Supervisory Control of Petri Nets with Uncontrollable/Unobservable Transitions

Technical Report of the ISIS Group
at the University of Notre Dame
ISIS-96-004
August, 1996

John O. Moody and Panos J. Antsaklis
Department of Electrical Engineering
University of Notre Dame, Notre Dame, IN 46556
Email: jmoody@maddog.ee.nd.edu

Interdisciplinary Studies of Intelligent Systems

A smaller version of this report appeared in the *Proceedings of the* 35$^{\text{th}}$ *IEEE CDC*, Kobe, Japan, December 1996, Session ID: FP06 2.

This research was partially supported by the Army Research Office (DAAH04–96–1–0134).

Abstract

This paper expands upon results of previous research dealing with the supervisory control of Petri net modeled discrete event systems that contain uncontrollable transitions. It represents the current state of progress, as well as recent results, in an ingoing research project in the area of Petri nets in discrete event system control. The concept of unobservable plant transitions is introduced here and incorporated into the controller design procedure. New conditions are developed which govern the existence of controllers for these problems. Two procedures are presented for automatically generating controllers for plants that incorporate uncontrollable and unobservable events, and the current work toward determining conditions that indicate the optimality of these controllers is illustrated.

Keywords: Supervisory Control, Discrete Event Systems, Petri Nets

1 Introduction

The representation of discrete event system by ordinary Petri nets [7,9] allows for the use of many powerful algebraic tools for the realization of supervisory controllers [8,10] for these systems. In particular, it is possible to enforce a set of constraints on the plant state $\mu_p \in \mathbb{Z}^m, \mu_p \geq 0$ of the form

$$L\mu_p \leq b \tag{1}$$

where $L \in \mathbb{Z}^{n_c \times m}, b \in \mathbb{Z}^{n_c}$ and $\mathbb{Z}$ is the set of integers. The inequality in (1) is read, like all of the vector and matrix inequalities in this paper, with respect to each element on the corresponding left and right hand sides of the inequality. If all of the transitions within the plant Petri net are controllable and observable, then it has been shown ([6,11]) that (1) can be enforced by a Petri net controller which produces a place invariant (see [7,9]) on the closed loop plant-controller system. The Petri net incidence matrix, $D_c \in Z^{n_c \times n}$, of the controller is given by

$$D_c = -LD_p \tag{2}$$

where $D_p \in \mathbb{Z}^{m \times n}$ is the incidence matrix of the plant. The initial marking of the controller, $\mu_{c_0} \in \mathbb{Z}^{n_c}$ is

$$\mu_{c_0} = b - L\mu_{p_0} \tag{3}$$

where $\mu_{p_0} \in \mathbb{Z}^m$ is the initial marking of the plant. The incidence matrix, D, and marking, μ, of the closed loop plant-controller system are given by

$$D = \begin{bmatrix} D_p \\ D_c \end{bmatrix} \qquad \mu = \begin{bmatrix} \mu_p \\ \mu_c \end{bmatrix} \tag{4}$$

Controllers constructed in this way are identical to the monitors introduced by Giua *et al.* [1]

Some sets of constraints can not be enforced and thus appropriate controllers do not exist. It is possible to enforce the set of constraints (1) iff

$$b - L\mu_{p_0} \geq 0 \tag{5}$$

The discussion above assumes that all of the transitions in the Petri net plant will permit observation and can be inhibited if the controller deems it necessary. Li and Wonham [2,3] have made important contributions involving the optimal (maximally permissive) transformation of an original set of marking constraints into a set which accounts for possible uncontrollable actions within the plant net. The uncontrollable actions correspond to transitions within the plant Petri net that the controller has no power to inhibit. In general, the fact that a constraint is linear does not imply that the maximally permissive version of this constraint that accounts for uncontrollable transitions will also be linear. However Li and Wonham have presented sufficient conditions dealing with the structure of the uncontrollable portion of the plant that indicate when this situation will occur. A summary of some of the more important contributions of [3] is included in section 2 of this paper.

An alternative method for generating transformations of constraints to account for uncontrollable transitions was introduced in [5]. This method is computationally more efficient than that presented in [3], however it always yields a linear transformation, and thus is not always optimal. Section 3 of this paper expands upon these results by including the idea of unobservable transitions. Unobservable transitions provide no information to the controller when they fire. When they are present in the plant, it is necessary to transform constraints in a similar manner as that required for uncontrollable transitions. These results as well as new conditions for the existence of valid controllers and the necessity of certain transformation mechanics are presented in section 3. Section 4 proposes two procedures for automatically generating the transformations described in section 3. Section 5 includes examples which may help lead to the proof of conditions that indicate the optimality of linear constraint transformations. Concluding remarks appear in section 6.

2 Contributions of Li and Wonham

Important contributions to solving the problem of uncontrollable transitions in discrete event system control have been made by the researchers Li and Wonham [2,3]. Their primary contribution involves a method for transforming a set of general constraints into a set which accounts for uncontrollable transitions in the plant. The transformation is accomplished by symbolically solving an integer linear programming problem. They have also presented some sufficient conditions on the structure of the uncontrollable portion of the plant net for describing the constraint transformation in closed-form. These results are summarized below.

Li and Wonham introduce and base their work around a DES model they call the vector discrete event system (VDES). The VDES is, at least in one of its forms, equivalent to an ordinary Petri net. The discussion here will refer only to the Petri net form of the VDES. Before proceeding with the main contribution, the following definitions are made.

G : The Petri net.

$G_{\text{uc}} \in G$: The uncontrollable portion of the Petri net. It consists of the uncontrollable transitions and all of the places which are either inputs or outputs to these transitions.

Σ : The set of events associated with the firing of transitions.

$\Sigma_{\text{uc}} \in \Sigma$: The set of uncontrollable events.

$L(G, \mu)$: The language of Petri net G starting at state μ. The language is composed of all possible sequences of events due to transition firings.

$w \in L(G, \mu)$: A valid or possible string of events in the PN G starting from the state μ.

D_{uc} : Uncontrollable columns of the plant incidence matrix D_p.

