1000-9825/2005/16(03)0419 ©2005 Journal of Software 软 件 学 报 Vol.16, No.3

针对一般线性约束的 Petri 网控制器设计方法*

王寿光+, 颜钢锋

(浙江大学 电气工程学院,浙江 杭州 310027)

A Method for the Design of Petri Net Controller Enforcing General Linear Constraints

WANG Shou-Guang+, YAN Gang-Feng

(College of Electrical Engineering, Zhejiang University, Hangzhou 310027, China)

+ Corresponding author: Phn: +86-571-87951637, E-mail: wsg5000@sina.com, http://ee.zju.edu.cn

Received 2004-03-23; Accepted 2004-11-15

Wang SG, Yan GF. A method for the design of Petri net controller enforcing general linear constraints. *Journal of Software*, 2005,16(3):419−426. DOI: 10.1360/jos160419

Abstract: The problem of constructing a Petri net feedback controller, which enforces linear inequality constraints involving the marking vector and the Parikh vector on a discrete event system (DES) modeled by Petri nets (PN), is discussed in this paper. A novel method for design of controller enforcing the constraints is presented. First the constraints involving the marking and Parikh vectors are transformed into the constraints involving Parikh vector only using Petri net state equality, and then the controller is constructed based on the viewpoint that a place can be seen as a linear inequality constraint on the Parikh vector. The method is proved to be simpler and more efficient than that presented by Iordache and Moody through an applied instance that was also used by Moody et al., and holds remarkable advantage especially for large systems.

Key words: Petri net (PN); DES (discrete event system); supervisor control; linear inequality constraints; Parikh vector

摘 要: 针对基于Petri网离散事件系统关于标识向量和Parikh向量的不等式约束反馈控制器设计问题,提出一种新的控制器设计方法.该方法首先利用 Petri 网的状态方程把关于标识向量和 Parikh 向量的不等式约束转变成关于 Parikh 向量的不等式约束,然后基于 Petri 网库所是关于 Parikh 向量的不等式约束的观点构造控制器.最后将该方法与 Iordache 和 Moody 提出的方法作比较,实验结果显示该方法更简单、有效.

关键词: Petri 网;离散事件系统;监控;线性不等式约束;Parikh 向量

中图法分类号: TP311 **文献标识码**: A

1 Introduction

PN models are examined in the DES control synthesis by many researchers due to the advantage of the graphical and distributed representation of the system state and the computational efficiencies[1]. In this paper we

* **WANG Shou-Guang** was born in 1977. He is a Ph.D. candidate at College of Electrical Engineering, Zhejiang University. His current research interests include Petri net theory and application. **YAN Gang-Feng** was born in 1959. He is a professor and doctoral supervisor at College of Electrical Engineering, Zhejiang University. His researches areas are CAD and CAM.

deal with synthesizing the supervisors of DES modeled by PN, which enforce the conjunction of a set of linear inequality constraints involving the Parikh vector and the reachable marking of a PN model. Many researchers[2−6] have studied the logical conjunction of separate linear constraints involving marking only, which have the following form

$$l^T u \le b \tag{1}$$

where $l: P \to Z$ is an $n\times 1$ weight vector, $b \in Z$, Z is the set of integers, n is the number of places, and $u: P \to Z$ is an $n\times 1$ marking vector. A large class of the forbidden marking problems for modeling of the finite resources condition for liveness and deadlock prevention can be specified by the constraints of the form (1)[7,8]. The constraints of the form (1) are called generalized mutual exclusion constraints (GMECs), and it is proved that GMECs can be enforced by a set of places called control places with arcs going to and coming from the plant transitions in Ref.[2]. Moody[3,4] the computed control places based on the concept of place invariants. The controllers designed to enforce constraints of the form (1) in Refs.[2−4] are Petri net, so they are called Petri net controller. While the controllers in Refs.[5,6] are logical controllers. One advantage of representing the controller as a PN is that the computation of the control action is faster, since it does not require separate computation of the control. An additional advantage is that a closed-loop model of the system under control can be built and analyzed for the properties of interest using PN techniques. Because of the above-mentioned advantage of PN controller over logical controller, this paper enforces the constraints of the form (1) using PN controller.

