Deadlock Avoidance in Petri Nets with Uncontrollable Transitions

Technical Report of the ISIS Group
at the University of Notre Dame
ISIS-97-016
October, 1997

John O. Moody and Panos J. Antsaklis
Department of Electrical Engineering
University of Notre Dame
Notre Dame, IN 46556

Interdisciplinary Studies of Intelligent Systems

Deadlock Avoidance in Petri Nets with Uncontrollable Transitions[1]

Abstract

Recent results in the literature have provided efficient control synthesis techniques for the problem of deadlock avoidance in Petri nets. These results are shown to fit within an established framework for the enforcement of linear constraints on the marking behavior of a net. Framing the problem in this way allows uncontrollable transitions to be included in the plant model when deadlock avoidance is performed. This method for constructing deadlock avoiding supervisors in the face of uncontrollable transitions is described and illustrated with an example.

1 Introduction

Deadlock avoidance is an important and difficult problem in the area of supervisory control. When the discrete event system (DES) is modeled by a Petri net (PN) [15–17], the net is said to be *deadlocked* if no transition in the net is able to fire. A net is called *live* if every transition can be, eventually, fired again and again. A deadlock-free net may not necessarily be completely live.

An efficient supervisory control technique for deadlock avoidance in Petri nets has been proposed in [4]. This method will guarantee deadlock-freedom, and also liveness for a large, useful class of Petri nets. The procedure does not account for possibly uncontrollable transitions within the plant, but uncontrollable events are a standard feature in most supervisory control frameworks.

Techniques for enforcing general linear constraints on Petri nets with uncontrollable transitions do exist [8, 9, 11–13]. These controllers enforce linear inequalities on the reachable markings of the plant while avoiding the inhibition of uncontrollable transitions. Unfortunately these controllers have not, in the past, accounted for the deadlock problem. In fact, the supervisors generated by these techniques may actually be the cause of plant deadlocks!

In this paper, the results for deadlock avoidance are placed within the framework for constraint enforcement in the face of uncontrollable transitions. The combined technique expands the applicability of both control procedures, adding the ability to handle uncontrollable transitions to one, and the avoidance of deadlock to the other.

Section 2 summarizes the Petri net concepts of traps and siphons, which are important to the understanding of the deadlock avoidance procedure discussed in the section 3. Section 4 then shows how this procedure can be placed within an established framework for the

[1]This research was partially funded by the National Science Foundation. Grant ECS95–31485.

supervisory control of Petri nets with uncontrollable transitions. The technique is illustrated in section 5 using an unreliable machine serviced by automated guided vehicles. Concluding remarks appear in section 6.

2 Petri Net Siphons and Traps

Traps and siphons[2] (see [5, 15, 17]) are sets of Petri net places. *Once the set of places in a trap become marked, the trap will always be marked for all future reachable markings. Similarly, once the marking of a siphon becomes empty, the siphon will remain empty.*

Traps and siphons are defined by the nature of the input and output transitions into a given set of places. Let $\bullet p$ refer to the set of input transitions into the place p, and let $p\bullet$ refer to the set of output transitions from the place p. The "bullet" notation can also be used with sets of places. If S is a set of places, then $\bullet S$ and $S\bullet$ refer to the set of input and output arcs into the entire set S.

Definition 1. A set of places S is a **siphon** iff

$$S\bullet \subseteq \bullet S$$

S is a **minimal siphon** iff there does not exist another siphon P such that $P \subset S$.

The definition of a trap is similar.

Definition 2. A set of places S is a **trap** iff

$$\bullet S \subseteq S\bullet$$

S is a **minimal siphon** iff there does not exist another siphon P such that $P \subset S$.

A place invariant vector with nonnegative elements indicates a set of places that is both a trap and siphon. The converse does not hold true. For example, consider a net with no source or sink transitions, i.e., all transitions have both input and output places. By definition, the set of all the net's places is both a trap and a siphon, however such a net may not be covered by a place invariant.

[2]Siphons are sometimes called "deadlocks."

3 Deadlock Avoidance

3.1 Structural Conditions for Liveness

Siphons are of particular interest in the area of deadlock avoidance; once a siphon becomes emptied of tokens, it will forever remain empty and all of the transitions that receive input arcs from these places will be dead. Barkaoui and Abdallah [4] have introduced the notion of a controlled siphon.

Definition 3. For a Petri net with initial marking μ_0, a **controlled siphon** is a siphon that remains marked for all markings reachable from μ_0.