The control goal is to enforce a single generalized mutual exclusion constraint on the marking behavior of G:

$$l_1\mu_1 + l_2\mu_2 + \ldots l_m\mu_m \leq b$$

or, in vector form,

$$l\mu \leq b \tag{6}$$

where l is a row vector composed of the elements $l_1 \ldots l_m$. The method of Li and Wonham is used to transform (6) into a form that eliminates the possibility of uncontrollable transition firings leading to a violation of (6). Li and Wonham define their controller as a boolean valued function which defines whether or not each controllable transition should be enabled given the current state.

A final state as a result of a sequence of transition firings can be represented as a single vector equation as follows. Given a sequence of N events, $w \in L(G, \mu)$, the final state is given by

$$\begin{aligned} \mu_N = \mu_0 + Dq_1 + Dq_2 + \ldots + Dq_N &= \mu + D(\underbrace{q_1 + q_2 + \ldots q_N}_{Q_w}) \\ \mu_N &= \mu + DQ_w \end{aligned} \tag{7}$$

where $q_i \in \mathbb{Z}^m$ is a vector of zeros and a single 1 corresponding to the transition which represents the i^{th} event in w.

Given (7), Li and Wonham define the control law $f^*(\alpha, \mu)$ which is equal to 1 when event α should be enabled given the current state μ.

$$f^*(\alpha, \mu) = \begin{cases} 1 & \text{if } \mu + Dq_\alpha \in [P] \\ 0 & \text{else} \end{cases}$$

where

$$\begin{aligned} [P] &= \{\mu | (\forall w \in L(G_{\text{uc}}, \mu)), l(\mu + D_{\text{uc}}Q_{\text{uc,w}}) \leq b\} \\ &= \{\mu | (l\mu + \max_{w \in L(G_{\text{uc}}, \mu)} lD_{\text{uc}}Q_{\text{uc,w}}) \leq b\} \end{aligned} \tag{8}$$

In other words, transition q_α can be enabled if it is impossible that any subsequent firing of uncontrollable transitions could violate the constraint $l\mu \leq b$.

The maximization problem in (8) involves searching a concave feasible region and may be very difficult to solve. However if the uncontrollable portion of the plant net $G_{\rm uc}$ contains no loops, then the set of possible strings $w \in L(G_{\rm uc}, \mu)$ can be simplified as follows.

$$\{Q_{\rm uc,w} | w \in L(G_{\rm uc}, \mu)\} = \{Q \in \mathbb{Z}^k | \mu + D_{\rm uc} Q \geq 0, Q \geq 0\}$$

With this simplification of the feasible region, the set $[P]$ of allowed states becomes

$$[P] = \{\mu | l\mu + l D_{\rm uc} Q^*(\mu) \leq b\} \tag{9}$$

where $Q^*(\mu)$ is the solution of

$$\max_{Q} \; l D_{\rm uc} Q$$
$$\text{s.t.} \begin{cases} D_{\rm uc} Q & \geq & -\mu \\ Q & \geq & 0 \text{ (integer)} \end{cases} \tag{10}$$

Symbolic solutions to the linear integer programming problem (10) will yield Q^* as a function of μ.

Li and Wonham have also shown how certain properties of the uncontrollable portion of the plant net, $D_{\rm uc}$, can limit the types of functions that $Q^*(\mu)$ can be. In general, if the graph of the uncontrollable portion of the net has a tree structure, then $Q^*(\mu)$ will have a closed-form expression.

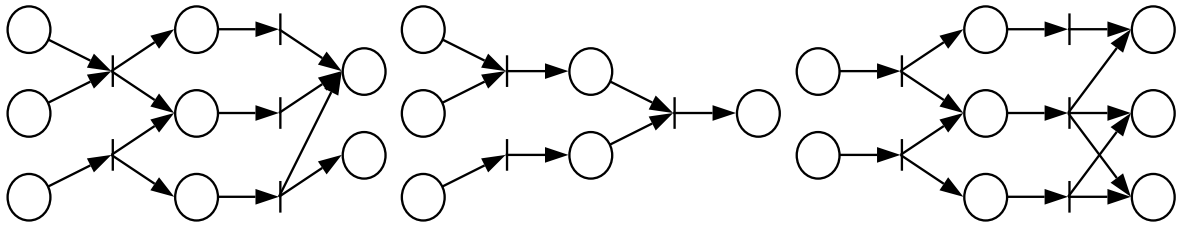
A Petri net is said to have a "general tree structure" if it is loop free and the places on one level of the Petri net graph act as sources for the transitions on the next level. These transitions then act as the exclusive source of tokens for the places in the next level of the graph. Places on one level receive tokens exclusively from the transitions one level back and deliver tokens to the transitions one level forward, and the same level-to-level relationship holds for the transitions as well. Furthermore, transitions do not share places, i.e., each place may contain no more than one output arc. Fig. 1a shows an example of a general tree structure. A "tree structure of type 1" exists when the Petri net has a general tree structure, and every transition has at most one output arc. A "tree structure of type 2" exists when the Petri net has a general tree structure, and every transition has at most one input arc. Figs. 1b and 1c show examples of types 1 and 2 tree structures. Strict definitions of these tree structures can be found in [3].

The following claims, proved in [3], are made based on the structure of $D_{\rm uc}$, the uncontrollable portion of the plant Petri net:

1. If $D_{\rm uc}$ has a type 1 tree structure, then the set $[P]$ of allowed states has the form of a disjunction of finitely many linear constraints:

$$[P] = \bigvee_{i=1}^{t} c_i \mu \leq b \tag{11}$$

for some c_i and t.



a) General b) Type 1 c) Type 2

Figure 1: Petri nets with tree structures.

2. If D_{uc} has a type 2 tree structure, then the set $[P]$ of allowed states has the form of a single linear constraint, in the same form as the original constraint $l\mu \leq b$:

$$[P] = c\mu \leq b \tag{12}$$

for some c.

Clearly, when situation 2 exists, a maximally permissive controller which will insure that the state remains within $[P]$ can be efficiently implemented using the control method outlined in section 1. However it is not necessary for D_{uc} to have a type 2 tree structure in order for the optimal control law to have the form $c\mu \leq b$. The following section describes a method for determining control laws in the form of (12) that account for uncontrollable and unobservable transitions and are guaranteed to enforce the constraints $L\mu \leq b$ with very little restriction on D_{uc}. Unfortunately these control laws are not guaranteed to be optimal.