The constraints of the form (1) have been extended in Ref.[9] to the form

$$l^T u + h^T v \le b \tag{2}$$

where $h: T \to Z$ is an $m\times 1$ weight vector, Z is the set of integers, m is the number of transitions, $v: T \to Z$ is an $m\times 1$ Parikh vector and $v(t)$ denotes the number of times the transition t has fired since system initialization. Iordache[9] viewed the Parikh vector term as a marking term, i.e. $v(t)$ was viewed as the marking of a sink place added to the transition t. According to the viewpoint, Iordache transformed the constraints of the form (2) into the constraints of the form (1) that are enforced on the transformed PN, and then designed controllers by using the method presented by Moody et al. The methods presented in Refs.[2−4] for the constraints of the form (1) were based on the concept of place invariants. The computation involves a single matrix multiplication. But for a complex plant with its high dimension incidence matrix, it is troublesome for incidence matrix multiplication. According to this problem, this paper presents a technique that can solve this problem. In this paper, first the constraints of the form (2) are transformed into the constraints involving Parikh vector only, which have the following form

$$h^T v \le b \tag{3}$$

and then the controller is constructed based on the viewpoint that a place can be seen as a linear inequality constraint on Parikh vector. The method presented in this paper has advantage over the method proposed by Iordache and Moody because the incidence matrix of the entire model, which is used and manipulated for the design of controller in Refs.[3,9], doesn't need to be considered. It is proved in Section 4that the method presented in this paper is simpler and more efficient than the one presented by Iordache and Moody by an example, and it holds remarkable advantage especially for large systems.

Assume that all the transitions are controllable because our interest is how to enforce the given constraint (2), not the controllability.

2 Background

A place/transition (P/T) net is a structure $N=(P,T,Pre,Post)$ where P is a set of n places represented by circles; T is a set of m transitions represented by bars; $P \cup T \ne \varnothing$, $P \cap T = \varnothing$; $pre: P \times T \to IN$ is pre-incidence matrix that specifies the arcs directed from places to transitions; *pre*: $P \times T \to IN$ is post-incidence matrix that specifies the arcs directed from transitions to places, where IN={0,1,2,…}. The incidence matrix C of the net is defined as C=*post*-*pre*. A pair of a place p and a transition t is called a *self-loop* if p is both an input and output place of t. A PN is *pure* if it has no *self-loop*s. In the sequel, we assume that the PN is *pure*. We denote the *preset* (*postset*) of a transition $t \in T$ as $\cdot t = \{p \in P \mid pre(p,t) \neq 0\}$ ($t\cdot = \{p \in P \mid post(p,t) \neq 0\}$). Similarly, $\cdot p = \{t \in T \mid post(p,t) \neq 0\}$ ($p\cdot$= $\{t \in T \mid pre(p,t) \neq 0\}$) denotes the *preset* (*postset*) of a place $p \in P$. The *preset* (*postset*) of a set is defined as the union of the *preset* (*postset*) of its elements. A marking of N $u: P \to IN$ is an $n \times 1$ vector. (N, u) is called a net system or a marked net. A transition $t \in T$ is enabled under u, in symbols $u[>t$, iff $\forall p \in \cdot t: u(p) \geq pre(p,t)$ hold. If $u[>t$ holds the transition t may fire, resulting in a new marking u', denoted by $u[t>u'$ with $u' = u + post(\cdot,t) - pre(\cdot,t) = u + C(\cdot,t)$. A *firing sequence* from u_0 is a (possibly empty) sequence of transitions $\sigma=t_1\dots t_k$ such that $u_0[t_1>u_1[t_2>u_2\dots[>t_k$. A marking u is reachable in (N,u_0) iff there exists a firing sequence σ such that $u_0[\sigma>u$. Given a net system, the set of reachable markings is denoted as $R(N,u_0)$. A Parikh vector v of (N,u_0) is a mapping $v: T \to IN$. For transition $t \in T$, $v(t)$ represents the number of times transition t has fired since the initial marking u_0. If marking u is reachable from initial marking u_0, the state equation $u = u_0 + c \cdot v$ is satisfied.

3 Controller Synthesis

We assume that the PN model discussed in the paper is a pure P/T and that its transitions are both controllable and observable.

3.1 The constraints containing the marking only

The system to be controlled is modeled by a PN with n places and m transitions and is known as the plant or process net. The incidence matrix of the process net is C. The control goal is to force the process to obey the constraints of the form (1). Moody et al. transformed this linear inequality constraint into an equality by introducing a nonnegative slack variable into it, and then computes the controller by solving the equality. The next theorem summarizes the construction above:

Theorem 1. Given a net system (N,u_0) with controllable and observable transitions, its incidence matrix C and conjunction of a set of constraints of the form (1). For each constraint $l^T u \leq b$, if $b-l^T u_0 \geq 0$, then a control place p_c to be added to the plant with its incidence matrix $C(p_c)=-l^T C$ and the initial marking $u_0(p_c)=b-l^T u_0$ enforces the constraint $l^T u \leq b$. Furthermore, the controller is maximally permissive.