A controlled siphon may be either trap-controlled or invariant-controlled. A *trap-controlled siphon* contains a trap that is initially marked, thus preventing the siphon from ever losing all of its tokens. An *invariant-controlled siphon's* marking is guaranteed by the presence of a place invariant. If the constant weighted sum of markings indicated by a place invariant insures that a siphon will never lose all of its tokens, then that siphon is invariant-controlled. The exact conditions that such a place invariant must meet appear in [4]. An example of an invariant-controlled siphon appears in the example below.

Example. The Petri net of Figure 1 contains two siphons, $S_1 = \{p_2, p_3\}$ and $S_2 = \{p_2, p_3, p_4\}$, where S_1 is the only minimal siphon since $S_1 \subset S_2$. The net contains no traps, so clearly the minimal siphon is not trap-controlled, however an analysis of the net's behavior reveals that this siphon will never be emptied.

The net contains a single place invariant:

$$\mu_2 + \mu_3 - \mu_4 = 1$$

where μ_i is the marking of place p_i. Thus $\mu_2 + \mu_3 \geq 1$ is always true and the siphon is invariant-controlled.

The following propositions relate controlled siphons to deadlock freedom and liveness. The results are well known in the literature, see [5, 15, 17], however Barkaoui *et al.* [3, 4] are responsible for the extension of the results to include the idea of invariant-controlled siphons as well as trap-controlled.

Proposition 4. **Deadlock condition.** A deadlocked Petri net contains at least one empty siphon.

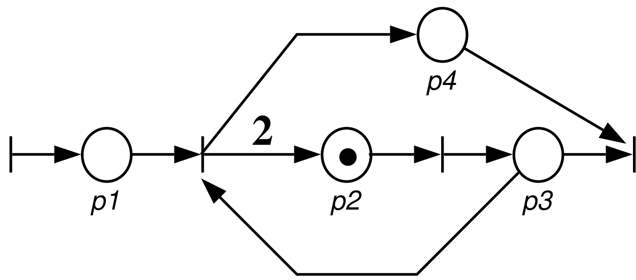

Figure 1: A Petri net with a controlled siphon but no trap.

Proposition 5. **Deadlock-freedom.** A Petri net is *deadlock-free* if every siphon in the net is a controlled siphon.

Proposition 6. **Commoner's Theorem.** An extended free choice (EFC) Petri net is *live* if and only if every siphon in the net is trap-controlled. (Invariant-controlled siphons are not required to insure the liveness of EFC nets.)

Proposition 7. **Liveness for AC nets.** An asymmetric choice (AC) Petri net is live if and only if every siphon in the net is a controlled siphon.

Nonminimal siphons always contain at least one minimal siphon, so it is only necessary to examine a net's minimal siphons when applying the propositions above.

The Petri net of Figure 1 is live. This follows from Proposition 7 since it is an AC net and its single minimal siphon is invariant-controlled.

There are other Petri net classes for which the condition that all siphons are controlled is sufficient for demonstrating that the Petri net is live. These include linear manufacturing lines [10], and production Petri nets [2]. The presence of controlled siphons is then sufficient to insure liveness for a wide variety of Petri nets, and will, at the very least, insure freedom from complete deadlock for Petri nets outside this class.

3.2 Deadlock Avoidance through Supervisory Control

A supervisory control technique is introduced in [4] for handling the problem when not all of the siphons in a given Petri net are controlled. The method involves adding a place for each uncontrolled siphon in the net such that they become controlled. These controller places act to restrict behaviors in the original plant that would lead to deadlock, thus they play the part of a supervisory controller: allowing the plant's state to evolve unrestricted except to prevent transition firings that lead to "forbidden states." An outline of this technique is described below.

Given a conservative, well-marked Petri net with uncontrolled siphons, for each uncontrolled siphon S, create a control place c such that

$$\begin{aligned} c\bullet &= \{t \in S\bullet : |\bullet t \cap S| > |t \bullet \cap S|\} \\ \bullet c &= \{t \in \bullet S : |t \bullet \cap S| > |\bullet t \cap S|\} \end{aligned} \tag{1}$$

where the notation $|x|$ refers to the number of elements in the set x, and the weights of the arc transitions are given by the differences $|\bullet t \cap S| - |t \bullet \cap S|$ and $|t \bullet \cap S| - |\bullet t \cap S|$ for the controller place's output and input arcs respectively. The initial marking of the control place, μ_{c_0}, is given by

$$\mu_{c_0} = \sum_{p_i \in S} \mu_{i_0} - 1 \tag{2}$$

where μ_{i_0} is the initial marking of place p_i in the plant.