3 Linear Constraint Transformations Due to Uncontrollable and Unobservable Transitions

Equation (2) in section 1 shows that it is possible to construct the incidence matrix D_c of a maximally permissive Petri net controller as a linear combination of the rows of the incidence matrix of the plant. Negative elements in D_c correspond to arcs from controller places to plant transitions. These arcs act to inhibit plant transitions when the corresponding controller places are empty, and thus they can only be applied to plant transitions which permit such external control. If, as in the previous section, we group all of the columns of D_p which correspond to transitions which can not be controlled into the matrix D_{uc}, then, in order for a set of constraints to be consistent with the uncontrollable transitions in the plant, it must be true that the matrix LD_{uc} contains no positive elements, as these will correspond to controlling arcs when constructing the controller as $D_c = -LD_p$. An enforceable set of constraints will satisfy

$$LD_{\text{uc}} \leq 0 \tag{13}$$

It is also possible that transitions within the plant may be unobservable, i.e., they are defined on the Petri net graph because they represent the occurrence of a real event, but these events are either impossible or too expensive to detect directly. It is also possible, in

the event of a sensor failure, that a transition might suddenly become unobservable, forcing a redesign or adaptation of the control law. The problem of handling unobservable transitions has been touched on in [4]. Here the constraints placed on a controller due to unobservable transitions are more rigorously defined, and a systematic method of dealing with them is proposed. It is illegal for the controller to change its state based upon the firing of an unobservable transition, because there is no direct way for the controller to be told that such a transition has fired. Both input and output arcs from the controller places are used to change the controller state based on the firings of plant transitions. Let the matrix D_{uo} represent the incidence matrix of the unobservable portion of the Petri net. This matrix is composed of the columns of D_p which correspond to unobservable transitions, just as D_{uc} is composed of the uncontrollable columns of D_p. It is illegal for the controller $D_c = -LD_p$ to contain any arcs in the unobservable portion of the net, thus an enforceable set of constraints will satisfy

$$LD_{\text{uo}} = 0 \tag{14}$$

Conditions (13) and (14) indicate that it is possible to observe a transition that we can not inhibit, but it is illegal to directly inhibit a transition that we can not observe.

Suppose, given a set of constraints $L\mu_p \leq b$, we construct the matrices LD_{uc} and LD_{uo} and observe that there are violations to conditions (13) and/or (14). Since the controller is made of a linear combination of the rows of D_p, it is interesting to consider the situation where we use the addition of further rows from D_{uc} in order to eliminate the positive elements of LD_{uc} and use rows from D_{uo} to eliminate the nonzero elements of LD_{uo}, i.e., if we are going to use a place invariant forming Petri net controller, what additions to the constraints would we need to make in order to eliminate positive elements from LD_{uc} and nonzero elements from LD_{uo}? What constraints, of the form $L'\mu_p \leq b'$, that can be enforced by an invariant-based controller, will also maintain the original constraint $L\mu_p \leq b$ while not interfering with the uncontrollable/unobservable portions of the plant? The following lemma appeared in [5].

Lemma 1.

$$\text{Let } R_1 \in \mathbb{Z}^{n_c \times m} \text{ satisfy } R_1\mu_p \geq 0 \ \forall \ \mu_p. \tag{15}$$

$$\text{Let } R_2 \in \mathbb{Z}^{n_c \times n_c} \text{ p.d. diagonal matrix} \tag{16}$$

If $L'\mu_p \leq b'$ where

$$L' = R_1 + R_2 L \tag{17}$$

$$b' = R_2(b + \mathbf{1}) - \mathbf{1} \tag{18}$$

and $\mathbf{1}$ is an n_c dimensional vector of 1's, then $L\mu_p \leq b$.

Proof. The transformed constraint is $(R_1 + R_2L)\mu_p \leq R_2(b + \mathbf{1}) - \mathbf{1}$. Because all of the elements are integers, the inequality can be transformed into a strict inequality: $(R_1 + R_2L)\mu_p < R_2(b + \mathbf{1})$. Because R_2 is diagonal and positive definite, $R_2^{-1}R_1\mu_p + L\mu_p < b + \mathbf{1}$. Assumptions (15) and (16) imply that all elements of the vector $R_2^{-1}R_1\mu_p \geq 0$, therefore $L\mu_p \leq b$. □

Lemma 1 shows a class of constraints, $L'\mu_p \leq b'$, which, if enforced, will imply that $L\mu_p \leq b$ are also enforced. The following lemma is used to show the conditions under which a particular set of constraints can be enforced on a particular Petri net.

Lemma 2. The constraint(s) $L'\mu_p \leq b'$, where $L' \neq 0$ and b' are defined by (17) and (18), can be enforced on a Petri net with initial marking μ_{p_0} iff

$$0 \leq R_1\mu_{p_0} \leq R_2(b + \mathbf{1} - L\mu_{p_0}) - \mathbf{1} \tag{19}$$

Proof. Substituting L' and b' into (19) gives $0 \leq b' - L'\mu_{p_0}$ which is equivalent to the condition $L'\mu_{p_0} \leq b'$, which states that the initial conditions of the plant must fall within the acceptable region of the constraints. Clearly, if a controller does exist, then the initial conditions of the plant must not violate the constraints. Furthermore, as shown in in [1,6,11], if the initial conditions lie within the acceptable region of the plant (inequality (5)), a controller to enforce the conditions can be computed with incidence matrix $D_c = -L'D_p$ and initial marking $\mu_{c_0} = b' - L'\mu_{p_0}$. □

Proposition 3 combines conditions (13) and (14) with the conditions for creating a valid set of transformed constraints in lemmas 1 and 2 to show how to construct a Petri net controller to enforce the constraints $L\mu_p \leq b$ which does not involve inappropriate interference with the uncontrollable and unobservable portions of the plant net.