From Theorem 1, we know that it is necessary for us to express the incidence matrix according to the PN model and operate on the incidence matrix of the plant in order to compute the controller.

3.2 The constraints containing marking vector and Parikh vector

The constraints containing marking vector only have been extended in Ref.[9] to the constraints containing marking vector and Parikh vector.

Given a constraint of the form (2) on a net system (N,u_0), Iordache[9] presented a transformation approach in which the constraints of the form (2) is transformed into the constraints of the form (1). We illustrate the idea of Iordache's transformation on an example.

Example 1. Consider the PN of Fig.1, and assume that we desire to enforce the following constraint

$$u(p_1)+v(t_4) \leq 2 \tag{4}$$

Iordache transformed the net as in Fig.2 by adding a net a sink place p_6 with its initial marking $u_0(p_6)=0$, the constraints (4) can be written without referring to v:

$$u(p_1)+u(p_6) \leq 2 \tag{5}$$

So Iordache reduced the problem to the supervisory synthesis problem for constraints of form (1) on the transformed net, and then designed the controller using Theorem 1.

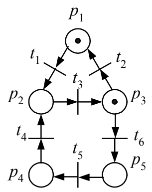

Fig.1 A PN

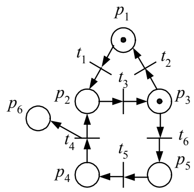

Fig.2 A transformed PN of Fig.1

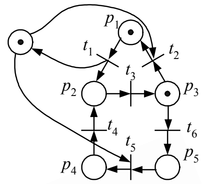

Fig.3 PN of Fig.1 with a controller

3.3 Description of designing the controller for the constraints containing marking vector and Parikh vector

Given a net system (N,u_0), $\forall p \in P$, we have the following state equation

$$u(p) = u_0(p) - \sum_{t \in p\cdot} pre(p,t) \cdot v(t) + \sum_{t \in \cdot p} post(p,t) \cdot v(t) \tag{6}$$

So we can transform the constraints of the form (2) into the constraints of the form (3) using the state equation (6).

Example 2. Consider the PN of Fig.1 and the constraint (4). According to Eq.(6), we can transform the constraint (4) into the following constraint, which involving Parikh vector only

$$u_0(p_1) - \sum_{t \in p_1\cdot} pre(p_1,t) \cdot v(t) + \sum_{t \in \cdot p_1} post(p_1,t) \cdot v(t) + v(t_4) \le 2$$

that is

$$-v(t_1)+v(t_2)+v(t_4)\le 1 \tag{7}$$

Because $u(p) \ge 0$, according to Eq.(6), we have

$$\sum_{t \in p\cdot} pre(p,t) \cdot v(t) - \sum_{t \in \cdot p} post(p,t) \cdot v(t) \le u_0(p) \tag{8}$$

So we have the following proposition

Proposition 1. Every place of a PN can be seen as a control place enforcing a single inequality of the form (3).

Proposition 1 shows that a constraint of the form (3) can been enforced by adding a control place to the plant if $b \ge 0$. The significance of Proposition 1 is that it provides a way to design controller for the constraints of the form (3) since every place of a PN can been seen as a control place enforcing a single inequality of the form (3).

For the convenience of the description, we make the following definitions before presenting the detailed steps of designing controller.

Definition 1. Given a net system (N,u_0) and a constraint $l^T u \le b$, a place $p \in P$ is said to be a constrained place for the constraint $l^T u \le b$ if $p \in \|l\|$, where $\|l\|=\{p \in P: l(p) \ne 0\}$ denotes the support of the vector l.

According to Definition 1, we can transform the constraints of the form (1) to the following form

$$\sum_{p \in \|l\|} l(p) u(p) \le b \tag{9}$$

Example 3. Consider the PN of Fig.1. The objective is to control the net so that places p_1 and p_2 never contain more than on token, i.e. we wish to enforce the constraint $u(p_1)+u(p_2)\le 1$.

According to Definition 1, we have $\|l\|=\{p_1,p_2\}$.