Each control place insures that its siphon will never be emptied of all of its tokens. An analysis of the synthesis technique, (1) and (2), shows that this is done by creating place invariants in the controlled Petri net. For each control place c, associated with siphon S, the following place invariant is established in the controlled Petri net.

$$\sum_{p_i \in S} \mu_i - \mu_c = 1 \tag{3}$$

Thus the synthesis technique causes formerly uncontrolled siphons in the plant net to become invariant-controlled siphons in the controlled net.

Example. The free choice Petri net of Figure 2 is conservative: it contains a covering place invariant $x = [1\ 1\ 1]^T$. The three places form a trap, and this trap is minimal. The net contains three siphons:

$$\begin{aligned} S_1 &= \{p_1, p_2\} \\ S_2 &= \{p_1, p_3\} \\ S_3 &= \{p_1, p_2, p_3\} \end{aligned}$$

Place p_1 is marked with two tokens, and it is involved in all three of the net's siphons, thus the net is well-marked.

Siphon S_3 is both trap and invariant-controlled, however it is not minimal. Neither minimal siphon, S_1 and S_2, is controlled. It is easy to see that if the two tokens in p_1 were to both transfer to p_2 or p_3, then deadlock would result.

A deadlock-avoiding controller is constructed according to (1) and (2). Control places c_1 and c_2 are associated with siphons S_1 and S_2 respectively. The controlled system, shown in Figure 3, is live. Note that the resulting net is no longer a free choice net, though it is asymmetric choice.

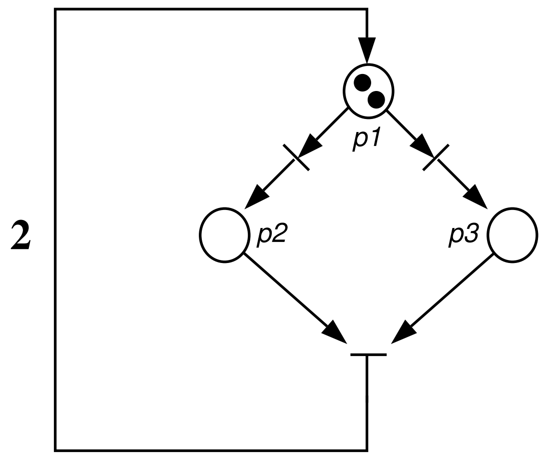

Figure 2: A free choice Petri net with uncontrolled siphons.

Controlling all of the formerly uncontrolled siphons in a net is sufficient for insuring liveness for a wide variety of Petri nets, as noted above. Liveness is not guaranteed for nets outside this class, though they will still be deadlock-free (weakly-live). For these nets, it is proved that if the net is not live (at least one transition is "dead"), then the marking of at least one control place must be zero. Based on this fact, an algorithm is presented in [4] that determines which transitions should fire in order to cause the filling of control places in the fewest number of steps. Because this algorithm actively seeks transitions that should fire, rather than the simple enabling and disabling of transitions of a supervisory controller, it is not discussed here.

4 Handling Uncontrollable Transitions

The deadlock avoidance technique of the previous section is extended here to include nets that contain *uncontrollable transitions*. The firing of an uncontrollable transition is restricted only by the state of the plant, a supervisor can not disable an otherwise enabled uncontrollable transition. Controllers constructed using the technique of the previous section may not be valid if a control place draws an arc to an uncontrollable transition, since this is the mechanism by which the controller inhibits transition firing. This problem is handled by first noting the similarity of the deadlock avoiding controllers of [4] to similar PN supervisory controllers. Methods for handling uncontrollable transitions for these techniques will then be applied to the deadlock avoidance problem.

The controllers in section 3.2 enforce the invariant equation (3). This is equivalent to enforcing the following inequality

$$\sum_{p_i \in S} \mu_i \geq 1 \tag{4}$$

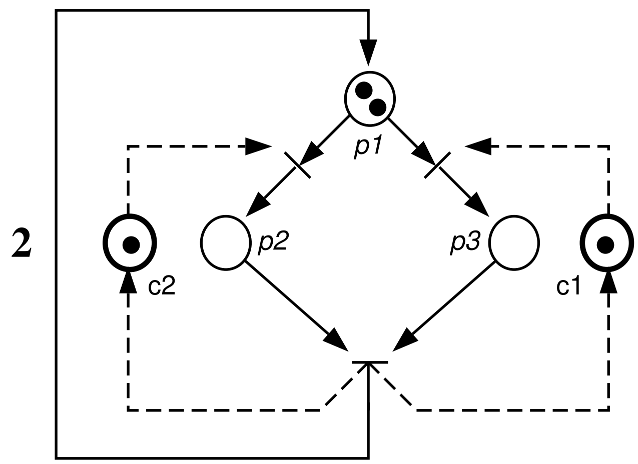

Figure 3: The controller, shown with bold places and dashed arcs, insures the liveness of the FC net of Figure 2.

