_0, b_0). At the second iteration, there is a single uncontrolled siphon, $\{p_1, p_2, C_1, p_4\}$ and the control place C_2 results. At the third iteration, the uncontrolled siphons are $\{p_1, p_3, C_2\}$ and $\{C_2, p_6\}$. Note that C_2 has the same connections as p_5 and so the siphon $\{p_1, p_3, C_2\}$ corresponds to $S_1 = \{p_1, p_3, p_5\}$ and $\{C_2, p_6\}$ to $S_3 = \{p_5, p_6\}$. The procedure diverges. At each iteration, it adds a control place as follows: 1) at an iteration $n = 2k$, the control place C_n is added to control the siphon $\{p_1, p_2, C_{n-1}, p_4\}$ and 2) at an iteration $n = 2k+1$, the control place C_n is added to control the siphon $\{p_1, p_3, C_{n-1}\}$. Then it can be noticed that C_n, for $n = 1, 2, \ldots$ enforces

$$n\mu_1 + \left\lfloor \frac{n}{2} \right\rfloor \mu_2 + \left\lceil \frac{n}{2} \right\rceil \mu_3 + \left\lfloor \frac{n}{2} \right\rfloor \mu_4 + \mu_5 \geq n. \qquad (21)$$

It can be shown that the system of inequalities (21) for $n = 1$ and $n = n_1$ implies (21) for $n = n_1 - 1$, for all $n_1 \geq 3$. Furthermore, it can also easily be shown that the new markings forbidden by adding (21) at the iteration n are as follows: 1) for $n = 2k$, $\mu_1 = \mu_2 = \mu_4 = 0$, $\mu_3 = 1$ and $\mu_5 = k - 1$ and 2) for $n = 2k + 1$, $\mu_1 = \mu_3 = 0$, $\mu_2 + \mu_4 = 1$ and $\mu_5 = k$. Now assume that we start with the initial constraints $\mu_i \leq 4$ for all $i = 1 \ldots 6$; the usage of initial constraints assumes that for all our initial markings of interest, all reachable markings satisfy them. Then, at the iteration $n = 11$, the markings forbidden if (21) would be enforced are $\mu_1 = \mu_3 = 0$, $\mu_2 + \mu_4 = 1$, and $\mu_5 = 5$. However, according to the initial constraints, these markings can never be reached, so the siphon $\{p_1, p_3, C_{10}\}$ is controlled. Therefore, at the iteration $n = 11$, no control place is added and so the procedure terminates. After removing the redundant constraints, the procedure terminates with

$$L = \begin{bmatrix} 1 & 0 & 1 & 0 & 1 & 0 \\ 10 & 5 & 5 & 5 & 1 & 0 \end{bmatrix} \quad b = \begin{bmatrix} 1 \\ 10 \end{bmatrix} \qquad (22)$$

and (L_0, b_0) containing $\mu_5 + \mu_6 \geq 1$ and the initial constraints $\mu_i \leq 4$. □

VI. Conclusion

This paper has introduced a new deadlock prevention procedure. The performance of the procedure is formally proved. The procedure is effective for PN structures which may be generalized, with uncontrollable and unobservable transitions, nonrepetitive and unbounded. The initial marking does not need to be known; instead, the initial markings for which deadlock is prevented are characterized by a set of linear inequalities. Our approach to deadlock prevention has been implemented in software that performs automated synthesis of deadlock prevention supervisors and is available from the authors.

Appendix I
Additional Proofs

A. Proof of Theorem 5.2

Lemma 1.1: Assume that the requirements 3) and 4) of Theorem 5.2 are satisfied. Let S be an active siphon of $\mathcal{N}_{i+1}$, $i \geq 1$, which does not appear in $\mathcal{N}_i$. Let μ_{i+1} be a valid marking of $\mathcal{N}_{i+1}$ such that S is empty and μ_i be μ_{i+1} restricted to $\mathcal{N}_i$. Let t_s be a transition of $\mathcal{N}_i$ with the property that there is a transition $t \in S\bullet$ of $\mathcal{N}_{i+1}$ such that $t_s = t$ or t_s is split in $\mathcal{N}_{i+1}$ and t appears in $\sigma_{i,i+1}(t_s)$. If $\exists \mu, \mu_s \in \mathcal{R}(\mathcal{N}_i, \mu_i)$ such[8] that $\mu[t_s > \mu_s$, then $(\mathcal{N}_i, \mu_s)$ has at least one empty active siphon.

