第29卷 第5期 自 动 化 学 报 Vol. 29, No. 5
2003年9月 ACTA AUTOMATICA SINICA Sept., 2003

Combined Petri Net Controller for Discrete Event Systems[1)]

WU Wei-Min DONG Li-Da WANG Xiao SU Hong-Ye CHU Jian
(*National Laboratory of Industrial Control Technology & Institute of Advanced Process Control, Zhejiang University, Hangzhou* 310027)
(E-mail: wmwu@iipc.zju.edu.cn)

Abstract This paper addresses the problem of controller synthesis of discrete event systems (DES) modeled by controlled Petri nets (PN) with uncontrollable transitions. The combined PN controller for DES proposed has both the virtues of the mapping controller and the compiled one, that is, it combines the advantages of the compiled controller in the aspect of obtaining and tracing system state and the mapping controller in the aspect of implementing of control action. Examples are used to illustrate the design method of the combined PN controller.

Key words Discrete event systems, Petri nets, uncontrollable transition, combined controller

1 Introduction

Petri net (PN) is an important tool to synthesize discrete event systems (DES) due to its advantages such as graphical, distributed representation of the system state and computational efficiencies. From the control standpoint, the controller or supervisor of DES can be distinguished between the mapping one, whose control law is a function computed after each new event generated by the system, and the compiled one, whose control law is presented as a DES structure [1]. Various mapping controllers have been designed in [2~4] while synthesis methods of compiled controller can be found in [5,6].

The ordinary PN model of DES considered here is able to model both resource conflict and process synchronization. The restrictions against PPC (precedence path condition) and PPIC (precedence path input condition)[2] are relaxed in this paper. In addition, the method presented here does not need any non-convex constraint transformation in order to deal with the firing of uncontrollable transitions. The combined PN controller designed in this paper is a mapping one but it exploits the advantages of compiled controller.

2 Foundation of CtlPN and Control Constraint

A controlled ordinary PN (CtlPN) is defined as a six-tuple $G=(P,T,E,C,B,m)$, where P is a finite set of state places, T is a finite set of transitions, $P\cap T=\varnothing$, $E\in(P\times T)\cup(T\times P)$ is a set of directed arcs connecting state places and transitions, C is the finite set of control places, $B\subseteq(C\times T)$ is the set of directed arcs associating control places with transitions, and m: $P\rightarrow Z$ is the marking of the places (Z is the set of nonnegative integers). It is assumed in this paper that one transition has at most one connected control place and one control place is exactly connected by one transition. The transitions connected by control place are controllable and the controllable transition set is denoted by T_c, otherwise uncontrollable, and the uncontrollable transition set is represented as T_u. The places, transitions, control places and marking are graphically represented by circles, bars, squares and dots, respectively, as shown in Fig. 1.

1) Supported by the National Outstanding Youth Science Foundation of P. R. China (60025308), the Key Project of CIMS under the National High Technology Research and Development Program of P. R. China (863 Program) (2001AA413020), Teaching and Research Award Program for Outstanding Young Teachers in Higher Education Institutions of Ministry of Education, P. R. China and Doctor Degree Program Foundation of P. R. China (20020335103). The preliminary version of this paper has been published in the Proceeding of 15th IFAC Word Congress
Received June 25, 2002; in revised form March 3, 2003
收稿日期 2002-06-25; 收修改稿日期 2003-03-03

