15th Triennial World Congress, Barcelona, Spain

CONTROL SYNTHESIS FOR A CLASS OF CONTROLLED PETRI NETS

Weimin Wu, Hongye Su, Jian Chu and Haifeng Zhai

National Laboratory of Industrial Control Technology &
Institute of Advanced Process Control, Zhejiang University
Hangzhou, 310027, P.R.China
Email: wmwu@iipc.zju.edu.cn

Abstract: This paper proposes a new control synthesis method for a class of discrete event systems modeled by controlled ordinary Petri nets with linear marking constraint. Monitor is constructed to track the system state resulted from the uncontrollable firing sequences. The maximally permissive feedback control policy then can be obtained based on the making of the monitor. No non-convex constraint transformation is introduced in the design procedure. The method is capable of synthesizing a class of net that cannot be treated using previous methods due to some necessary restrictions. *Copyright © 2002 IFAC*

Keywords: discrete event systems, Petri nets, control system synthesis, constraint, monitor.

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 the computational efficiencies. From the control standpoint, the controller or supervisor of DES can be distinguished between *mapping* one, whose control law is a function computed after each new event generated by the system, and *compiled* one, whose control law is presented as a DES structure (Giua, 1996). A mapping controller has been designed by Holloway and Krogh (1990, 1991) to solve the forbidden state problem of safe cyclic marked graphs. Boel *et al* (1995) have obtained the mapping controller to address the same problem in the setting of state machines. For the control of vector DES, Li and Wonham (1994) presented an integer linear programming method to compute the mapping controller. Giua *et al* (1992) and Moody and Antsaklis (2000) used P-invariant method to construct a compiled controller to enforce linear marking constraint, while the controller enforcing the same constraint was obtained by Chen (2000) based on the concept of S-decrease. When design a compiled controller to enforce the given constraint, the constraint usually has to be transformed into another one in order to account for uncontrollable transitions. Moody and Antsaklis (2000) and Cho and Kwon (1998) have proposed some transformation methods.

The ordinary PN model of DES considered here is able to model both of the resource conflict and process synchronization. The restrictions against PPC (precedence path condition) and PPIC (precedence path input condition) (Holloway and Krogh, 1990, 1991; Boel *et al*, 1995) 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 the uncontrollable transitions. The controller designed in this paper is a mapping one, but it exploits the advantage of compiled controller.

The remainder of this paper is organized as follows. Section 2 introduces some foundations of the controlled ordinary Petri nets and control constraint. The construction method of monitor is given in section 3. Section 4 presents the control synthesis policy to satisfy the control constraint. In the last section, the conclusion is given.

2. FOUNDATION OF CTLPN AND CONTROL CONSTRAINT

A controlled ordinary PN is defined as a six-tuple $\mathcal{G} = (\mathcal{P}, \mathcal{T}, \mathcal{E}, \mathcal{C}, \mathcal{B}, m)$, where $\mathcal{P}$ is a finite set of state places, $\mathcal{T}$ is a finite set of transitions, $\mathcal{P} \cap \mathcal{T} = \phi$, $\mathcal{E} \subseteq (\mathcal{P} \times \mathcal{T}) \cup (\mathcal{T} \times \mathcal{P})$ is a set of directed arcs connecting state places and transitions, $\mathcal{C}$ is the finite set of control places, $\mathcal{B} \subseteq (\mathcal{C} \times \mathcal{T})$ is the set of directed arcs associating control places with transitions, and m: $\mathcal{P} \rightarrow \mathcal{Z}$ is the marking of the places ($\mathcal{Z}$ is the set of nonnegative integers). Places and transitions are called nodes in generally. The marking m and the sets $\mathcal{P}$, $\mathcal{T}$ and $\mathcal{E}$ constitute an ordinary PN $\mathcal{G}_o$, i.e., $\mathcal{G}_o = (\mathcal{P}, \mathcal{T}, \mathcal{E}, m)$. The controlled ordinary PN is sometimes referred to as controlled PN (CtlPN) and state places to as places. It is assumed in this paper that one transition has at most one connected control place and one control place has exactly connected by one transition. The transitions connected by control place are controllable and the controllable transition set is denoted by $\mathcal{T}_c$, otherwise uncontrollable and the uncontrollable transition set is represented as $\mathcal{T}_u$. The places, transitions, control places and marking are graphically represented by circles, bars, squares and dots, as shown in Fig. 1.