Proof: Let $\mathcal{C}$ be the set of control places added in the iteration i and P_R the set of places resulting through transition splits in the iteration i: $P_R = P_{i+1} \setminus (P_i \cup \mathcal{C})$. Let σ be the firing sequence that was used to reach μ: $\mu_i[\sigma > \mu$. Consider firing σ in $(\mathcal{N}_i, \mu_i)$ and $\sigma' = \sigma_{i,i+1}(\sigma)$ in $(\mathcal{N}_{i+1}, \mu_{i+1})$. The only reason for σ' not to be enabled in $\mathcal{N}_{i+1}$ by μ_{i+1} is that a control place prevents it.

If σ' is not enabled, $\sigma = \sigma_1 t_1 \sigma_2$, $\mu_i[\sigma_1 > \mu_1$, $\mu_{i+1}[\sigma_{i,i+1}(\sigma_1) > \mu_1'$, μ_1 enables t_1, but μ_1' does not enable $\sigma_{i,i+1}(t_1)$. This corresponds to the following: $\mathcal{N}_i$ has an active siphon S_1 that is controlled in $\mathcal{N}_{i+1}$ with C_1; when C_1 was added, $t_1 \in C_1\bullet$ and if $W(C_1, t_1) > 1$, t_1 was split in iteration i in $\sigma_{i,i+1}(t_1)$, or if $W(C_1, t_1) = 1$, $\sigma_{i,i+1}(t_1) = t_1$. So $t_1 \in S_1\bullet$ and since t_1 is not allowed by C_1 to fire from μ_1, it means that firing it would make S_1 empty. Since t_1 is fired in the sequence $\sigma = \sigma_1 t_1 \sigma_2$, after σ is fired, S_1 is an empty active siphon in $(\mathcal{N}_i, \mu_s)$.

If σ' is enabled by μ_{i+1}, let μ' be the marking reached: $\mu_{i+1}[\sigma' > \mu'$. Because σ' may contain only entire replacement sequences of split transitions and μ_{i+1} is a valid marking (which implies $\mu_{i+1}(p) = 0\ \forall p \in P_R$), $\mu'(p) = 0\ \forall p \in P_R$. Also, μ_{i+1} and μ_i are equivalent and $\sigma' = \sigma_{i,i+1}(\sigma)$, therefore $\mu(p) = \mu'(p)\ \forall p \in P_i$. Because S is a siphon, S empty for μ_{i+1} implies S empty for all reachable markings and thus for μ' too. There are two cases: 1) t_s is not split in $\mathcal{N}_{i+1}$ and 2) t_s is split.

1) If t_s is not split, $\bullet t_s \cap P_R = \emptyset$. Further on, μ enables t_s in $\mathcal{N}_i$ but μ' does not enable t_s in $\mathcal{N}_{i+1}$, so in $\mathcal{N}_{i+1}$, $\bullet t_s \cap \mathcal{C} \neq \emptyset$ and there is $C \in \bullet t_s \cap \mathcal{C}$ such that $\mu'(C) = 0$. Let S_C be the active siphon of $\mathcal{N}_i$ controlled by C. t_s was not split, so $W(C, t_s)$ was 1; t_s enabled by μ, $\mu'(C) = 0$ and $t_s \in C\bullet \Rightarrow t_s \in (S_C\bullet) \setminus (\bullet S_C)$. Since $S_C \subseteq P_i$ and

[8] $\mathcal{R}(\mathcal{N}_i, \mu_i)$ denotes the set of reachable markings of $(\mathcal{N}_i, \mu_i)$.

$\mu'(C) = 0$, $\sum_{p \in S_C} \mu(p) = 1$. Because t_s is enabled by μ, firing t_s empties S_C, so $(\mathcal{N}_i, \mu_s)$ has an empty active siphon.

2) If t_s was split, then t_s was connected to one or more of the control places C of $\mathcal{C}$, for only transitions connected to such places are split. (This is so because for all $i \geq 1$ $\mathcal{N}_i$ is PT-ordinary and hence only the new added control places can make the PN not PT-ordinary.) Let $\mathcal{C}_S$ be the set of control places added to $\bullet t_s$ in the iteration i. Since μ enables t_s and S is empty at μ', $t \in S\bullet$ implies $\bullet t_s \cap P_i \cap S = \emptyset$. Then, by recalling the split transition operation, $\exists C \in \mathcal{C}_S$ such that $C \in S$. Let S_C be the active siphon controlled by C. Since $C \in S$ and S is empty, $\sum_{p \in S_C} \mu(p) = 1$. Then firing t_s empties S_C, as $C \in \bullet t_s$ before the split of t_s. ($C \in \bullet t_s$ shows that firing t_s in $\mathcal{N}_i$ reduces the marking of S_C.) ■

Next is the proof of Theorem 5.2.