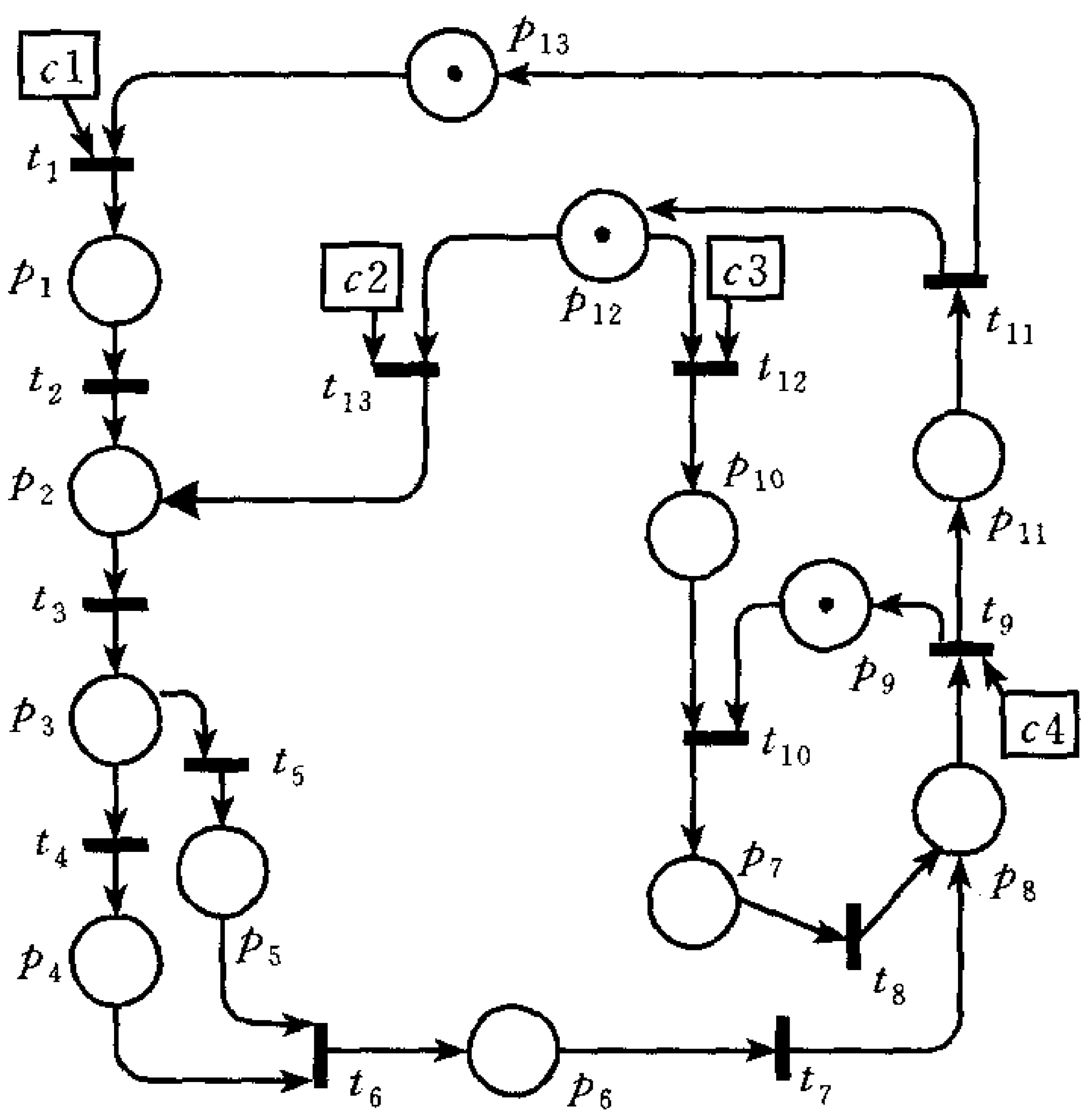

Fig. 1 A controlled PN

For a transition $t \in T, t$ is called an input transition to p if arc $(t, p) \in E$. The input transition set of p is denoted by ${}^{(t)}p$. Similarly, the input place set of transition t is denoted by ${}^{(p)}t$, and output sets can be defined as $p^{(t)}$ and $t^{(p)}$. The notation $c^{(t)}$ represents the only transition associated with the control place c, and ${}^{(c)}t$ denotes the only control place associated with t.

A control u: $C \rightarrow (0,1)$ assigns a binary token count to each control place. The set of all controls is denoted as U. For two controls u_1 and u_2, $u_1 \geqslant u_2$ holds if $u_1(c) \geqslant u_2(c)$ for all $c \in C$, and $u_1 > u_2$ holds if $u_1(c) \geqslant u_2(c)$ and $u_1(c) > u_2(c)$ for at least one $c \in C$. A control u_1 is more permissive than another control u_2 if $u_1 > u_2$. The control $u_{\text{one}}, u_{\text{one}}(c) = 1$ for all $c \in C$, is the most permissive, and the control $u_{\text{zero}}, u_{\text{zero}}(c) = 0$ for all $c \in C$, is the least permissive.

A transition $t \in T$ is said to be state enabled under marking m if $m(p) \geqslant 1$ for all $p \in {}^{(p)}t$. A transition $t \in T_c$ is said to be control enabled (disabled) if $m({}^{(c)}t) = 1$ (0). Conventionally, all the transitions in T_u are assumed to be control enabled. A state enabled and control enabled transition $t \in T$ is said to be enabled.

The control constraint enforced in this paper is a linear marking constraint, which has the following form

$$\sum_{i=1}^{n} l_i m(p_i) \leqslant b \tag{1}$$

where coefficient l_i is a non-negative integer, $m(p_i)$ is the marking of place p_i, b is a positive integer constant and n is the number of places in the net. For convenience, the notation $M_C(m)$ is sometimes used to denote the value of left side of (1) under marking m. Let $R_\infty(M_C(m))$ be the set of possible values of $M_C(m')$ under any reachable marking m' from m, and $\max[M_C(m)]$ be the maximal in the set $R_\infty(M_C(m))$.