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

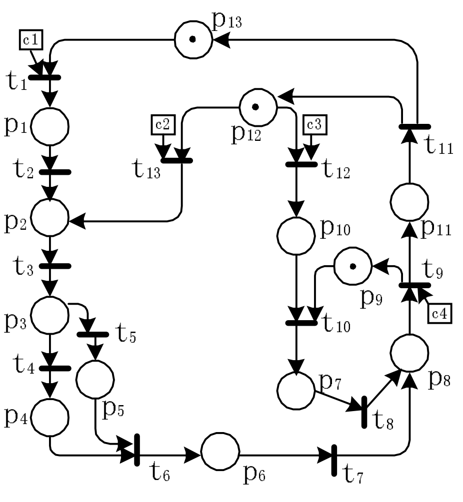

Fig. 1. A controlled PN

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

A transition $t \in \mathcal{T}$ is said to be state enabled under marking m if all its input places are marked i.e., $m(p) \geq 1$ for all $p \in {}^{(p)}t$. A transition $t \in \mathcal{T}_c$ is said to be control enabled (disabled) if its input control place is (not) assigned a token, i.e., $m({}^{(c)}t)=1$ (0). Conventionally, all the transitions in $\mathcal{T}_u$ are assumed to be control enabled. A state enabled and control enabled transition $t \in \mathcal{T}$ is said to be enabled. The firing of an enabled transition t under marking m will result in a new marking m' according to the following equation:

$$m'(p) = m(p) - |p^{(t)} \cap \{t\}| + |{}^{(t)}p \cap \{t\}|$$

where $|\cdot|$ denotes the cardinality of a set.

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

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

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 the 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 value of $M_C(m')$ under any reachable marking m' from m, and $max[M_C(m)]$ be the maximal in the set $R_\infty(M_C(m))$.

The main purpose in this paper is to seek a control policy $U \subseteq \mathcal{U}$ to make the constraint (1) be always satisfied. 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 $m' \in \Omega$.

The following definitions are about the places involved in the constraint (1) and their associated transitions.

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,

$$C_p = \{p = p_i \mid \sum_{i=1}^{n} l_i m(p_i) \leq b \ \text{ for } \ l_i \neq 0\} \qquad (2)$$

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,

$${}^{(t)}C_p = \{t \mid t \in {}^{(t)}p \text{ for } p \in C_p\} \qquad (3)$$

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,

$$C_p^{(t)} = \{t \mid t \in p^{(t)} \text{ for } p \in C_p\} \qquad (4)$$

Definition 4: The set of transitions denoted by CC_t is said to be *common constrained transition set*, if its entry t satisfies

$$t \in {}^{(t)}C_p \cap C_p^{(t)} \qquad (5)$$

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 ${}^{(t)}C_{c-pure-t} = {}^{(t)}C_{pure-t} \cap \mathcal{T}_c$ and ${}^{(t)}C_{u-pure-t} = {}^{(t)}C_{pure-t} \cap \mathcal{T}_u$, respectively.

Example 1: Consider the Petri net illustrated in Fig.1. Assume the net satisfy below constraint in its evolution:

$$2m(p_2)+m(p_4)+m(p_7) \le 3 \qquad (6)$$

then, its corresponding sets are: 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 $C^{(t)}_{pure-t} = C^{(t)}_p$. Note that $CC_t = \phi$ holds in this example.

3. CONSTRUCTION OF MONITOR

This section describes how to construct a monitor to track the state of the constrained places. At first, influence path is constructed to account for the firing of uncontrollable transitions.

3.1 Influence Path

The concept of influence path in this paper is different with the ones in (Holloway and Krogh, 1990; Boel *et al*, 1995). The influence path (IP), which is not existent in the plant, can be regarded a copy of 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 of the beginning and end nodes are transitions and $p_i \in t_i^{(p)} \cap {}^{(p)}t_{i+1}$ for $1 \le i \le n-1$. The expression '$x \in$ (or $\notin$) π' means that x is (or is not) a node in π. A sub-path of π is denoted by $\pi(x_i, x_j)$, where x is a node and $1 \le i < j \le n$.