Definition 2. Given a net system (N,u_0) and a constraint $h^T v \le b$, a transition $t \in T$ is said to be a constrained transition for the constraint $h^T v \le b$ if $t \in \| h \|$ where $\| h \| = \{t \in T : h(t) \ne 0\}$ denotes the support of the vector h. We define $\| h^+ \| = \{t \in T : h(t) > 0\}$, and $\| h^- \| = \{t \in T : h(t) < 0\}$. Obviously, we have $\| h \| = \| h^+ \| \cup \| h^- \|$.

According to Definition 2, we can transform the constraints of the form (3) to the following form

$$\sum_{t\in\|h^+\|} h(t)v(t) - \sum_{t\in\|h^-\|} |h(t)|\, v(t) \le b \tag{10}$$

Example 4. Consider the PN of Fig.1, and the constraint (7). According to Definition 2, we have $\|h\|=\{t_1,t_2,t_4\}$, $\|h^+\|=\{t_2,t_4\}$ and $\|h^-\|=\{t_1\}$.

Assume that there exists a controller that enforces the plant to obey the constraints of the form (2). With the above definitions, the design method of the control place conforming to the constraint of the form (2) is summarized as the following algorithm.

Algorithm 1.

1) Transform the constraint of the form (2) to the constraint of the form (3)

Given a constraint of the form (2) as follows

$$l^T u + h^T v \le b \tag{11}$$

According to Eq.(6), the constraint (11) is transformed to

$$h'^T v \le b' \tag{12}$$

where

$$b' = b - \sum_{p\in\|l\|} l(p)u_0(p) \tag{13}$$

and

$$h'(t) = h(t) + \sum_{p\in\|l\|\cap t\cdot} l(p)\,post(p,t)\cdot v(t) - \sum_{p\in\|l\|\cap\cdot t} l(p)\,pre(p,t)\cdot v(t) \tag{14}$$

Constraint (12) can be written as

$$\sum_{t\in\|h'^+\|} h'(t)v(t) - \sum_{t\in\|h'^-\|} |h'(t)|\, v(t) \le b' \tag{15}$$

2) Design the controller for constraint (12)

A) For each $t\in\|h'^+\|$, draw an arc from the controller place p_c to transition t, and set the weight of the arc $h'(t)$, i.e. $pre(p_c,t)=h'(t)$.

B) For each $t\in\|h'^-\|$, draw an arc from the transition t to controller place p_c, and set the weight of the arc $|h'(t)|$, i.e. $pre(p_c,t)=|h'(t)|$.

C) Let the initial marking of the control place $u_0(p_c)=b'$.

We illustrate the above algorithm by using the following example.

Example 4. Consider the PN of Fig.1, and assume that we desire to enforce the constraint of the form (2) $u(p_1)+v(t_5)\le 2$. By Algorithm 1, first we can transform this constraint into the following constraint of the form (3)

$$-v(t_1)+v(t_2)+v(t_5)\le 1.$$

We have $\|h'\|=\{t_1,t_2,t_5\}$, $\|h'^+\|=\{t_2,t_5\}$ and $\|h'^-\|=\{t_1\}$. Then we draw arcs between the transitions in $\|h'\|$ and the control place p_c. Finally we set the initial marking of the control place $u_0(p_c)=b'=1$.

The plant with the addition of a controller is shown in Fig.3. The controller is the same as the one computed by the technique proposed by Iordache, but our method need not consider the entire plant, just consider part of the entire plant in the above example, because the incidence matrix of the entire plant and its operation, which are used in Iordache's method, are not be used in our method. So our method is simper and more efficient in computation than the methods presented in Refs.[3,9]. When the plant is a large system, the advantage of this design method is more obvious, which will be illustrated in the example in Section 4.

Lemma 1. Given a system (N,u_0) and the constraint $h^T v\le b$, if $b<0$, then there doesn't exist a control place for the constraint.

Proof. When the plant is in the initial state, i.e. v=0, the constraint $h^T v\le b$ is not satisfied. So the constraint cannot be enforced.

Proposition 2. Given a system (N,u_0) and the constraint $l^T u + h^T v \le b$. If $b - \sum_{p\in\|l\|} l(p)u_0(p) < 0$, then there doesn't exist a control place for the constraint.

Proof. The proof follows immediately from Algorithm 1 and Lemma 1.

Theorem 2. Given a system (N,u_0) and the constraint $l^T u+h^T v\le b$. If $b - \sum_{p\in\|l\|} l(p)u_0(p) \ge 0$, then a control place p_c, which is designed according to Algorithm 1, enforces the constraint $l^T u+h^T v\le b$.