Proposition 3. Let a plant Petri net with incidence matrix D_p be given with a set of uncontrollable transitions described by D_{uc} and a set of unobservable transitions described by D_{uo}. A set of linear constraints on the net marking, $L\mu_p \leq b$, are to be imposed. Assume R_1 and R_2 meet (15) and (16) with $R_1 + R_2L \neq 0$ and let

$$\begin{bmatrix} R_1 & R_2 \end{bmatrix} \begin{bmatrix} D_{\text{uc}} & D_{\text{uo}} & -D_{\text{uo}} & \mu_{p_0} \\ LD_{\text{uc}} & LD_{\text{uo}} & -LD_{\text{uo}} & L\mu_{p_0} - b - \mathbf{1} \end{bmatrix} \leq \begin{bmatrix} 0 & 0 & 0 & -\mathbf{1} \end{bmatrix} \tag{20}$$

Then the controller

$$D_c = -(R_1 + R_2L)D_p = -L'D_p \tag{21}$$

$$\mu_{c_0} = R_2(b + \mathbf{1}) - \mathbf{1} - (R_1 + R_2L)\mu_{p_0} = b' - L'\mu_{p_0} \tag{22}$$

exists and causes all subsequent markings of the closed loop system (4) to satisfy the constraint $L\mu_p \leq b$ without attempting to inhibit uncontrollable transitions and without detecting unobservable transitions.

Proof. According to (2) and (3), equations (21) and (22) define a controller that enforces the constraint $L'\mu_p \leq b'$. Lemma 1 shows that if assumptions (15) and (16) are met then a controller which enforces a particular constraint $L'\mu_p \leq b'$ will also enforce the constraint $L\mu_p \leq b$. The fourth column of inequality (20) indicates that the condition in lemma 2 is satisfied, thus the controller exists and the control law can be enforced. The first column

of (20) indicates that $L'D_{\rm uc} \leq 0$, thus condition (13) is satisfied and no controller arcs are drawn to the uncontrollable transitions. The second and third columns of (20) indicate that $L'D_{\rm uo} = 0$, thus condition (14) is satisfied and no arcs are drawn between the controller places and the unobservable plant transitions. □

Note that $\left[\begin{array}{cc} R_1 & R_2 \end{array}\right]$, which is used to describe the constraint transformation, multiplies from the left in (20), thus these matrices represent the use of rows from $D_{\rm uc}$ to eliminate positive elements from $LD_{\rm uc}$, and the use of rows from $D_{\rm uo}$ to zero the elements of $LD_{\rm uo}$, as discussed above.

The procedure above allows for the complete transformation of one set of linear constraints, $L\mu_p \leq b$, into another, $L'\mu_p \leq b'$. However the procedure described in section 2 indicated that when it is possible to do a simple linear transformation of the constraints, it is not necessary to change the right-hand-side of the constraint inequality, i.e., $b' = b$. The following example is used to illustrate why $b' = R_2(b + \mathbf{1}) - \mathbf{1}$ in the development above and that it is sometimes necessary to change both the left-hand and the right-hand sides of the constraint inequality in order to obtain a linear form for the optimal transformed constraint.

Example. Consider the Petri net of Fig. 2. The control goal is to impose the condition

$$\mu_4 \leq 2 \tag{23}$$

or, in matrix form,

$$\underbrace{\left[\begin{array}{cccc} 0 & 0 & 0 & 1 \end{array}\right]}_{L} \mu_p \leq \underbrace{2}_{b}$$

The incidence matrices of the plant and the uncontrollable portion of the plant are

$$D_p = \left[\begin{array}{rrrr} -1 & 0 & 0 & 1 \\ 0 & -1 & 0 & 2 \\ 2 & 1 & -2 & 0 \\ 0 & 0 & 1 & -2 \end{array}\right] \qquad D_{\rm uc} = \left[\begin{array}{r} 0 \\ 0 \\ -2 \\ 1 \end{array}\right]$$

Transition t_3 is uncontrollable, thus $D_{\rm uc}$ is the third column of D_p.

Checking the quantity $LD_{\rm uc}$ we see that $LD_{\rm uc} = 1$ is positive, thus a direct attempt to enforce (23) would attempt to control the uncontrollable transition. A simple row operation is then performed on $\left[\begin{array}{c} D_{\rm uc} \\ LD_{\rm uc} \end{array}\right]$ to find R_1 and R_2 which satisfy (15), (16), and (20):

$$\begin{array}{rcl} R_1 & = & \left[\begin{array}{cccc} 0 & 0 & 1 & 0 \end{array}\right] \\ R_2 & = & 2 \end{array}$$

which yields the following transformation of the constraint:

$$\left(\left[\begin{array}{cccc} 0 & 0 & 1 & 0 \end{array}\right] + 2\left[\begin{array}{cccc} 0 & 0 & 0 & 1 \end{array}\right]\right)\mu_p \leq 2(2+1) - 1$$

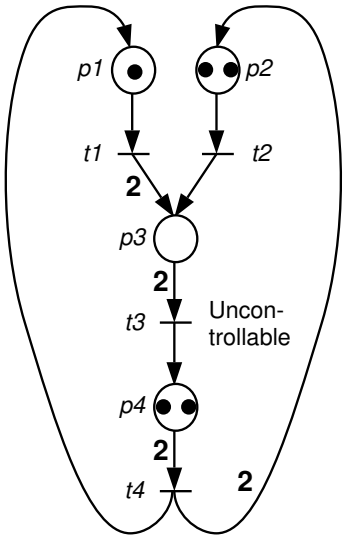


Figure 2: Should transition t_2 be allowed to fire if the constraint is $\mu_4 \leq 2$?

or

$$\mu_3 + 2\mu_4 \leq 5 \tag{24}$$

Now we will obtain a similar transformation using the method described in section 2. The single uncontrollable transition contains one input and one output arc, giving it a type 2 tree structure. The optimal constraint transformation should then be a linear function of μ_p given by (9) which requires us to solve the integer linear programming problem (10):

$$\max_Q \; LD_{\text{uc}}Q = \max Q$$

$$\text{s.t.} \left\{ \begin{array}{rcl} \left[\begin{array}{c} \mu_1 \\ \mu_2 \\ \mu_3 \\ \mu_4 \end{array} \right] + \left[\begin{array}{c} 0 \\ 0 \\ -2 \\ 1 \end{array} \right] Q & \geq & 0 \\ Q & \geq & 0 \text{ (integer)} \end{array} \right.$$