Proof: Let $\mathcal{S}$ be the set of supervisors subject to the initial constraints which enforce that all transitions appearing in the maximal active subnet are live in the target PN. Note that when we compare our procedure to any other supervisor we assume an initial marking for which that supervisor is defined: we do not require a supervisor in $\mathcal{S}$ to be defined for all initial markings for which the supervisor given by our procedure is defined.

First consider the case when there are no initial constraints. The proof is by contradiction. It shows that any marking forbidden by the deadlock prevention method also is forbidden by any supervisor in $\mathcal{S}$. (Recall that our procedure forbids markings which will produce an empty active siphon in an $\mathcal{N}_k$ for some k.)

Let $\mu^{(1)}$ be a marking of $\mathcal{N}_0$ and $\mu_k^{(1)}$ the equivalent marking in $\mathcal{N}_k$. Suppose that for the marking $\mu_k^{(1)}$ there is an empty active siphon S_k in $\mathcal{N}_k$. Because $\mu_k^{(1)}$ is valid, S_k is a new siphon which does not appear in $\mathcal{N}_{k-1}$; $\mu^{(1)}$ is forbidden by iteration k, which adds the constraint that S_k be controlled. Assume that $\mu^{(1)}$ is not forbidden by some supervisor enforcing in $\mathcal{N}_0$ that all transitions of the active subnet are live and so there is an infinite firing sequence σ enabled by $\mu^{(1)}$ such that every transition of $\mathcal{N}_0^A$ appears infinitely often in σ. Let $\mu_{k-1}^{(1)}$ be the marking of $\mathcal{N}_{k-1}$ equivalent to $\mu^{(1)}$. According to Lemma 1.1, there is a transition t'_{k-1} of $\mathcal{N}_{k-1}$ such that in any possible firing sequence enabled by $\mu_{k-1}^{(1)}$, after t'_{k-1} fires, there is an empty active siphon S_{k-1} of $\mathcal{N}_{k-1}$. Let $t_{k-1} \in T_0$ such that t'_{k-1} appears in $\sigma_{0,k-1}(t_{k-1})$. Let $\mu^{(2)}$ be the marking of $\mathcal{N}_0$ that appears while σ is fired, immediately after t_{k-1} fires for the first time. Also, let σ_1 be the subsequence of σ that was fired so far, that is $\mu^{(1)}[\sigma_1 > \mu^{(2)}$. Let $i \geq 0$ be the largest integer such that $\mu_i^{(2)}$ is a valid marking of $\mathcal{N}_i$ and the restriction of $\mu_i^{(2)}$ to $\mathcal{N}_0$ is $\mu^{(2)}$. By Lemma 1.1, $i \leq k-1$. Indeed, if σ_1 is allowed to fire in $\mathcal{N}_{k-1}$, there is an empty siphon S_{k-1} for the marking $\mu_{k-1}^{(2)}$, but there is no valid marking of $\mathcal{N}_k$ such that S_{k-1} is empty. Now, the fact that $\mu^{(2)}$ has an equivalent marking $\mu_i^{(2)}$ in $\mathcal{N}_i$ but not in $\mathcal{N}_{i+1}$ shows that there is an empty active siphon S_i in $\mathcal{N}_i$ and that S_i does not appear in $\mathcal{N}_{i-1}$. Further on, the same idea as before is used, that a transition t_{i-1} with the same property as t_{k-1} exists and, following this idea, an index $j \leq i-1$ is found such that for the marking $\mu^{(3)}$ of $\mathcal{N}_0$ there is an empty active siphon in $\mathcal{N}_j$. This procedure is repeated and finally two cases may appear (Lemma 1.1 applies for $i > 0$ only) after the first n transitions of σ are fired, where n is a finite number. Let σ_p denote the sequence that enumerates the first n transitions of σ and let $\mu^{(p)}$ be the marking reached by firing σ_p (that is, $\mu^{(1)}[\sigma_p > \mu^{(p)}$) and $\mu_1^{(p)}$ the valid marking of $\mathcal{N}_1$ which restricted to $\mathcal{N}_0$ is $\mu^{(p)}$. Then: (a) there is an empty active siphon in $(\mathcal{N}_0, \mu^{(p)})$ or (b) there is an empty active siphon in $(\mathcal{N}_1, \mu_1^{(p)})$. Case (a) contradicts the fact that every transition of $\mathcal{N}_0^A$ appears infinitely often in σ and $\mu^{(1)}$ enables σ, since after n firings none of the transitions in the postset of the empty siphon may fire again. Case (b) leads to the same type of contradiction, because the sequence $\sigma' = \sigma_{0,1}(\sigma)$ is enabled by $\mu_1^{(1)}$, where $\mu_1^{(1)}$ is the equivalent marking of $\mu^{(1)}$ in $\mathcal{N}_1$ and by construction every transition of $\mathcal{N}_1^A$ appears infinitely often in σ'.