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

A PP π_t for t has only one controllable transition t_n, and t_n is called to be the (unique) controllable transition of π_t. The case that t_n is uncontrollable is not considered here since this case will lead to the uncontrollability of the plant (Boel *et al*, 1995) or has no influence on the decision of control policy.

For a given transition $t \in {}^{(t)}C_{u\text{-}pure\text{-}t}$, it may have more than one PP. These paths are joined at some places or transitions, and these places or transitions are called to be joining nodes. The set of precedence paths for t is denoted as $\prod_t$. Let Γ_t ={t_n | $t_n \in \pi_t$ for $\pi_t \in \prod_t$, t_n is controllable} be the controllable transition set with respect to t. Let Γ_t(s) be the subset of Γ_t in which each transition is state enabled, i.e., Γ_t(s)={t_n | $t_n \in \Gamma_t$, t_n is state enabled}. $x \in$ (or $\notin$) $\prod_t$ if $x \in$ (or $\notin$) π_t for $\pi_t \in \prod_t$. 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 1 (continued): In Fig.1, there are three PP sets: $\prod_{t2}$ ={ $\pi_{t2}(t_1)$ }, $\prod_{t4}$ ={ $\pi_{t4}(t_1)$, $\pi_{t4}(t_{13})$ }, $\prod_{t10}$ ={ $\pi_{t10}(t_{12})$, $\pi_{t10}(t_9)$ }, 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 $p_2 \in \prod_{t4}$ and the controllable transitions t_{13} and t_{12} are in conflict, so this example does not satisfy PPC and PPIC.

The following definition makes it possible to evaluate the influence of firing of transitions in Γ_t on the state of constrained places without analyzing the uncontrollable reachable marking problem.

Definition 8: Given an uncontrollable input constrained transition $t \in {}^{(t)}C_{u\text{-}pure\text{-}t}$, 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 $\prod_t$, and the copied transitions are arranged in the same order as the originals. The transitions t and t_k are called to be 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 transition and its neighboring copied transitions.
3) Connect the adjacent nodes obtained above by arcs from the controllable transitions to t_{ip} in the same direction of the corresponding PP.

The transitions t_{ip-k} and t_{ip} in IP are associated to the so-called 'always occurring' events (denoted as e) (David and Alla, 1994). These transitions are fired as soon as it is enabled, and can be regarded as a kind of uncontrollable transition.

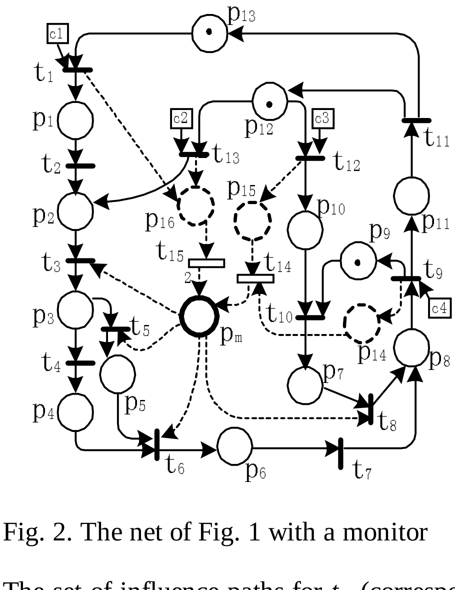

Fig. 2. The net of Fig. 1 with a monitor

The set of influence paths for t_{ip} (corresponding to t) is denoted as $\nabla_{t_{ip}}$ and the notation $\pi_{t_{ip}}(t_n)$

represents the IP with controllable transition t_n. Similarly to the case of PP set, $x\in \nabla_{tip}$ indicates that x lies in the IP set. Transition t_{ip} is called to be *influence transition* and the set of influence transition is denoted as $\mathcal{T}_{ip}$.

Example 1 (continued): 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 $\prod_{t10}$ and $\prod_{t4}$, 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 $\prod_{t2}$ is omitted since it is a subset of ∇_{t15}. Note that p_{16} and t_{14} are joining nodes. The arc, place and transition in IP set is distinguished from those in PP set by dashed line, dashed circle and rectangle as shown in Fig.2, respectively.