A marking m is said to be admissible if $\max[M_C(m)]$ is not bigger than b under u_{zero} and the set of admissible marking is denoted as Ω. The control policy U is a state feedback policy that maps every $m \in \Omega$ to a set of controls $U(m)$. For two control policies U_1 and U_2, U_1 is said to be more permissive than U_2, denoted as $U_1 > U_2$, if $U_1(m) \supseteq U_2(m)$ for all $m \in \Omega$ and $U_1(m') \supset U_2(m')$ for some $m' \in \Omega$.

Definition 1. The place in the constraint inequality (1) is named as constrained place. The entire constrained places constitute constrained place set, denoted by C_p, that is, $C_p = \{p = p_i \mid \sum_{i=1}^{n} l_i m(p_i) \leqslant b \text{ for } l_i \neq 0\}$.

Definition 2. The set of input transitions for the entire constrained place set C_p is said to be input constrained transition set, denoted by ${}^{(t)}C_p$, that is, ${}^{(t)}C_p=\{t\mid t\in {}^{(t)}p \text{ for } p\in C_p\}$.

Definition 3. The set of output transitions for the entire constrained place set C_p is said to be output constrained transition set, denoted by $C_p^{(t)}$, that is, $C_p^{(t)}=\{t\mid t\in p^{(t)} \text{ for } p\in C_p\}$.

Definition 4. The set of transitions denoted by CC_t is said to be common constrained transition set, if its entry t satisfies $t\in {}^{(t)}C_p\cap C_p^{(t)}$.

Definition 5. Given the input constrained transition set ${}^{(t)}C_p$, the set ${}^{(t)}C_{pure-t}={}^{(t)}C_p-CC_t$ is said to be pure input constrained transition set.

Definition 6. Given the output constrained transition set $C_p^{(t)}$, the set $C_{pure-t}^{(t)}=C_p^{(t)}-CC_t$ is said to be pure output constrained transition set.

According to the controllability of transitions, ${}^{(t)}C_{pure-t}$ is divided into two subsets ${}^{(t)}C_{c-pure-t}$ and ${}^{(t)}C_{u-pure-t}$, where ${}^{(t)}C_{c-pure-t}={}^{(t)}C_{pure-t}\cap T_c$ and ${}^{(t)}C_{u-pure-t}={}^{(t)}C_{pure-t}\cap T_u$, respectively.

Example 1. Consider the Petri net illustrated in Fig. 1. Assume the net satisfies the following constraint in its evolution:

$$2m(p_2)+m(p_4)+m(p_7)\leqslant 3 \tag{2}$$

Then, $C_p=\{p_2,p_4,p_7\}$; ${}^{(t)}C_p=\{t_2,t_{13},t_4,t_{10}\}$, $C_p^{(t)}=\{t_3,t_6,t_8\}$; ${}^{(t)}C_{pure-t}={}^{(t)}C_p$, ${}^{(t)}C_{c-pure-t}=\{t_{13}\}$, ${}^{(t)}C_{u-pure-t}=\{t_2,t_4,t_{10}\}$ and $C_{pure-t}^{(t)}=C_p^{(t)}$. Note that $CC_t=\phi$ holds in this example.

3 Construction of Monitor

3.1 Influence Path

The concept of influence path (IP) here is different from the ones in [2,4]. IP, which does not exist in the plant, can be regarded as a copy of the precedence path (PP) in the sense of construction. A path $\pi=(t_1p_1t_2p_2\cdots t_{n-1}p_{n-1}t_n)$ defined in this paper is a string of nodes such that both the beginning and end nodes are transitions and $p_i\in t_i^{(p)}\cap {}^{(p)}t_{i+1}$ for $1\leqslant i\leqslant n-1$. The expression '$x\in$ (or $\notin$) π' means that x is (or is not) a node in π. A subpath of π is denoted by $\pi(x_i,x_j)$, where x is a node and $1\leqslant i<j\leqslant n$.

Definition 7. Given an uncontrollable input constrained transition $t\in {}^{(t)}C_{u-pure-t}$, precedence path π_t is a path such that: $t_1=t$, t_i is uncontrollable for $1\leqslant i\leqslant n-1$ and t_n is controllable.

A PP π_t for t has only one controllable transition t_n, and t_n is called the (unique) controllable transition of π_t. The case that t_n is uncontrollable is not considered here since this case will lead to uncontrollability of the plant[4] or has no influence on decision of the control policy.

For a given transition $t\in {}^{(t)}C_{u-pure-t}$, it may have more than one PP. These paths are joined together at some places or transitions, and these places or transitions are called the joining nodes. The set of precedence paths for t is denoted as Π_t. Let $\Gamma_t=\{t_n\mid t_n\in\pi_t$ for $\pi_t\in\Pi_t$, t_n is controllable$\}$ be the controllable transition set with respect to t. Let $\Gamma_t(s)$ be the subset of Γ_t in which each transition is state enabled, i. e., $\Gamma_t(s)=\{t_n\mid t_n\in\Gamma_t, t_n$ is state enabled$\}$. $x\in(\notin)\Pi_t$ if $x\in(\notin)\pi_t$ for $\pi_t\in\Pi_t$. The notation $\pi_t(t_n)$ is used to represent the PP whose controllable transition is t_n. Note that there are no restrictions against the PPC and the PPIC in the definitions of PP and PP set. For a PP π_t, when $\exists p\in\pi_t, p\in C_p$, the PP violates PPC.