where μ_c plays the part of a (nonnegative) excess variable. The inequality is intuitively appealing, simply stating that the siphon should never be emptied of tokens. A technique for creating Petri net supervisors for enforcing general linear inequalities on the markings of Petri nets have been in developed in [7, 14, 18]. Furthermore, methods for modifying the inequality such that the resulting controller accounts for uncontrollable transitions have been presented in [8, 9, 11, 13]. The methods of [11, 13, 14, 18] (see also [12], for a summary) are discussed below.

The following constraint is to be imposed on the plant state, $\mu_p \in \mathbb{Z}^n, \mu_p \geq 0$

$$l^T \mu_p \leq b \tag{5}$$

where $l \in \mathbb{Z}^n, b \in \mathbb{Z}$, and $\mathbb{Z}$ is the set of integers. For inequality (4),

$$\begin{aligned} l_i &= \begin{cases} -1 & \text{if } p_i \in S \\ 0 & \text{else} \end{cases} \\ b &= -1 \end{aligned} \tag{6}$$

for $i = 1 \ldots n$, where l_i is the i^{th} element of l. The definition of (6) shows that both sides of inequality (4) were multiplied by -1 to achieve the "less-than-or-equal-to" form of inequality (5).

If all of the transitions within the plant Petri net are controllable and observable, then it has been shown that (5) can be enforced with maximal permissivity by a Petri net controller that produces a place invariant on the closed loop plant-controller system.

The incidence matrix of the closed loop system, D, and its marking, μ, are given by

$$D = \begin{bmatrix} D_p \\ D_c \end{bmatrix} \quad \mu = \begin{bmatrix} \mu_p \\ \mu_c \end{bmatrix} \tag{7}$$

where $D_p \in \mathbb{Z}^{n\times m}$ is the incidence matrix of the plant, $D_c \in \mathbb{Z}^{1\times m}$ is the incidence matrix of the controller, and $\mu_c \in \mathbb{Z}, \mu_c \geq 0$ is its marking. The controller and its initial marking μ_{c_0} is calculated using

$$D_c = -l^T D_p \tag{8}$$

$$\mu_{c_0} = b - l^T \mu_{p_0} \tag{9}$$

where $\mu_{p_0} \in \mathbb{Z}^n$ is the initial marking of the plant.

Some sets of constraints can not be enforced and thus appropriate controllers do not exist. It is possible to enforce the set of constraints (5) iff

$$b - L\mu_{p_0} \geq 0 \tag{10}$$

Inspection of the constraint definition given by (6) combined with the controller construction of (8) and (9) shows that this controller is identical to the controller constructed using (1) and (2) of section 3.2. Both methods assume that the plant transitions to which the controller arcs are directed are controllable. However, it is possible to transform the original constraint into a new constraint that will result in a controller that does not interfere with uncontrollable transitions while still insuring that the original inequality is maintained.

Negative numbers in the controller incidence matrix D_c correspond to arcs from the control place to the plant transitions. These are the arcs that must be restricted if some transitions are uncontrollable. If the columns of D_c associated with uncontrollable transitions all have nonnegative values, then the controller will meet the requirement imposed by uncontrollability.

Let D_{uc} be an incidence matrix composed of the columns of D_p that correspond to uncontrollable transitions. An examination of equation (8) reveals that the controller will not interfere with the uncontrollable transitions if

$$l^T D_{\mathrm{uc}} \leq 0 \tag{11}$$

where the inequality is read with respect to each element in the vector $l^T D_{\mathrm{uc}}$.

If the inequality is not met, then it is desirable to obtain a new inequality $l'^T \mu_p \leq b'$ such that

1. $l'^T \mu_p \leq b' \rightarrow l^T \mu_p \leq b$
2. $l'^T D_{\mathrm{uc}} \leq 0$

That is, we wish to transform the inequality into a form such that enforcement of the new inequality will also imply the enforcement of the original, while obeying the constraint imposed by the plant's uncontrollable transitions. Analytical and computational techniques for obtaining transformations with these two properties appear in [11–13]. One method of computing the transformation involves performing positive row operations on the matrix $\begin{bmatrix} D_{\mathrm{uc}} \\ l^T D_{\mathrm{uc}} \end{bmatrix}$ in order to eliminate the positive numbers in the $l^T D_{\mathrm{uc}}$ portion of the matrix. This technique is illustrated in the example of the following section.