Influence transition and influence path play an important role in the synthesis procedure of CtlPN with uncontrollable transitions. The firing of influence transitions represents the maximally influence of the uncontrollable transitions on the control constraint. Following lemma indicates this fact.

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 $\prod_t$, 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 $p\in\prod_t$ 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 $\prod_t$ and ∇_{tip}, and reach ${}^{(p)}t$ and ${}^{(p)}t_{ip}$, respectively. For any joining transition $\tau_k\in\prod_t$ 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 is also associated to the 'always occurring' event e, which has no difference to 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 $\prod_t$, and a PP $\pi_t(t_n)\in\prod_t$, 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 $t_2\notin\prod_t$ 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 $t_5\notin\prod_t$, where $t_4\in\{\Gamma_t - t_n\}$.
2) For any conflict, if not all the transitions involved in it is controllable, there is at most one transition in some PP.

The TCC ensures that the firing of any transition $\tau\notin\prod_t$, $\tau\in p^{(t)}$ for $p\in\prod_t$ 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:
$$w=\sum_{i=1}^{n} l_i \tag{7}$$
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 also satisfies (7), but where l_i is the coefficient of $p_i\in t^{(p)}$, $p_i\in C_p$.
2) If there exists some transitions $\tau\notin\prod_t$ such that τ and $t_0\in\prod_t$ 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:
$$w=|\omega| \tag{8}$$
and
$$\omega=\sum_{i=1}^{n} l_i-\sum_{j=1}^{n} l_j \tag{9}$$
where l_i and l_j are the coefficient 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
$$m_0(p_m)=\sum_{i=1}^{n} l_i m_0(p_i) \tag{10}$$
2) The subnet constituted by IP sets. For any place p_{ip} in the subnet, its initial marking is calculated according to the following equation
$$m_0(p_{ip})=\sum m_0(p) \tag{11}$$
where $p\in\pi_t(t_i, t_j)$, t_i and t_j are any originals in the plant of $t_{ip-i}\in p^{(t)}_{ip}$ 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, the weight w of the arc from t_{ip} to the monitor place satisfies:
$$w=\max\{l_j, l_k\} \tag{12}$$
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: When the plant is a state machine, the calculation of w can be simplified. For example, w is just l_i in (7).

Remark 3: In Step 3, when t is an uncontrollable transition, it is assumed that ω>0. If ω<0, this case should be treated like 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 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 1 (continued): By the construction algorithm, the monitor shown in Fig.2 is constructed to track the state of given constraint (6). 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.

From the algorithm, below lemma that claims the relation between the states of the monitor and the constrained places can be obtained.

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

This lemma states that the number of tokens resided in the monitor place is the maximum that the constrained places can reach under the control of u_{one}. Note that the token number in the monitor constructed by Giua *et al* (1992) represents the further token number that the constrained places can get before the violation of the constraint. From lemma 2, the following corollary can be deduced directly.

Corollary 2.1 For any marking $m \in \Omega$, $max[M_C(m)] \le b$ if and only if $m(p_m) \le b$.

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, $m_0(p_m) \ge w$. 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 $m(p_m) \ge w$ be held. Thus, the monitor place will not disable the already enabled transitions in the plant.

Remark 4: If the transition $\tau \in {}^{(t)}p_i$ is also an output of p_m (it is a 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 make t and τ enable simultaneously.

4. CONTROL SYNTHESIS METHOD

This section describes how to determine the control policy based on the state of the monitor to enforce constraint (1). At first, the definition of the maximally permissiveness is presented.

Definition 9: A control policy U is maximally permissive if the following statements are true:
1) For any $m \in \Omega$, $R_\infty(m(p_m), U) \cap M_F(p_m) = \phi$.
2) For any policy U' more permissive than U, for some $m \in \Omega$, $R_\infty(m(p_m), U') \cap M_F(p_m) \ne \phi$.

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 $M_F(p_m) = \{m(p_m) \mid m(p_m) > b\}$. From corollary 2.1, the first statement claims that the constraint (1) is satisfied. The second statement states that U is more permissive than any other control policies in $\mathcal{U}$.