Example 2. In Fig. 1, there are three PP sets: $\Pi_{t2}=\{\pi_{t2}(t_1)\}$, $\Pi_{t4}=\{\pi_{t4}(t_1),\pi_{t4}(t_{13})\}$, $\Pi_{t10}=\{\pi_{t10}(t_{12}),\ \pi_{t10}(t_9)\}$, where $\pi_{t2}(t_1)=(t_2p_1t_1)$, $\pi_{t4}(t_1)=(t_4p_3t_3p_2t_2p_1t_1)$, $\pi_{t4}(t_{13})=(t_4p_3t_3p_2t_{13})$, $\pi_{t10}(t_{12})=(t_{10}p_{10}t_{12})$ and $\pi_{t10}(t_9)=(t_{10}p_9t_9)$. Note that the constrained place $p_2\in\Pi_{t4}$ and the controllable transitions t_{13} and t_{12} are in conflict, so this example does not satisfy PPC and PPIC.

Definition 8. Given an uncontrollable input constrained transition $t\in {}^{(t)}C_{u-pure-t}$, the influence path set is constructed as follows:

1) Draw a copy of the transition t and the joining transitions t_k (denoted as t_{ip} and t_{ip-k} for t and $t_k \in \pi$, respectively) in the paths of Π_t, and the copied transitions are arranged in the same order as the originals. The transitions t and t_k are called the original of t_{ip} and t_{ip-k}, respectively.

2) Draw a place between the two adjacent copied transitions. Note that the originals of the two transitions should be in the same PP. A place should also be drawn between the controllable transitions in Γ_t and its neighboring copied transitions.

3) Connect the adjacent nodes obtained above by arcs from the controllable transitions in Γ_t to t_{ip} in the same direction of the corresponding PP.

The transitions t_{ip} and t_{ip-k} in IP are associated with the so-called 'always occurring' events (denoted as e)[7]. These transitions are fired as soon as they are enabled. The set of IP for t_{ip} (corresponding to t) is denoted as ∇_{tip} and the notation $\pi_{tip}(t_n)$ represents the IP with controllable transition t_n. Similar to the case of PP set, $x \in \nabla_{tip}$ indicates that x lies in the IP set. Transition t_{ip} is called influence transition and the set of influence transition is denoted as T_{ip}.

Example 3. Fig. 2 illustrates the IP sets $\nabla_{t14} = \{\pi_{t14}(t_9), \pi_{t14}(t_{12})\}$ and $\nabla_{t15} = \{\pi_{t15}(t_1), \pi_{t15}(t_{13})\}$, which correspond to Π_{t10} and Π_{t4}, respectively, where $\pi_{t14}(t_9) = (t_{14}\, p_{14}\, t_9)$, $\pi_{t14}(t_{12}) = (t_{14} p_{15} t_{12})$, $\pi_{t15}(t_1) = (t_{15} p_{16} t_1)$ and $\pi_{t15}(t_{13}) = (t_{15} p_{16} t_{13})$. The IP set corresponding Π_{t2} is omitted since it is a subset of ∇_{t15}. Note that p_{16} and t_{14} are joining nodes.

The following lemma indicates that the firing of influence transitions represents the maximal influence of the uncontrollable transitions on the control constraint.