5 Example – The Unreliable Machine

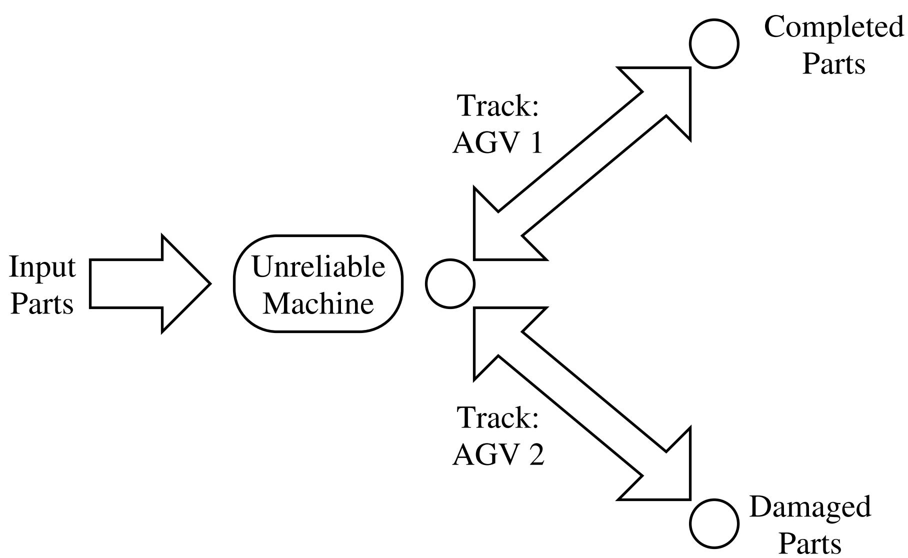

Figure 4: Basic operation of the plant.

The plant of Figure 4 is based around an "unreliable machine" (see [6] and [13]). The machine is used to process parts from an input queue, completed parts are moved to an output queue by an automated guided vehicle (AGV). The machine is considered unreliable because it is possible that it may break down and damage a part during operation. This behavior is captured in the plant model. Damaged parts are moved to a separate queue by a second AGV.

The Petri net model of the plant is shown in Figure 5, and a description of the various places and transitions is given in table 1. Places p_{11}, p_{12}, and p_{13} are used for supervisory control. They insure the mutual exclusion of certain operations within the plant. There is room for only one of the AGV's to pick up a part (either completed or damaged) from the machine, the exclusivity of p_5 and p_{10} is guaranteed by p_{11}. There is also only room for a single part, either damaged or complete, to wait for pickup by an AGV (places p_1, p_2 and p_6), this is guaranteed by p_{12}. Finally, p_{13} insures that if the machine breaks down, a new part will not be loaded until repairs are complete. It will be shown below that the combined action of these supervisory mechanisms create the potential for the plant to become deadlocked.