The only nontrivial constraint above is $\mu_3 - 2Q \geq 0$. Thus Q^*, the maximum possible value of Q, is given by

$$Q^* = \text{floor } \frac{1}{2}\mu_3$$

which, according to (9), gives the control law

$$\text{floor } \frac{1}{2}\mu_3 + \mu_4 \leq 2 \tag{25}$$

which is indeed the optimal control law for this problem, but it is not a linear constraint. If we naively multiplied both sides of (26) by 2 in order to eliminate the fraction, we would have

$$\mu_3 + 2\mu_4 \leq 4 \tag{26}$$

which is a linear constraint, but is not maximally permissive. Consider the plant in the state shown in Fig. 2. If we are to insure that μ_4 remains less than or equal to 2, should the firing of transition t_2 be allowed? Clearly the answer is yes, and (24) and (25) would both allow this, but condition (26) would not, thus it is not maximally permissive.

Because all of the elements in the inequalities are integers, constraint (24) is the linear equivalent of the nonlinear constraint (25) and is in a form which can be instantly translated into a Petri net controller. This is the reason why the procedure outlined above includes a transformation for b to b'.

4 Generating Constraint Transformations

The usefulness of proposition 3 for specifying controllers to handle plants with uncontrollable and unobservable transitions lies in the ease in which the matrices R_1 and R_2, with the appropriate properties, can be generated. Section 4.1 shows how the information in the proposition can be converted into an integer linear programming for determining R_1 and R_2, and section 4.2 proposes a scheme for determining valid R_1 and R_2 values by performing matrix row operations.

4.1 An Integer Linear Program for finding R_1 and R_2

It is possible to convert the conditions in proposition 3 into an integer linear programming problem (ILP) in the standard form of

$$\min_x z(x) = x^T c \quad \text{s.t.} \begin{cases} x^T A &= d \\ x &\geq 0 \text{ (integer)} \end{cases} \tag{27}$$

We will consider only a single constraint on the system; multiple constraints can be handled individually and independently. Thus $n_c = 1$, L and R_1 are vectors, and b and R_2 are scalars.

In order to satisfy condition (15), we can specify that $R_1 \geq 0$, since we know that $\mu_p \geq 0$. In fact, it is necessary to specify that all of the elements of R_1 are greater than or equal to zero if the markings of the plant places are unbounded or they are bounded but the bound is not known.

Condition (16) states that we want $R_2 > 0$. In order to obtain variables that fit the conditions for x in (27), define

$$R_2' = R_2 - 1 \tag{28}$$

Since R_2 is an integer, $R_2' \geq 0$ implies that $R_2 > 0$.

Substituting the new variable R_2' into condition (13) yields

$$R_1 D_{\text{uc}} + R_2' L D_{\text{uc}} \leq -L D_{\text{uc}}$$

A vector of slack variables, $R_3 \geq 0$, is introduced in order to convert the inequality into an equality.

$$R_1 D_{\text{uc}} + R_2' L D_{\text{uc}} + R_3 = -L D_{\text{uc}}$$

R_3 is a column vector like R_1 but with dimension equal to the number of columns of D_{uc} (the number of uncontrollable transitions).

Substituting R_2' into condition (14) gives

$$R_1 D_{\text{uo}} + R_2' L D_{\text{uo}} = -L D_{\text{uo}}$$

which is already in the form of an equality, so an additional slack variable is unnecessary.

The new variable R_2' is now substituted into the condition given in lemma 2 which indicates whether the given constraint transformation can be implemented.

$$R_1 \mu_{p_0} + R_2'(L\mu_{p_0} - (b+1)) \leq b - L\mu_{p_0} \tag{29}$$

Let

$$R = \begin{bmatrix} R_1 & R_2' & R_3 \end{bmatrix}$$

and the ILP can now be defined as

$$\begin{aligned} \min_R z(R) = R \begin{bmatrix} \mu_{p_0} \\ L\mu_{p_0} - b - 1 \\ 0 \end{bmatrix} \\ \text{s.t.} \begin{cases} R \begin{bmatrix} D_{\text{uc}} & D_{\text{uo}} \\ LD_{\text{uc}} & LD_{\text{uo}} \\ I & 0 \end{bmatrix} = -L \begin{bmatrix} D_{\text{uc}} & D_{\text{uo}} \end{bmatrix} \\ R \geq 0 \text{ (integer)} \end{cases} \end{aligned} \tag{30}$$

which is in the form of (27).

After solving (30), if the minimum of the objective function $z^* = z(R^*)$ is greater than $b - L\mu_{p_0}$ then the problem can not be solved as there are no values of R_1 and R_2 which will satisfy the condition in lemma 2. If the minimum is less than or equal to $b - L\mu_{p_0}$, then transform R_2' back into R_2 and generate the controller using the formulae in proposition 3.

It is possible that there may be problems associated with this method of generating R_1 and R_2. For a controller to exist, we need the objective function of the ILP, $z(R) \leq b - L\mu_{p_0}$, however it not clear why we should attempt to minimize this function or what the results of such an attempt might be. It is also possible that the ILP may yield the pathological transformation $L' = R_1 + R_2 L = 0$ when there are other nonzero possibilities for L'.

4.2 A Procedure Based on Matrix Row Operations

It is possible to obtain appropriate constraint transformations by performing row operations on a matrix containing the uncontrollable and unobservable columns of the plant

incidence matrix. The computational part of this procedure involves little more than the integer triangularization of a matrix, and thus it is simpler to compute R_1 and R_2 using this method than by using the ILP presented in the previous section. Before presenting the algorithm itself, the following terms are clarified:

D_{uc} : An $m \times n_{\text{uc}}$ matrix consisting of the columns of the plant incidence matrix D_p that correspond to transitions that are uncontrollable, but which may be observed. n_{uc} is the number of these transitions.