The case when there are initial constraints is similar to the case when there are no such constraints if the procedure is never in the situation that a constraint at step 3.b.ii of the procedure is infeasible. This is always the case, as the assumption 1) of the theorem implies that $\mathcal{S}$ is nonempty. Indeed, if infeasibilities at some steps 3.b.ii were possible, consider the first occurrence: there is an active siphon S which must be empty for all valid markings, in order not to have a conflict with the initial constraints. Then, by the first part of the proof, there are no supervisors in $\mathcal{S}$. ■

APPENDIX II
COMPUTATION OF THE ACTIVE SUBNETS

The following algorithm computes the maximal active subnet which does not contain the transitions in a set X.

Input: The PN $\mathcal{N} = (P, T, F, W)$, its incidence matrix D and the set X.

Output: The active subnet $\mathcal{N}^A = (P^A, T^A, F^A, W^A)$.

Let $M = T \setminus X$ and $x_s = \mathbf{0}_{|T| \times 1}$
While $M \neq \emptyset$ **do**
1. Check whether the system of $Dx \geq 0$, $x \geq 0$, $x(i) = 0$ $\forall t_i \in X$, $\sum_{t_i \in M} x(i) \geq 1$ and $x \in \mathbb{R}^{|T|}$ is feasible.
2. **If** feasible **then** let x^* be a solution; $M = M \setminus \|x^*\|$ and[9] $x_s = x^* + x_s$. **Else** $M = \emptyset$.
End while

Let $M = T \setminus X$ and $x_s = \mathbf{0}_{|T| \times 1}$
While $M \neq \emptyset$ **do**
1. Check whether the system of $Dx \geq 0$, $x \geq 0$, $x(i) = 0$ $\forall t_i \in X$, $\sum_{t_i \in M} x(i) \geq 1$ and $x \in \mathbb{R}^{|T|}$ is feasible.
2. **If** feasible **then** let x^* be a solution; $M = M \setminus \|x^*\|$ and[10] $x_s = x^* + x_s$. **Else** $M = \emptyset$.
End while

[9] $\|x\|$ denotes the set of transitions t_i such that $x(i) \neq 0$.

[10] $\|x\|$ denotes the set of transitions t_i such that $x(i) \neq 0$.

The active subnet is $\mathcal{N}^A = (P^A, T^A, F^A, W^A)$, $T^A = ||x_s||$, $P^A = T^A\bullet$, $F^A = F \cap \{(T^A \times P^A) \cup (P^A \times T^A)\}$ and W^A is the restriction of W to F^A.

REFERENCES

[1] J. O. Moody and P. J. Antsaklis, *Supervisory Control of Discrete Event Systems Using PN's*. Norwell, MA: Kluwer, 1998.

[2] E. Yamalidou, J. O. Moody, P. J. Antsaklis, and M. D. Lemmon, "Feedback control of PN's based on place invariants," *Automatica*, vol. 32, no. 1, pp. 15–28, Jan. 1996.

[3] A. Giua, F. DiCesare, and M. Silva, "Generalized mutual exclusion constraints on nets with uncontrollable transitions," in *Proc. IEEE Conf. Syst. Man, Cybern*, 1992, pp. 974–979.

[4] J. Ezpeleta, J. M. Colom, and J. Martinez, "A PN based deadlock prevention policy for flexible manufacturing systems," *IEEE Trans. Robot. Automat.*, vol. 11, pp. 173–184, Apr. 1995.

[5] K. Xing, B. Hu, and H. Chen, "Deadlock avoidance policy for PN modeling of flexible manufacturing systems with shared resources," *IEEE Trans. Automat. Contr.*, vol. 41, pp. 289–295, Apr. 1996.

[6] K. Lautenbach and H. Ridder, "The linear algebra of deadlock avoidance—A PN approach," Institute for Computer Science, Univ. Koblenz, 1996.