Similarly to $R_\infty(m(p_m), U)$, $M_C(m, U)$, $R_\infty(M_C(m), U)$ and $max[M_C(m), U]$ represent $M_C(m)$, $R_\infty(M_C(m))$ and $max[M_C(m)]$ under the control policy U, respectively.

Theorem 1: For any marking $m \in \Omega$, $max[M_C(m), U] \le b$ if and only if $R_\infty(m(p_m), U) \cap M_F(p_m) = \phi$.

Proof: From Corollary 2.1, the proof is trivial.

Theorem 1 implies that if a control policy U such that $m(p_m) \le b \;\; \forall m \in \Omega$ can be found, then the constraint (1) is satisfied.

Definition 10: For a transition $t \in \mathcal{T}_{\text{ip}}$, 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 to be 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, otherwise, 0-enabled. A k-enabled influence transition t means that t can fire k times at most. It is always possible to reduce the firing time of a k-enabled transition through control disabling some state enabled transitions.

The basic idea of control is that 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 current marking m. Usually, the candidate set $\Psi(m)$ is not unique.

Algorithm for control

Step 1: Search a candidate set $\Psi(m)$. The k-enabled ($k \neq 0$) transitions 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 Γ_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 is corresponded to k_t. If the search still fails, subtract 1 from b' and re-search until $b' = 0$. If $b' = 0$, $\Psi(m) = \phi$.

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 enable any already disabled t_n, t will be enabled. For other transitions in Γ_t (s), let $u(^{(c)}t_n)$=1. If t is controllable and state enabled, let $u(^{(c)}t)$=0.

Remark 5: When some controllable transitions in $\Psi(m)$ or Γ_t (s) ($t \in \Psi(m)$) are in conflict, it is a case of PPIC.

Remark 6: In Step 2.1 of above control algorithm, if k_t has been reduced to K_t, not all the transitions in Γ_t (s) is 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.

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 1 (continued): In Fig.2, the control policy U under current marking m ($m(p_i)$=1 for i=9, 12, 13 and $m(p_i)$=0 for others) is determined as following. 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. CONCLUSION

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. The monitor is constructed based on the concepts of precedence path and influence path. 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. So the control algorithm described above is capable of dealing with PPIC in some cases. Though the method 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.

ACKNOWLEDGEMENT

This work is supported in part by the National Natural Science Foundations of China (69974035, 60025308), Teaching and Research Award Program for Outstanding Young Teachers in Higher Education Institutions of MOE, P.R.C. and the Natural Science Key Foundation of Zhejiang Province (ZD9905)

REFERENCES

Boel, R.K, L. Ben-Naoum and V. V. Breusegem (1995). On forbidden state problems for a class of controlled Petri nets. *IEEE T-AC*, **40**, 1717-1731.

Chen, H (2000). Control Synthesis of Petri Nets Based on S-decreases. *Discrete Event Dynamic Systems: Theory and Application*, **10**, 233-249.

Cho, Y.C and W.H. Kwon (1998). On forbidden state problems in a general class of non-ordinary controlled Petri nets. in the *Proc. of 1998 IEEE Int. Conf. On SMC.* 90-95.

David, R and H. Alla (1994). Petri nets for modeling of dynamic systems—a survey. *Automatica*, **30**, 175-202.

Giua, A, F. DiCesare and M. Silva (1992). Generalized mutual exclusion constraints on nets with uncontrollable transitions. In *Proc. of the 1992 IEEE Int. Conf. on SMC.* 974-979. Chicago.

Giua, A (1996). Petri net techniques for supervisory control of discrete event systems., in *Proc. of 1st Workshops on Manufacturing and Petri Nets.* 1-30. Osaka, Japan.

Holloway, L.E and B.H. Krogh (1990). Synthesis of feedback logic for a class of controlled Petri nets. *IEEE T-AC*, **35**, 514-523.

Krogh, B.H and L.E. Holloway (1991). Synthesis of feedback logic for discrete manufacturing systems. *Automatica*, **27**, 641-651.

Li, Y and W.M. Wonham (1994). Control of vector discrete-event systems Ⅱ—Controller synthesis. *IEEE T-AC*, **39**, 512-531.

Moody, J.O and P.J. Antsaklis (2000). Petri net supervisors for DES with uncontrollable and unobservable transitions. *IEEE T-AC*, **45**, 462-476.