D_{uo} : An $m \times n_{\text{uo}}$ matrix consisting of the columns of D_p which are unobservable (just as defined in previous sections).

In the discussions in previous sections, D_{uc} may have included columns which were unobservable as well as uncontrollable, but here all of the columns of D_{uc} are observable. Conditions (13) and (14) show that unobservability implies stricter demands than uncontrollability, and in fact, any transition labeled as unobservable is also uncontrollable. D_{uc} is defined here as being strictly observable so that we can relax our requirements when dealing with this section of the matrix. Algorithm 1 presents the procedure for determining R_1 and R_2.

Algorithm 1 (Constraint Transformation).

Input: $L \in \mathbb{Z}^{n_c \times m}, b \in \mathbb{Z}^{n_c}, D_{\rm uc} \in \mathbb{Z}^{m \times n_{\rm uc}}, D_{\rm uo} \in \mathbb{Z}^{m \times n_{\rm uo}}, \mu_{p_0} \in \mathbb{Z}^m$

if ($LD_{\rm uc} \leq 0$ and $LD_{\rm uo} = 0$) then

  $R_1 := 0_{n_c \times m}, R_2 := I_{n_c \times n_c}$

else

  $$M := \begin{bmatrix} D_{\rm uc} & D_{\rm uo} & \\ LD_{\rm uc} & LD_{\rm uo} & I \end{bmatrix}$$

  Let $M(i,j)$ represent the $(i,j)^{\rm th}$ element of matrix M.

  Zero all positive elements in the $LD_{\rm uc}$ portion of M following the procedure in Algorithm 2.

  if $M(m+1 \ldots m+nc, 1 \ldots n_{\rm uc})$ has any positive elements then

    FAILURE: Controller arcs were introduced by the row operations into the uncontrollable portion of the plant, and they can not be eliminated due to lack of negative elements in $D_{\rm uc}$.

  end if

  Zero all elements in the $LD_{\rm uo}$ portion of the M matrix following the procedure in Algorithm 3.

  if $M(m+1 \ldots m+nc, n_{\rm uc}+1 \ldots n_{\rm uc}+n_{\rm uo})$ has any nonzero elements then

    FAILURE: The unobservable portion of the plant contains arcs which can not be eliminated.

  end if

  $R_1 := M(m+1 \ldots m+n_c, n_{\rm uc}+n_{\rm uo}+1 \ldots n_{\rm uc}+n_{\rm uo}+m)$

  $R_2 := M(m+1 \ldots m+n_c, n_{\rm uc}+n_{\rm uo}+m+1 \ldots n_{\rm uc}+n_{\rm uo}+m+n_c)$

end if

$L' := R_1 + R_2 L$

$b' := R_2(b+\mathbf{1}) - \mathbf{1}$

if $L'\mu_{p_0} > b'$ then

  FAILURE: Control law is infeasible.

end if

Output: L' and b'.

As was done in section 4.1, we shall insure that condition (15) is met by making $R_1 \geq 0$. In terms of row operations, this means that elements in rows are eliminated strictly through addition, never through subtraction, and that rows can be premultiplied only by positive integers. The procedure for zeroing out the elements in a column of numbers which have the opposite sign of the "pivot" is given in Algorithm 4.

Algorithm 2 (Elimination of positive elements from D_{uc}).
Input: Working matrix $M \in \mathbb{Z}^{(m+n_c)\times(n_{\text{uc}}+n_{\text{uo}}+m+n_c)}$
$i := 1$
while $i \leq \min(n_{\text{uc}}, m)$
  if $M(i \ldots m, i)$ has any negative elements then
    Let j be the index of a row in $M(i \ldots m, i)$ which
      contains a negative element.
    Exchange rows i and j of M
    Use the negative pivot value at $M(i,i)$ to eliminate
      all positive integers in the column $M(i \ldots m + n_c, i)$
      (See Algorithm 4.)
  else if $M(m+1 \ldots m+n_c, i)$ has any positive elements then

    FAILURE: A controller arc can not be eliminated
      because there are no negative elements in the
      corresponding column of D_{uc}.
  end if
  $i := i + 1$
end while
Output M and i

Algorithm 1 insures that condition (16) is met because the procedure for choosing the "pivot" elements never picks from the LD_{uc} and LD_{uo} portions of the M matrix. Combined with the zeroing procedure of Algorithm 4, these steps insure that the R_2 portion of the M matrix is diagonal with strictly positive elements.

Algorithms 2 and 3 (called by Algorithm 1) are used to make sure that the transformed constraints meet conditions (13) and (14). The feasibility check at the end of Algorithm 1 directly tests the condition of lemma 2 to insure that the controller does exist. The instructions for picking positive or negative elements to act as pivots in the two main loops are left specifically vague. Different methods of choosing the pivot will lead to different constraint transformations. It would be possible, for instance, to find a basis for all valid constraint transformations by repeating the procedures in Algorithm 1 whenever there was a choice of more than one pivot for a given column.

Even if all the possible pivots are used in Algorithm 2, there may still be other R_1 and R_2 values which yield transformed constraints that meet condition (13). Algorithm 2 forces all positive elements in the LD_{uc} portion of the matrix to go to zero. However condition

(13) states that we need $LD_{\text{uc}} \leq 0$, so the question is, is it ever desirable or necessary to transform positive elements of LD_{uc} into negative elements? Should Algorithm 2 incorporate this ability? At this time, this question can not be answered definitely, however the following points, starting with the lemma below, are presented to shed light on the question.