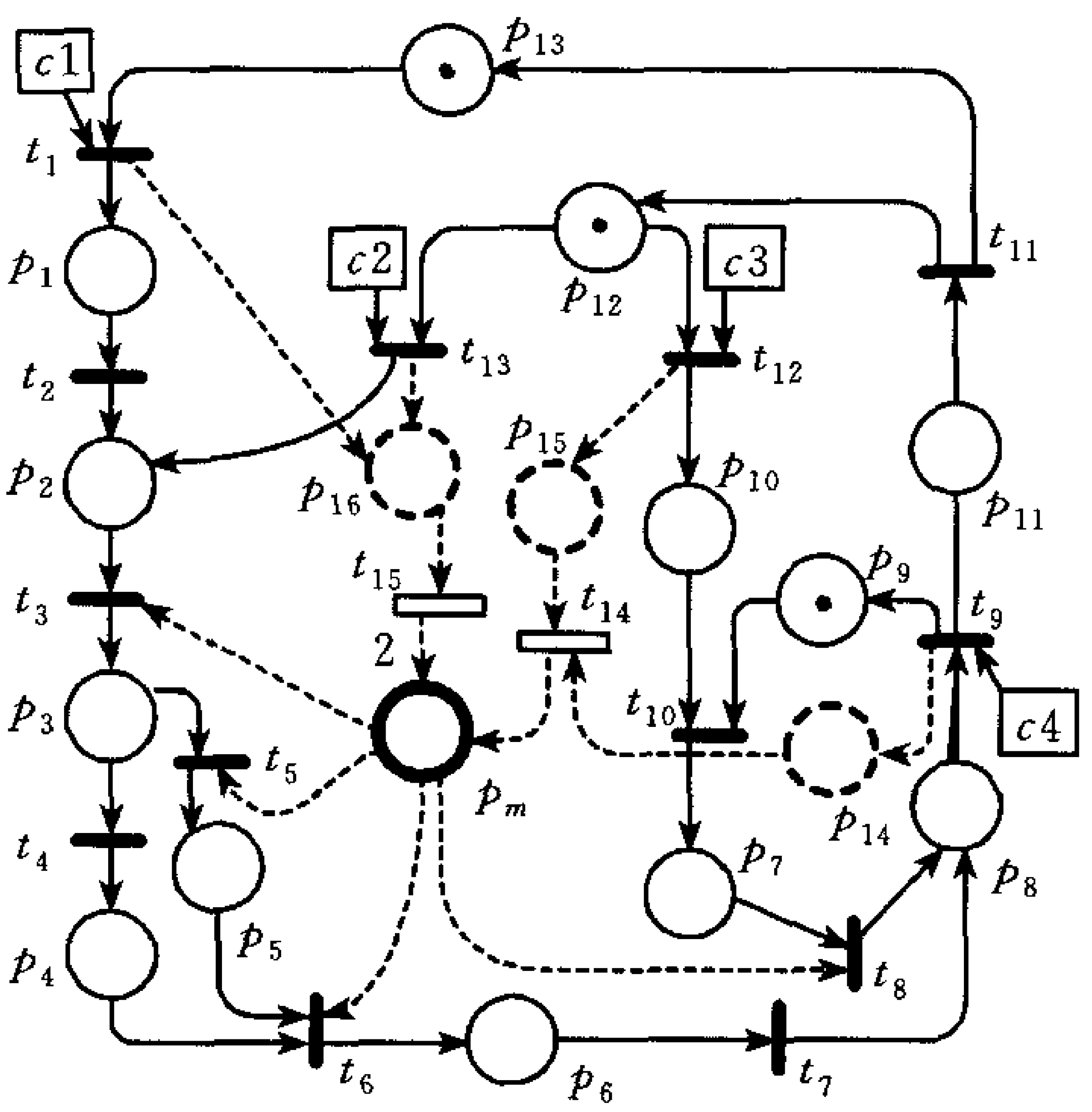

Fig. 2 The net of Fig. 1 with a monitor

Lemma 1. For $t \in {}^{(t)}C_{u-pure-t}$ and its corresponding influence transition t_{ip}, if there are no conflicts and no initial tokens in Π_t, t and t_{ip} have the same firing times in the evolution of the system.

Proof. By the definitions of PP and IP, both of the markings of $p \in \Pi_t$ and $p_{ip} \in \nabla_{tip}$ can be influenced only by the same controllable transition $\tau \in \Gamma_t$. Once one or several transitions in Γ_t fire, the same amount of tokens will enter Π_t and ∇_{tip}, and reach ${}^{(p)}t$ and ${}^{(p)}t_{ip}$, respectively. For any joining transition $\tau_k \in \Pi_t$ and its corresponding joining transition $\tau_{ip-k} \in \nabla_{tip}$, $|{}^{(p)}\tau_k| = |{}^{(p)}\tau_{ip-k}|$, i. e., the number of PP joined at τ_k is equal to that of IP joined at τ_{ip-k}. Suppose that the uncontrollable transitions in the PP set are also associated with the 'always occurring' event e, which is the same as the original case in the sense of evaluating the firing times of t. Then $m(p)=m(p_{ip})$ at any time for $p\in{}^{(p)}\tau_k$ and $p_{ip}\in{}^{(p)}\tau_{ip-k}$. It is also true for t and t_{ip}. Thus the lemma is proved. □

3.2 Monitor

Definition 9. Given any PP set Π_t, and a PP $\pi_t(t_n)\in\Pi_t$, the set is said to satisfy the transition conflict condition (TCC) if the following statements are true:

1) For any two transitions $t_1\in\pi_t(t_n)$ and $t_2\notin\Pi_t$ such that t_1 and t_2 are in conflict and any joining transition $t_3\in\pi_t(t,t_1)$, there are no transitions in any sub-path $\pi_t(t_3,t_4)$ that are in conflict with $t_5\notin\Pi_t$, where $t_4\in\{\Gamma_t-t_n\}$.

2) For any conflict, if not all the transitions involved in it are controllable, there is at most one transition in some PP.