Proof. According to Algorithm 1, the marking of the control place p_c satisfies

$$u(p_c)=b'-h'^T v \tag{16}$$

Because $b' = b - \sum_{p\in\|l\|} l(p)u_0(p)$, and $h'(t) = h(t) + \sum_{p\in\|l\|\cap t\cdot} l(p)post(p,t)\cdot v(t) - \sum_{p\in\|l\|\cap\cdot t} l(p)pre(p,t)\cdot v(t)$. The equality (16) can be written as

$$u(p_c)=b-l^T u-h^T v \tag{17}$$

This concludes the proof.

Theorem 3. The controller derived using Algorithm 1 is maximally permissive in that it forces the set of constraints of the form (2) to be obeyed, while allowing any action that is not directly forbidden by the constraints.

Proof. The PN enabling condition indicates that a transition is inhibited only if its firing would cause the marking of any of its input places to become negative. Thus a controller place only acts to inhibit a transition when the firing would cause $u(p_c)<0$. According to equality (17), we can know that if $u(p_c)<0$, then $l^T u>b$. The controller will only act to inhibit in situations where the firing of a transition would cause a direct violation of the constraint inequality.

4 Example

In this section, the example of An Automated Guided Vehicle (AGV) coordination system is used to illustrate the efficiency of the method presented in this paper. The example originally appeared in Ref.[5] and has been studied intensively in the area of DES[3,5,10,11]. AGV includes three workstations, two part-receiving stations and one completed parts station. There are five AGVs which can transport material between the stations. To avoid conflict within shared zones, which are shown as the shaded regions in the PN model of Fig.4, it is specified that AGVs in this system are to be controlled so that any zone is occupied by no more than one AGV at any time.

The control objective can be written as the following constraints

$$\sum_{p\in Zone1} u(p) \le 1 \tag{18}$$

$$\sum_{p\in Zone2} u(p) \le 1 \tag{19}$$

$$\sum_{p\in Zone3} u(p) \le 1 \tag{20}$$

$$\sum_{p\in Zone4} u(p) \le 1 \tag{21}$$

For constraint (18), according to Algorithm 1, first transform it to the following constraint

$$-v_{11}+v_{12}-v_{13}+v_{14}+v_{15}-v_{16}-v_{17}+v_{18}\le 1 \tag{22}$$

then draw the arcs between the transitions in $\{t_{11},t_{12},t_{13},t_{14},t_{15},t_{16},t_{17},t_{18}\}$ and the control place p_{c1}, and finally set $u_0(p_{c1})=1$. The other control places p_{c2}, p_{c3} and p_{c4} associated to the corresponding constraint can be synthesized with the same algorithm and the detailed procedure is omitted. The controller consists of four control places p_{c1}, p_{c2}, p_{c3} and p_{c4} that are connected to the system model as shown in Fig.5.