Algorithm 3 (Zeroing of all elements in D_{uo}).
Input: Working matrix $M \in \mathbb{Z}^{(m+n_c)\times(n_{\text{uc}}+n_{\text{uo}}+m+n_c)}$ and i
while $i \leq \min(n_{\text{uc}} + n_{\text{uo}}, m)$
  if $M(i \ldots m, i)$ contains any negative elements then
    Let j be the index of a row in $M(i \ldots m, i)$ which
      contains a negative element.
    Exchange rows i and j of M
    $k := 1$
  else
    $k := 0$
  end if
  if $M(i \ldots m, i)$ has any positive elements then
    Let j be the index of a row in $M(i \ldots m, i)$ which
      contains a positive element.
    Exchange rows $i + k$ and j of M
    Use the positive pivot value at $M(i + k, i)$ to eliminate
      all negative integers in the column $M(i+k \ldots m+n_c, i)$
      (See Algorithm 4.)
  end if
  if $k = 1$ then
    Use the negative pivot value at $M(i, i)$ to eliminate
      all positive integers in the column $M(i \ldots m + n_c, i)$
  end if
  if $M(m + 1 \ldots m + n_c, i)$ has nonzero elements then
    FAILURE: Controller arc(s) can not be eliminated from
      the unobservable portion of the plant.
  end if
  $i := i + 1$
end while
Output M

Lemma 4. A single controller place can possess either an input or an output arc (or neither) to any given plant transition, but it will never contain both, i.e., controllers constructed according to the method outlined in section 1 contain no self-loops.

Proof. The incidence matrix of the invariant forming controller for establishing the constraint $L\mu_p \leq b$ is defined by $D_c = -LD_p$. Positive elements in D_c refer to arcs from

plant transitions to the controller, and negative elements refer to arcs from the controller to the plant transitions. There are no allowances for self loops in this controller construction method. The matrix D_c^+ consists of the positive elements in D_c, D_c^- consists of the negative, and all other elements are zero, i.e., there are never any cancellations between D_c^+ and D_c^- used in forming D_c. See [11]. □

Algorithm 4 (Column Zeroing).
Input: Working matrix $M \in \mathbb{Z}^{(m+n_c)\times(n_{uc}+n_{uo}+m+n_c)}$ and pivot position (p, j).
$i := p + 1$
while $i \leq m + n_c$
  if $M(i,j)$ is opposite $M(p,j)$ in sign then
    while $M(i,j) \neq 0$
      if $|M(p,j)| > |M(i,j)|$ then
        $d :=$ floor $(-M(p,j)/M(i,j))$
        if $(\text{mod}(M(p,j), M(i,j)) = 0)$ then
          $d := d - 1$ (Don't zero-out the pivot!)
        end if
        $M(p,.) := M(p,.) + dM(i,.)$ (row operation)
      else
        $d :=$ floor $(-M(i,j)/M(p,j))$
        $M(i,.) := M(i,.) + dM(p,.)$ (row operation)
      end if
    end while
  end if
  $i := i + 1$
end while
Output M

A positive element in the matrix LD_{uc} means that the controller would draw an arc to a transition in the uncontrollable portion of the plant. Lemma 4 tells us that this element is not the result of battling input and output arcs. The controller wishes to output to this transition, and it has no reason to receive input from this transition. Algorithm 2 would then be used to eliminate the controller's interaction with the transition all together. But what would happen if row operations were used to change this output transition of the controller into an input transition?

Place invariant forming controllers use their places to take up the slack that exists in the inequality $L\mu \leq b$. Specifically, the marking of the controller, μ_c, is given by

$$\mu_c = b - L\mu_p$$

when the closed loop system is in operation. Negative elements in D_c correspond to transitions which decrease the slack μ_c and lead to the controller placing inhibitions on the plant's

behavior. Positive elements correspond to transitions which increase the slack and give the plant a greater margin of freedom. A positive element in LD_{uc} means that the firing of the corresponding transition will decrease the slack. If this positive number is changed, via row operations and the transformation of L to L', to a negative number, it means that an operation that indicated a decrease in the slack and the freedom of the plant in the untransformed system has become one that indicates an increase in slack and plant freedom in the transformed system. The fact that the quantity $b - L\mu_p$ has decreased while the quantity $b' - L'\mu_p$ has increased seems to indicate that the controller in the transformed system is concerning itself with matters that have gone beyond the original request of insuring that $L\mu_p \leq b$. This is the reason that Algorithm 2 makes no attempt to find row operations that would lead to this situation, however the ideas presented here are by no means a proof that it is never necessary to convert output arcs into input arcs in order to handle uncontrollable transitions.

5 Thoughts on Constraint Transformation Optimality

The place invariant controller method yields optimal maximally permissive supervisors for enforcing linear constraints of the form $L\mu_p \leq b$. When these constraints are transformed, because of uncontrollability and unobservability of certain transitions, into $L'\mu_p \leq b'$, the invariant based control method will still yield an optimal realization of the transformed constraint, unfortunately the new constraint itself may not be a maximally permissive realization of the original constraint. The maximally permissive transformation of a linear predicate on the plant's marking may be a nonlinear predicate which can not be optimally controlled by an invariant based controller.

At this time, a complete description of the conditions under which an optimal transformation of linear constraints is another set of linear constraints is unknown. Li and Wonham [3] have shown that when the uncontrollable portion of the plant has a "type 2 tree structure" (see section 2) that the optimal transformation will be linear, however this condition is only sufficient, not necessary.

In the following subsections, three simple examples are presented which indicate the current direction of research on the attempt to answer the question of the existence of an optimal linear constraint transformation. These examples are not presented as a replacement for a mathematical proof, they merely indicate the current direction of thinking which may in the near future lead to a proof.

5.1 Unique L', Unique Controller, Optimal Control

The incidence matrix of the Petri net in Fig. 3 is

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

Transition t_2 is uncontrollable and the uncontrollable portion of the net has a type 2 tree structure. The control goal is

$$\underbrace{\begin{bmatrix} 0 & 0 & 1 \end{bmatrix}}_{L} \mu_p \leq b$$

Following the procedure of Algorithm 1 indicates that all valid transformations of L to L', ignoring transformations which would change output arcs to input arcs as discussed at the end of section 4, are given by

$$L' = k \begin{bmatrix} 0 & 1 & 1 \end{bmatrix}$$

where $k \in \mathbb{Z}, k > 0$. The multiplication of the constraint inequality by a constant does not change the set of allowable states, so we shall set $k = 1$. Thus, for this example, there is a unique transformation of L to L'. The incidence matrix of the corresponding controller is, of course, also unique.

$$D_c = \begin{bmatrix} -1 & 0 & 1 \end{bmatrix}$$

and this is the optimal controller, and L' is the optimal constraint transformation, as we should expect since the uncontrollable portion of the plant has a type 2 tree structure.