The TCC ensures that the firing of any transition $\tau\notin\Pi_t$, $\tau\in p^{(t)}$ for $p\in\Pi_t$ will result in the reduction of same firing times of t. Note that a conflict in which all the involved transitions are controllable does not violate the TCC. The PP set considered in this paper is assumed to satisfy the TCC.

Algorithm for construction of the monitor.

Step 1. For each $t\in C^{(t)}_{pure-t}$, draw an arc from the monitor place p_m to t. The weight function w of the arc satisfies: $w=\sum_{i=1}^{n}l_i$, where l_i is the coefficient of $p_i\in{}^{(p)}t$, $p_i\in C_p$.

Step 2. For each $t\in{}^{(t)}C_{pure-t}$

1) if $t\in{}^{(t)}C_{c-pure-t}$, draw an arc from t to p_m, else if $t\in{}^{(t)}C_{u-pure-t}$, the beginning of the arc is the copy of t, i.e., t_{ip}. The weight of the added arc is also w determined in the last step, but l_i is the coefficient of $p_i\in t^{(p)}$, $p_i\in C_p$.

2) if there exists some transitions $\tau\notin\Pi_t$ such that τ and $\tau'\in\Pi_t$ are in conflict, draw an arc with weight of w from p_m to τ.

Step 3. For each $t\in CC_t$, draw an arc between p_m and t, the weight function w of the arc satisfies: $w=|\omega|$, $\omega=\sum_{i=1}^{n}l_i-\sum_{j=1}^{n}l_j$, where l_i and l_j are the coefficients of $p_i\in{}^{(p)}t$ and $p_j\in t^{(p)}$, respectively, $p_i,p_j\in C_p$ and $|\omega|$ denotes the absolute value of ω. If $\omega<0$ ($\omega>0$), let p_m be the output (input) place of t, and if $\omega=0$, there is no arc between p_m and t at all.

Step 4. Calculate the initial marking of monitor.

1) The monitor place p_m: $m_0(p_m)=\sum_{i=1}^{n}l_i m_0(p_i)$.

2) The subnet constituted by IP sets: for any place p_{ip} in the subnet, its initial marking is calculated according to the equation $m_0(p_{ip})=\sum m_0(p)$, where $p\in\pi_t(t_i,t_j)$, t_i and t_j are any originals in the plant of $t_{ip-i}\in p_{ip}^{(t)}$ and $t_{ip-j}\in{}^{(t)}p_{ip}$, respectively.

The above algorithm does not consider the case when PP violates PPC except for the case that the constrained places are connected by the transitions in CC_t. In the case of PPC, there is a slight modification for the algorithm. For simplicity, only the case that the transitions in a PP have exactly one output constrained place is considered. Suppose $\tau\in\pi_t$, $p_j\in\tau^{(p)}$ is a constrained place. The following remark 1 represents the corresponding algorithm.

Remark 1. In this case, weight w of the arc from t_{ip} to the monitor place satisfies $w=\max\{l_j,l_k\}$, where l_j and l_k are the coefficients of $p_j\in\tau^{(p)}$ and $p_k\in t^{(p)}$, respectively, $p_j,p_k\in C_p$. There is no arc between the monitor place and τ_{ip} (τ_{ip} exists when τ is a joining transition). If $l_j>l_k$, there is an arc from p_m to $p_j^{(t)}$ with the weight of l_j-l_k, else the arc is omitted. If τ is controllable, the arc from τ to p_m is also omitted since it has already been treated as one element in Γ_t. The arc from p_m to $p_k^{(t)}$ is designed in the same way as Step 1.

Remark 2. In Step 3, when t is an uncontrollable transition, it is assumed that $\omega>0$. If

$\omega<0$, this case should be treated as in the normal case of violating PPC mentioned in Remark 1 above. This assumption ensures that there are no uncontrollable input transitions to p_m.

The basic idea behind the above algorithm is that the monitor is constructed in such a way that it will get or lose the same tokens as the constrained places will do when the related transitions fire. Step 2. 1 ensures that the monitor can track the set of marking for which the control constraint (1) can be violated due to uncontrollable firing sequences. To compensate for the excessive firing of the influence transition caused by the conflict, Step 2. 2 also connects an arc from the monitor to the conflicted transitions that are not in the PP.

Example 4. By the construction algorithm, the monitor shown in Fig. 2 is constructed to track the state of given constraint (2). Note that the arcs between p_m and t_{15} and t_3 are designed according to Remark 1 and the weight of arc (t_{15}, p_m) is 2.

By the algorithm, the following lemma can be obtained, which claims that the number of tokens resided in the monitor place is the maximum that C_p can reach under the control of u_{one}.

Lemma 2. For any marking m, $\max[M_C(m)]=m(p_m)$.

From Lemma 2, we have the following corollary.