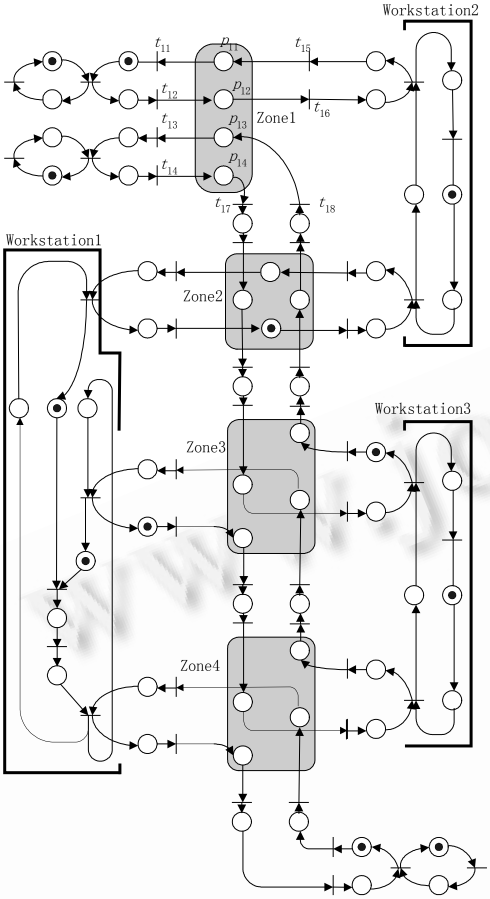

Fig.4 The automated guided vehicle PN

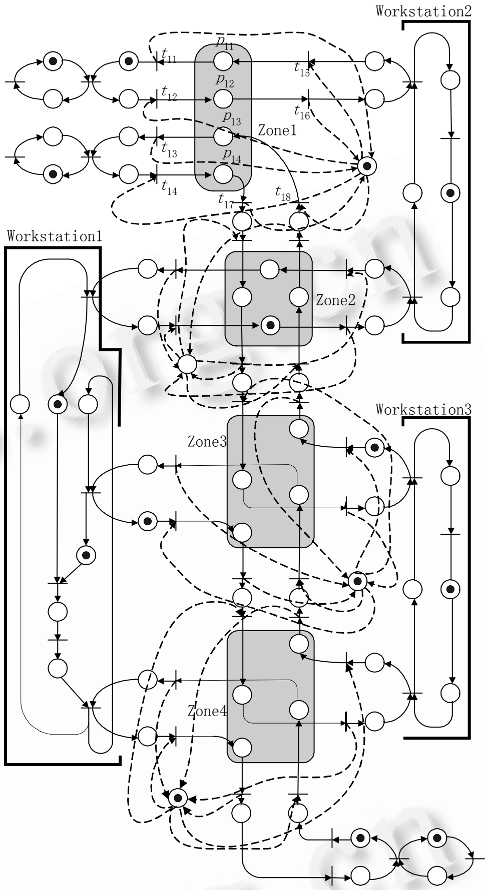

Fig.5 The controlled AGV PN

The controller obtained using Algorithm 1 is identical to the controller of Moody and Iordache based on place invariant[3,9]. But the method in this paper is simper and more efficient than the method proposed by Moody and Iordache. Our method has advantage over the method by Moody and Iordache because the incidence matrix of the entire model, which is used and manipulated for the design of controller in Refs.[3,9], doesn't need to be considered in Algorithm 1. The advantage is obvious when the system to be controlled is large and complex. Consider the AGV example again. The PN model of the AGV system has 64 places and 53 transitions and its incidence matrix is a 64×53 dimension matrix, which has 3392 entries. It is very troublesome for the incidence matrix multiplication. But for each constraint, only 8 entries are needed to participate in operation for solving the control place in this paper (see the constraint (22)). So our method holds remarkable advantage over the method in Refs.[3,9] in this example.

5 Conclusions

This paper has discussed the issue of synthesizing PN controller, which enforces the conjunction of a set of linear inequalities on the reachable marking vector and the Parikh vector of the plant modeled by PN. A comparison with other techniques has been carried out to prove the effectiveness of the proposed approach.

References:

[1] Murata T. Petri nets: Properties, analysis and applications. Proc. of the IEEE, 1989,77(4):541−580.

[2] Giua A, DiCesare F, Silva M. Generalized mutual exclusion constraints on nets with uncontrollable transitions. In: Proc. of the IEEE Int'l Conf. on SMC. 1992. 975−979. http://intl.ieeexplore.ieee.org/Xplore/DynWel.jsp

[3] Yamalidou K, Moody JO, Lemmon M, Antsaklis PJ. Feedback control of Petri nets based on place invariants. Automatica, 1996,32(1):15−18.

[4] Moody J, Antsaklis PJ. Petri net supervisors for DES with uncontrollable and unobservable transitions. IEEE T-AC, 2000,45(3): 462−472.

[5] Holloway LE, Krogh BH. Synthesis of feedback logic for a class of controlled Petri nets. IEEE T-AC, 1990,35(5):514−523.

[6] Holloway LE, Guan X, Zhang L. A generalization of state avoidance policies for controlled Petri nets. IEEE T-AC, 1996,41(6): 804−816.

[7] Iordache UV, Moody JO, Antsaklis PJ. Synthesis of deadlock preventions supervisors using Petri nets. IEEE T-AC, 2002,18(1): 59−68.

[8] Iordache UV, Antsaklis PJ. Design of T-liveness enforcing supervisors in Petri nets. IEEE T-AC, 2003,48(11):1962−1974.

[9] Iordache UV, Antsaklis PJ. Synthesis of supervisors enforcing general linear constraints in Petri nets. IEEE T-AC, 2003,48(11): 2036−2039.

[10] Krogh BH, Holloway LE. Synthesis of feedback logic for discrete manufacturing systems. Automatica, 1991,27(4):641−651.

[11] Li Y, Wonham WU. Control of vector discrete-event systems II-The base model. IEEE T-AC, 1994,39(3):512−531.