[7] K. Barkaoui, A. Chaoui, and B. Zouari, "Supervisory control of discrete event systems using strucutre theory of PN's," in *Proc. IEEE Conf. Syst., Man, Cybern.*, 1997, pp. 3750–3755.

[8] W. Reisig and N. Petri, *EATCS Monographs on Theoretical Computer Science*. Berlin, Germany: Springer-Verlag, 1985, vol. 4, Petri Nets.

[9] K. Barkaoui and I. Abdallah, "Deadlock avoidance in FMSS based on structural theory of PN's," in *IEEE Symp. Emerging Technologies and Factory Automation*, 1995.

[10] ——, "A deadlock prevention method for a class of fms," in *Proc. IEEE Conf. Syst., Man, Cybern.*, 1995, pp. 4119–4124.

[11] K. Barkaoui and L. Petrucci, "Structural analysis of workflow nets with shared resources," in *Proc. First Int. Workshop on Workflow and PN's*, 1998.

[12] K. Barkaoui and J. F. Pradat-Peyre, "On liveness and controlled siphons in PN's," in *Application and Theory of PN's 1996*. Berlin, Germany: Springer-Verlag, 1996, vol. 1091, pp. 57–72.

[13] M. V. Iordache and P. J. Antsaklis, "Generalized conditions for liveness enforcement and deadlock prevention in PN's," in *Application and Theory of PN's 2001*. Berlin, Germany: Springer-Verlag, 2001, vol. 2075, pp. 184–203.

[14] T. Murata, "PNs: Properties analysis and applications," *Proc. IEEE*, pp. 541–580, Apr. 1989.

[15] M. V. Iordache, J. O. Moody, and P. J. Antsaklis, "Automated synthesis of deadlock preventions supervisors using PN's," Univ. Notre Dame, 2000.

Marian V. Iordache received the undergraduate degree from Politechnica University of Bucharest, Romania, in 1996 and the M.S. degree in electrical engineering from University of Notre Dame, Notre Dame, IN, in 1999. He is currently working toward the Ph.D. degree at the University of Notre Dame.

He was a recipient of the Center for Applied Mathematics fellowship from the University of Notre Dame. His technical research interests include Petri nets, discrete event systems, supervisory control, and hybrid systems.

Mr. Iordache is a member of Eta Kappa Nu.

John O. Moody (S'96–M'98) received the B.S., M.S., and Ph.D. degrees in electrical engineering from the University of Notre Dame, Notre Dame, IN.

He is an Advisory Engineer and Research Scientist for Lockheed Martin Federal Systems, Owego, NY. He is co-author of the book *Supervisory Control of Discrete Event Systems Using Petri Nets* (Norwell, MA: Kluwer, 1998, with P. Antsaklis). His research interests include discrete event and hybrid control systems, neural network construction and architecture, parallel computing, and autonomous, intelligent control systems.

Dr. Moody was awarded an Arthur J. Schmitt fellowship as well as the Center for Applied Mathematics fellowship from the University of Notre Dame. He is a member of Eta Kappa Nu and Tau Beta Pi.

Panos J. Antsaklis (S'74–M'76–SM'86–F'91) is Professor of Electrical Engineering and Director of the Center for Applied Mathematics at the University of Notre Dame, Notre Dame, IN. His work includes analysis of behavior and design of control strategies for complex autonomous, intelligent systems. His recent research focuses on networked embedded systems and addresses problems in the interdisciplinary research area of control, computing and communication networks and on hybrid and discrete event dynamical systems. He has authored a number of publications in journals, conference proceedings, and books and he has edited four books on Intelligent Autonomous Control and on Hybrid Systems. In addition, he has co-authored the research monograph *Supervisory Control of Discrete Event Systems Using Petri Nets* (Norwell, MA: Kluwer, 1998, with J. Moody) and the graduate textbook *Linear Systems* (New York: McGraw-Hill, 1997, with A. N. Michel). He serves on the editorial boards of several journals and he has been Guest Editor of several special issues.

Prof. Antsaklis has served as program chair and general chair of major systems and control conferences, and he was the 1997 President of the IEEE Control Systems Society (CSS). He has served as Guest Editor of the "Special Issue on Hybrid Systems" of the PROCEEDINGS OF THE IEEE, October 2001. He is a Distinguished Lecturer of the IEEE Control Systems Society, a recipient of the IEEE Distinguished Member Award of the Control Systems Society, and an IEEE Third Millennium Medal recipient.