Corollary 1. For any marking $m\in\Omega$, $\max[M_C(m)]\leqslant b$ if and only if $m(p_m)\leqslant b$.

The following lemma states that the monitor has no influence on its output transitions.

Lemma 3. The monitor is incapable of disabling any already enabled transition in the plant.

Proof. Suppose t is an already enabled output transition of the monitor place p_m and $p_i\in{}^{(p)}t\cap C_p$. Then $t\in C^{(t)}_{pure-t}$ or $t\in CC_t$. By the algorithm, the weight function of the arc from p_m to t is w. Obviously, $m_0(p_m)\geqslant w$. For the case of $m(p_m)$, p_i must be marked since t is enabled, and p_m obtains $\sum_{i=1}^{n} l_i$ tokens when the transitions in ${}^{(t)}p_i$ or the corresponding influence transitions fire, where l_i is the coefficient of $p_i\in{}^{(p)}t\cap C_p$. Note that $\sum_{i=1}^{n} l_i=w$ when $t\in C^{(t)}_{pure-t}$ or $\sum_{i=1}^{n} l_i>w$ when $t\in CC_t$. The additional tokens suffice to make the inequality $m(p_m)\geqslant w$ hold. Thus, the monitor will not disable the already enabled transitions in the plant. □

Remark 3. If the transition $\tau\in{}^{(t)}p_i$ is also an output of p_m (under the case of PPC), the firing of input transition to ${}^{(p)}\tau$ should be considered. The firing of input transition to ${}^{(p)}\tau$ still ensures that the monitor place has enough tokens to enable t and τ simultaneously.

4 Control Synthesis Method

Definition 9. A control policy U is maximally permissive if the following statements are true:

1) For any $m\in\Omega$, $R_\infty(m(p_m),U)\cap M_F(p_m)=\varnothing$;

2) For any policy U' more permissive than U, for some $m\in\Omega$, $R_\infty(m(p_m),U')\cap M_F(p_m)\neq\varnothing$.

The above notation $R_\infty(m(p_m),U)$ denotes the reachable marking set of p_m from marking m under control policy U, and $M_F(p_m)=\{m(p_m)\mid m(p_m)>b, m\in\Omega\}$. By Corollary 1, the first statement claims that constraint (1) is satisfied. The second statement states that U is more permissive than any other control policies. Similarly, $M_C(m,U)$, $R_\infty(M_C(m),U)$ and $\max[M_C(m),U]$ represent $M_C(m)$, $R_\infty(M_C(m))$ and $\max[M_C(m)]$ under U, respectively.

The following theorem implies that if a control policy U such that $m(p_m)\leqslant b\ \forall m\in\Omega$ can be found, then constraint (1) is satisfied.

Theorem 1. For any marking $m\in\Omega$, $\max[M_C(m),U]\leqslant b$ iff $R_\infty(m(p_m),U)\cap M_F(p_m)=\varnothing$.

Proof. From Corollary 1, the proof is trivial. □

Definition 10. For a transition $t\in T_{ip}$, if t will be enabled and fires k times after firing all the transitions in $\Gamma_t(s)$, then t is said to be k-enabled and k is called the enabling factor.

An influence transition t is defined as 0-enabled if it cannot be enabled though all the state enabled controllable transitions in Γ_t are fired. A controllable transition is conventionally defined as 1-enabled if it is state enabled, and 0-enabled otherwise. A k-enabled influence transition t means that t can fire k times at most. It is always possible to reduce the firing times of a k-enabled transition through control disabling some state enabled transitions.

The basic idea behind the following control algorithm is to search a candidate set $\psi(m)$ of the input transitions to p_m such that p_m will get $b'=b-m(p_m)$ tokens when the transitions in $\psi(m)$ are permitted to fire simultaneously under marking m. Usually, $\psi(m)$ is not unique.

Algorithm for control.

Step 1. Search a candidate set $\psi(m)$. The k-enabled ($k\neq 0$) transition in $\psi(m)$ is selected from ${}^{(t)}p_m$ such that $\sum k_t w(t,p_m)=b'\ \forall t\in\psi(m)$ where k_t denotes the enabling factor of t. When some controllable transitions in $\psi(m)$ or $\Gamma_t(s)$ ($t\in\psi(m)$) are in conflict, their common input place should have enough tokens to ensure that they are simultaneously fireable if control enabled. Otherwise, some controllable transitions should not be selected. If $\psi(m)$ cannot be searched in this way, reduce some enabling factors that are bigger than 1. A reduced factor is denoted as K_t, which corresponds to k_t. If the search still fails, subtract 1 from b' and re-search until $b'=0$. If $b'=0$, $\psi(m)=\varnothing$.

Step 2. Determine $U(m)$.

1) For $t\in\psi(m)$: If t is an influence transition with enabling factor k_t, let $u({}^{(c)}t_n)=1$ for each transition $t_n\in\Gamma_t(s)$, else if t is controllable, let $u({}^{(c)}t)=1$.