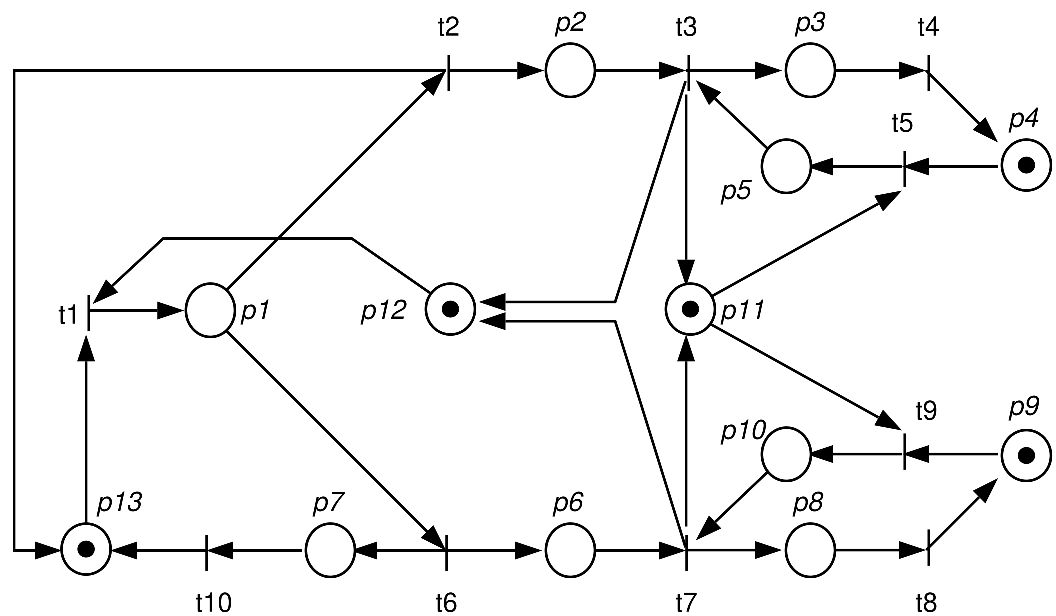

Figure 5: Petri net model of the unreliable machine plant, before accounting for possible deadlock.

The plant model has two uncontrollable transitions, t_2 and t_6. Transition t_6 represents machine break down and so obviously can not be controlled. Transition t_2 is considered uncontrollable because the controller can not force the machine to instantly finish a part that is not yet completed, nor does it direct the machine to stop working on an unfinished part. The transition is labeled uncontrollable to prevent the control design from attempting either of these two actions.

Places

p_1	Machine is "up and busy," part is being processed.
p_2	Part is waiting for transfer to completed-parts queue.
p_3	Part is being carried to completed-parts queue by AGV 1.
p_4	AGV 1 is free, away from part pick-up position.
p_5	AGV 1 is at pick-up position at machine.
p_6	Part is waiting for transfer to damaged-parts queue.
p_7	Machine is waiting to be repaired.
p_8	Part is being carried to damaged-parts queue by AGV 2.
p_9	AGV 2 is free, away from part pick-up position.
p_{10}	AGV 2 is at pick-up position at machine.
p_{11}	Control: Only one AGV at machine.
p_{12}	Control: Only one completed/damaged part at machine.
p_{13}	Control: Wait for repairs before starting new part.

Transitions

t_1	Part moves from input queue to machine.
t_2	*Uncontrollable:* Part processing is complete.
t_3	Part is picked up by AGV 1.
t_4	Part is deposited in completed-parts queue by AGV 1.
t_5	AGV 1 moves into pick-up position at machine.
t_6	*Uncontrollable:* Machine fails, part is damaged
t_7	Part is picked up by AGV 2.
t_8	Part is deposited in damaged-parts queue by AGV 2.
t_9	AGV 2 moves into pick-up position at machine.
t_{10}	Machine is repaired.

Table 1: Place and transition descriptions for the Petri net of Figure 5.

The plant has the following incidence matrix.

$$D_p = \begin{bmatrix} 1 & -1 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & -1 & 0 & 1 & 0 & -1 & 0 \\ -1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{12}$$

There are two uncontrolled siphons in the plant:

$$\begin{aligned} S_1 &= \{p_1, p_2, p_{10}, p_{11}, p_{12}\} \\ S_2 &= \{p_1, p_5, p_6, p_{11}, p_{12}\} \end{aligned}$$

Note that both uncontrolled siphons involve places p_{11} and p_{12}. The interaction of the mutual exclusions being enforced by these two places can result in a deadlock condition. Suppose that AGV 1 moves into position at the machine while it is working on a part, i.e., t_5 fires while $\mu_1 = 1$. Now suppose the machine breaks down (t_6 fires). AGV 1 is stuck waiting for a completed part, and AGV 2 is stuck waiting for AGV 1 to move out of the way: deadlock has occurred. This deadlock condition corresponds to S_1, S_2 corresponds to the analogous situation with the roles of the two machines reversed.

Further supervisors will be added to cause the two siphons to become controlled and thus prevent the possibility of deadlock. For each siphon, a control place will be created that insures that sum of the tokens in the siphon remains greater than or equal to one. Using the notation $l^T \mu_p \leq b$ we have, for S_1,

$$\begin{aligned} l &= \begin{bmatrix} -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & -1 & -1 & 0 \end{bmatrix}^T \\ b &= -1 \end{aligned}$$

Before proceeding to create the control structure using equation (8) and (9), we must check to see if the constraint meets condition (11). The incidence matrix of the uncontrollable portion of the plant, D_{uc}, is given by the second and sixth columns of D_p (since the uncontrollable transitions are t_2 and t_6).

$$l^T D_{\mathrm{uc}} = \begin{bmatrix} 0 & 1 \end{bmatrix}$$

We need all elements of $l^T D_{\mathrm{uc}}$ to be nonpositive to meet inequality (11). If the supervisor were created using the given value of l, then it would attempt to achieve its goal by inhibiting transition t_6, which corresponds to machine break down. Unfortunately the unreliable machine is not impressed by requests from the controller to simply not break, so we will construct a transformed constraint that eliminates the influence of the controller on t_6.