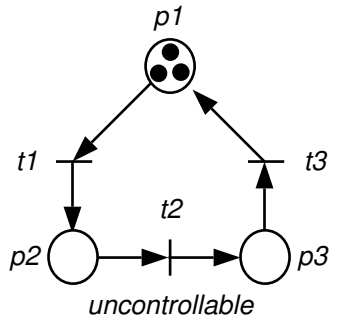


Figure 3: A Petri net that yields an optimal constraint transformation.

5.2 Infinite L', Unique Controller, Optimal Control

The incidence matrix of the Petri net in Fig. 4 is

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

Transition t_2 is uncontrollable and the uncontrollable portion of the net does not have a type 2 tree structure (it is type 1). The control goal is

$$\underbrace{\begin{bmatrix} 0 & 0 & 0 & 1 \end{bmatrix}}_{L} \mu_p \leq b$$

An infinite number of different L' values can be obtained by Algorithm 1

$$L' = \begin{bmatrix} 0 & k_1 & k_2 & k_1 + k_2 \end{bmatrix}$$

where the constants $k_1, k_2 \in \mathbb{Z}, k_1, k_2 \geq 0$, and at least one constant is nonzero. However, when we obtain the controller for these transformed constraints we get

$$D_c = k \begin{bmatrix} -1 & 0 & 1 \end{bmatrix}$$

where $k = k_1 + k_2$. Multiplying the incidence matrix of the controller does nothing to change the actual control law, it only changes the number of tokens the controller uses in its internal representation of slack. Thus the controller for this problem is unique, and in fact, the control law which it enforces is the optimal control law. The only states that it prevents the plant from entering are those states that would directly or indirectly (through the firing of t_2) lead to the violation of the original constraint.

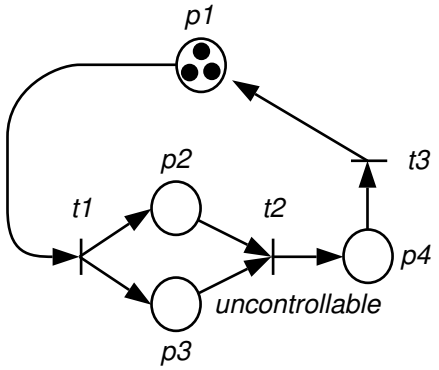


Figure 4: A Petri net that yields many constraint transformations but a unique optimal controller.

5.3 Infinite L', Infinite Controllers, Suboptimal Control

The incidence matrix of the Petri net in Fig. 5 is

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

Transition t_3 is uncontrollable and the uncontrollable portion of the net does not have a type 2 tree structure, but it does have the same structure as the example in section 5.2. The control goal is

$$\underbrace{\begin{bmatrix} 0 & 0 & 0 & 1 \end{bmatrix}}_{L} \mu_p \leq b$$

As in 5.2, an infinite number of different L' values can be obtained by Algorithm 1

$$L' = \begin{bmatrix} 0 & k_1 & k_2 & k_1 + k_2 \end{bmatrix}$$

where $k_1, k_2 \in \mathbb{Z}, k_1, k_2 \geq 0$, and at least one constant is nonzero. This time, each constraint yields a different controller

$$D_c = \begin{bmatrix} -k_1 & -k_2 & 0 & k_1 + k_2 \end{bmatrix}$$

All of these controllers result in the enforcement of suboptimal control laws. The optimal control law is

$$\mu_4 + \min(\mu_2, \mu_3) \leq b$$

which can not be realized by a linear place invariant forming controller.

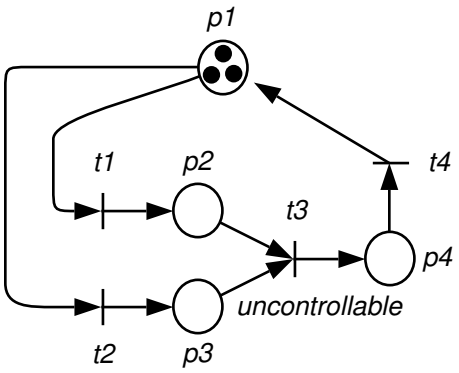


Figure 5: A Petri net that yields many constraint transformations and controllers which all lead to suboptimal control.

6 Conclusions

This paper represents ongoing work in the formulation and formalization of a discrete event system control design procedure that relies on simple and efficient linear algebraic techniques. A systematic method for dealing with unobservable, as well as uncontrollable, plant transitions has been presented in section 3 which included new conditions dictating whether or not a controller exists for a particular set of transformed constraints. The usefulness and necessity of transforming both the right and left hand sides of the set of plant constraints have been demonstrated, and two new procedures for automatically generating valid constraint transformations have been presented.

An important question for future research in this area deals with the optimality of controllers that are produced using the procedures described here. The examples presented in section 5 point to a relationship between the uniqueness of the controller and the optimality of the control law. These examples certainly do not represent a proof of this relationship, but it is hoped that they will help guide in the search for a method of proving or disproving the implied hypothesis.

References

[1] 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.

[2] 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.

[3] 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.

[4] J. O. Moody, P. J. Antsaklis, and M. D. Lemmon, "Automated design of a Petri net feedback controller for a robotic assembly cell", In *Proceedings of 1995 INRIA/IEEE Symposium on Emerging Technologies and Factory Automation*, volume 2, pp. 117–128, Paris, France, October 1995.

[5] 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 Conference on Decision and Control*, volume 1, pp. 905–906, New Orleans, LA, December 1995.

[6] 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 Conference on Decision and Control*, volume 3, pp. 3104–3109, Lake Buena Vista, FL, December 1994.

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

[8] P. J. G. Ramadge and W. M. Wonham, "The control of discrete event systems", *Proceedings of the IEEE*, vol. 77, no. 1, pp. 81–97, 1989.

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

[10] W. M. Wonham and P. J. G. Ramadge, "On the supremal controllable sublanguage of a given language", *SIAM Journal of Control Optimization*, vol. 25, no. 3, pp. 637–659, 1987.

[11] 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.