2) For $t\notin\psi(m)$: If t is an influence transition, the number of transition $t_n\in\Gamma_t(s)$ that should be control disabled is determined in such a way that if any already disabled t_n is enabled, t will be enabled. For other transitions in $\Gamma_t(s)$, let $u({}^{(c)}t_n)=1$. If t is controllable and state enabled, let $u({}^{(c)}t)=0$.

Remark 4. In Step 2.1 of the above control algorithm, if k_t has been reduced to K_t, not all the transitions in $\Gamma_t(s)$ are control enabled. The number of transitions that should be disabled is determined in such a way that after enabling any already disabled t_n, t will be enabled and fire more than K_t times. Note that it is a case of PPIC when some controllable transitions in $\psi(m)$ or $\Gamma_t(s)$ ($t\in\psi(m)$) are in conflict.

From the detailed steps of the control algorithm, it is easy to prove the following theorem.

Theorem 2. The obtained control policy U is maximally permissive.

Example 5. In Fig. 2, the control policy U under marking m ($m(p_i)=1$ for $i=9,12,13$ and $m(p_i)=0$ for others) is determined as follows. By the control algorithm, $\psi(m)=\{t_{15},t_{14}\}$, both of K_{t15} and k_{t14} are 1. K_{t15} is a reduced enabling factor (k_{t15} is 2). Let $u(c_i)=1$ for $i=1,3$ and $u(c_2)=0$. $u(c_4)$ has no influence on the control at current marking since t_9 is not state enabled.

5 Conclusions

This paper has addressed the control synthesis problem for a class of DES modeled by a more general CtlPN whose control specification is described as a place marking inequality. The net can model resource conflict as well as process synchronization. The uncontrollable marking from which the control constrain may be violated is tracked by a monitor. A maximally permissive control policy has been obtained based on the monitor state. Some restrictions such as PPC (PPIC) needed by previous work are relaxed in this paper. Though the designed combined PN controller has a characteristic of compiled controller in acquiring system state, it does not involve a non-convex constraint transformation that is usually unavoidable in the compiled controller when there are some uncontrollable transitions in the net. In the future, it is necessary to extend the method to a non-ordinary PN.

References

1 Giua A. Petri net techniques for supervisory control of discrete event systems. In: Proceedings of 1st Workshop on Manufacturing and Petri Nets, Osaka, 1996, 1～30

2 Holloway L E, Krogh B H. Synthesis of feedback logic for a class of controlled Petri nets. *IEEE Transitions on Automatic Control*, 1990, **35**(5): 514～523

3 Li Y, Wonham W M. Control of vector discrete-event systems Ⅱ——Controller synthesis. *IEEE Transitions on Automatic Control*, 1994, **39**(3): 512～531

4 Boel R K, Ben-Naoum L, Breusegem V V. On forbidden state problems for a class of controlled Petri nets. *IEEE Transitions on Automatic Control*, 1995, **40**(6): 1717～1731

5 Yamalidou K, Moody J O, Lemmon M D, Antsaklis P J. Feedback control of Petri nets based on place invariants. *Automatica*, 1996, **32**(1): 15～28

6 Xing Ke-Yi, Xi Yu-Geng, Hu Bao-Sheng. Petri net controller for discrete event systems with uncontrollable transitions. *Acta Automatica Sinica*, 2001, **27**(2): 180～185 (in Chinese)

7 David R, Alla H. Petri nets for modeling of dynamic systems——A survey. *Automatica*, 1994, **30**(2): 175～202

WU Wei-Min Ph. D. His research interests include DES, hybrid systems and Petri nets.

DONG Li-Da Ph. D. candidate. His research interests include DES.

WANG Xiao Ph. D. candidate. His research interests include DES.

SU Hong-Ye Ph. D. and Professor. His research interests include non linear systems, robust control and DES.

CHU Jian Ph. D. and Professor. His research interests include control theory and application, the research and development of Computer Control Systems and Advanced Process Control Software.

离散事件系统的混合型 Petri 网控制器

吴维敏　董利达　王　肖　苏宏业　褚　健

(工业控制技术国家重点实验室 浙江大学先进控制研究所　杭州　310027)

(E-mail: wmwu@iipc.zju.edu.cn)

摘　要　考虑由具有不可控变迁的受控 Petri 网建模的 DES 的控制器综合问题. 提出了兼具 DES 的逻辑型和结构型二种控制器优点的混合型 Petri 网控制器: 在系统状态的获取和跟踪上具有结构型控制器的优点, 而在控制作用的实施上则具有逻辑型控制器的优点. 全文通过实例说明了混合型 Petri 网控制器的设计方法.

关键词　离散事件系统, Petri 网, 不可控变迁, 混合型控制器

中图分类号　O158; TP271.8