Following the technique of [11, 13], the transformed constraint is constructed by using positive row operations on $\begin{bmatrix} D_{\text{uc}} \\ l^T D_{\text{uc}} \end{bmatrix}$ to eliminate the positive number in $l^T D_{\text{uc}}$.

$$\begin{array}{c} \begin{bmatrix} -1 & -1 \\ 1 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 1 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 1 & 0 \end{bmatrix} \\ \begin{bmatrix} 0 & 1 \end{bmatrix} \end{array} \quad \underset{\text{Row } 14 = \text{Row } 1 + \text{Row } 14}{\Rightarrow} \quad \begin{array}{c} \begin{bmatrix} -1 & -1 \\ 1 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 1 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 1 & 0 \end{bmatrix} \\ \begin{bmatrix} -1 & 0 \end{bmatrix} \end{array}$$

Adding row 1 of D_{uc} to eliminate the positive number in $l^T D_{\text{uc}}$ corresponds to adding 1 to the first element of l to construct the new constraint vector l'.

$$\begin{aligned} l' &= \begin{bmatrix} 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & -1 & -1 & 0 \end{bmatrix}^T \\ b' &= -1 \end{aligned}$$

The new constraint, $l'^T \mu_p \leq b'$, represents the following inequality:

$$\mu_2 + \mu_{10} + \mu_{11} + \mu_{12} \geq 1$$

This inequality will insure that the number of tokens in S_1 remains positive and is also admissible with respect to the plant's uncontrollable transitions.

The incidence matrix and initial marking of the control place are now calculated.

$$\begin{aligned} D_c = -l'^T D_p &= \begin{bmatrix} -1 & 1 & 1 & 0 & -1 & 0 & 1 & 0 & 0 & 0 \end{bmatrix} \\ \mu_{c1_0} = b' - l'^T \mu_{p_0} &= 1 \end{aligned}$$

The control for siphon S_1 is shown as c_1 in Figure 6. The control, c_2, for siphon S_2 is calculated in a way directly analogous to that of c_1. The supervised plant meets the constraints placed on its behavior and is live.

6 Conclusions

A practical method for deadlock avoidance has been combined with results for enforcing constraints on Petri nets in the presence of uncontrollable transitions. The results expand

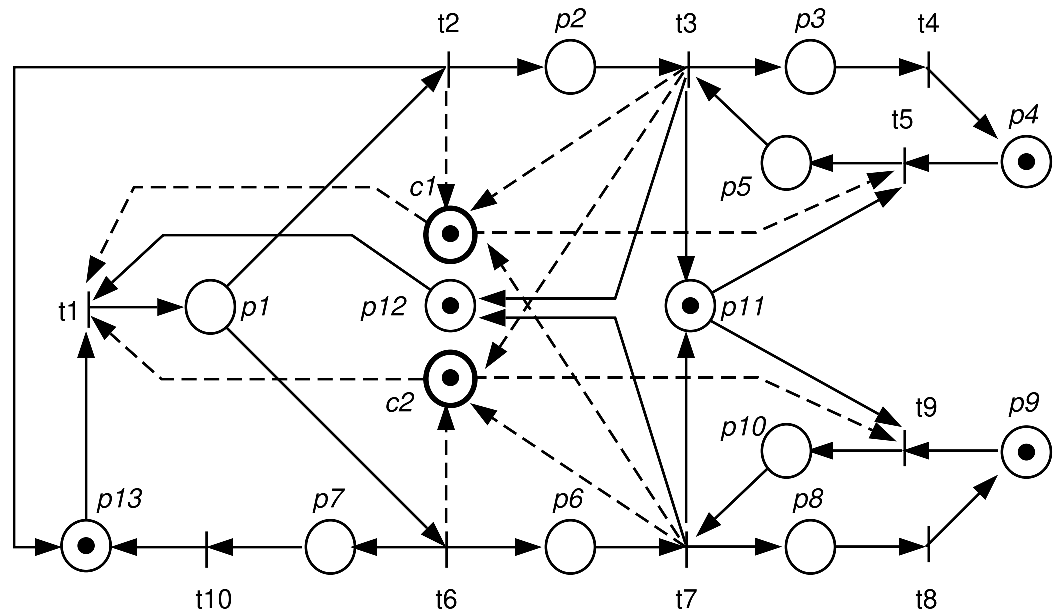

Figure 6: The unreliable machine model is now live.

the applicability and utility of the linear constraint inequality, or "general mutual exclusion constraint", used in [7–9, 11, 18]. Furthermore, they introduce a useful method for dealing with the deadlock problem into the area of PN DES control with its concept of uncontrollable plant transitions.

For some Petri nets, the maintenance of tokens in all of the net's siphons may not be sufficient to guarantee general liveness, though it will at least prevent complete deadlock. In this case, the deadlock avoidance method presented here may form a first step while another control layer actively plans firings to insure that all transitions remain live. This concept fits well with the idea of the "hierarchical intelligent controller" [1]. Here supervision is performed at one layer of the controller, to insure general system and safety constraints, and optimization and planning routines are carried out at another level, working within the boundaries established by the supervisor.

References

[1] P. J. Antsaklis and K. M. Passino, editors, *An Introduction to Intelligent and Autonomous Control*, Kluwer Academic Publishers, 1993.

[2] Z. A. Banaszak and B. H. Krogh, "Deadlock avoidance in flexible manufacturing systems with concurrently competing process flows", *IEEE Transactions on Robotics and Automation*, vol. 6, no. 6, pp. 724–734, June 1990.

[3] K. Barkaoui, "Liveness of Petri nets and its relations with deadlocks, traps, and invariants", Report 92-06, Laboratoire CEDRIC-CNAM, Paris, France, 1995.

[4] K. Barkaoui and I. B. Abdallah, "Deadlock avoidance in FMS based on structural theory of petri nets", In *IEEE Symposium on Emerging Technologies and Factory Automation*, volume 2, pp. 499–510, Piscataway, NJ, 1995. IEEE.

[5] J. Desel and J. Esparza, *Free Choice Petri Nets*, Cambridge University Press, 1995.

[6] A. A. Desrochers and R. Y. Al-Jaar, *Applications of Petri Nets in Manufacturing Systems*, IEEE Press, Piscataway, NJ, 1995.

[7] A. Giua, F. DiCesare, and M. Silva, "Generalized mutual exclusion constraints on nets with uncontrollable transitions", In *Proceedings of the 1992 IEEE International Conference on Systems, Man, and Cybernetics*, pp. 974–979, Chicago, IL, October 1992.

[8] Y. Li and W. M. Wonham, "Control of vector discrete event systems I – the base model", *IEEE Transactions on Automatic Control*, vol. 38, no. 8, pp. 1214–1227, August 1993, Correction in IEEE TAC v. 39 no. 8, pg. 1771, Aug. 1994.

[9] Y. Li and W. M. Wonham, "Control of vector discrete event systems II – controller synthesis", *IEEE Transactions on Automatic Control*, vol. 39, no. 3, pp. 512–530, March 1994.

[10] T. Minoura and C. Ding, "A deadlock prevention method for a sequence controller for manufacturing control", *International Journal of Robotics and Automation*, vol. 6, no. 3, March 1991.

[11] J. O. Moody and P. J. Antsaklis, "Supervisory control of Petri nets with uncontrollable/unobservable transitions", In *Proceedings of the 35th IEEE Conference on Decision and Control*, pp. 4433–4438, Kobe, Japan, December 1996.

[12] J. O. Moody and P. J. Antsaklis, "Supervisory control using computationally efficient linear techniques: A tutorial introduction", In *Proceedings of 5th IEEE Mediterranean Conference on Control and Systems*, volume Session MP1, Paphos, Cyprus, July 1997.

[13] J. O. Moody, P. J. Antsaklis, and M. D. Lemmon, "Feedback Petri net control design in the presence of uncontrollable transitions", In *Proceedings of the 34th IEEE Conference on Decision and Control*, volume 1, pp. 905–906, New Orleans, LA, December 1995.

[14] J. O. Moody, K. Yamalidou, M. D. Lemmon, and P. J. Antsaklis, "Feedback control of Petri nets based on place invariants", In *Proceedings of the 33rd IEEE Conference on Decision and Control*, volume 3, pp. 3104–3109, Lake Buena Vista, FL, December 1994.

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

[16] J. L. Peterson, *Petri Net Theory and the Modeling of Systems*, Prentice Hall, Engelwood Cliffs, NJ, 1981.

[17] W. Reisig, *Petri Nets*, Springer-Verlag, Berlin; New York, 1985.

[18] K. Yamalidou, J. O. Moody, M. D. Lemmon, and P. J. Antsaklis, "Feedback control of Petri nets based on place invariants", *Automatica*, vol. 32, no. 1, pp. 15–28, January 1996.