Continuous Petri Nets Augmented with Maximal and Minimal Firing Speeds*

Tianlong Gu, Rongsheng Dong
School of Computer Science
Guilin University of Electronic Technology
Guilin 541004, P.R. China
cctlgu@gliet.edu.cn

Yu-Chu Tian
School of Software Eng & Data Com.
Queensland University of Technology
Brisbane QLD 4001, Australia
y.tian@qut.edu.au

Abstract - *CPNs has been a useful tool not only to approximate a discrete system but also to model a continuous process. In this paper, CPNs are augmented with maximal and minimal firing speeds, and Interval speed CPNs (ICPNs) is defined. The enabling and firing of transitions of ICPNs are discussed, and the enabling of continuous transitions is classified into three levels: 0-level, 1-level and 2-level. Some rules to calculate the instantaneous firing speeds are also developed. In addition, illustrative examples are presented.*

Keywords: *Continuous Petri nets, hybrid systems, discrete event systems.*

1 Introduction

Petri net (PN), as a graphical and mathematical tool, provides a powerful and uniform environment for modeling, analysis, and control of discrete event systems [9]. In order to handle time, classic Petri nets have been extended, resulting in two basic models---timed Petri nets [10] and time Petri nets [8]. Timed Petri nets are derived by associating a finite time duration with transition or place, and the classic firing rule is modified to account for the time it takes to fire a transition. In a time Petri net, transition or place has one time interval with two values of time. The first denotes the minimal time that must elapse, starting from the time at which the transition is enabled until this transition can fire. The other one denotes the maximum time that the transition can be enabled, and after which the transition must fire. Time Petri nets are more general than timed Petri nets: a timed Petri net can be modeled by using a time Petri net, but the converse is not. Time Petri nets have been proved very convenient for expressing most of the temporal constraints while some of these constraints were difficult to express only in terms of time duration.

Motivated by approximating the discrete event systems where there exist considerable states and events, David and Alla define continuous Petri nets (CPNs) [3]. The main differences of CPNs from classic Petri nets are nonnegative real number markings of places and continuous firing of transitions at certain speed. The instantaneous firing speeds of transitions play an important role in the evolution of a CPN, which is specified uniquely by either maximal constant speed or maximal variable speed (maximal speed function with time) [4]. Due to different ways to calculate instantaneous firing speeds of transitions, various continuous Petri nets models, such as CCPN (constant speed CPN), VCPN (variable speed CPN) and ACPN (asymptotic CPN) [1][2], have been developed. However, in applications there exist many cases, where the specification of minimal and maximal speeds is required for continuous flow. Balduzzi, Giua and Menga presented the first-order hybrid Petri nets (FOHPN), where the instantaneous firing speeds are limited by minimal and maximal speeds, and calculated iteratively by linear programming [2]. Gu and Parisa discussed one typical application of sugar milling systems, and developed the hybrid time Petri net model [6][7].

On the view that the dynamics of time Petri nets can also be approximated continuously, a general CPNs formalism --- Interval speed CPNs (ICPNs) is defined in this paper. Due to constraints of maximal and minimal firing speeds, ICPNs require more subtle and complicated semantics for transitions' enabling and firing. In ICPNs, the enabling of transitions is classified into three level: 0-level, 1-level and 2-level. In addition, as a novel tool for modeling and analyzing discrete or continuous systems, ICPNs are illustrated through several cases.

2 Interval speed continuous Petri nets

2.1 Approximating time Petri nets

Time Petri net is a general model for time dependent systems. In a time Petri net, time intervals can be associated with places (called as time places) or transitions (called as time transitions). It is known that a time Petri net with time transitions can be transformed into a time

* 0-7803-7952-7/03/$17.00

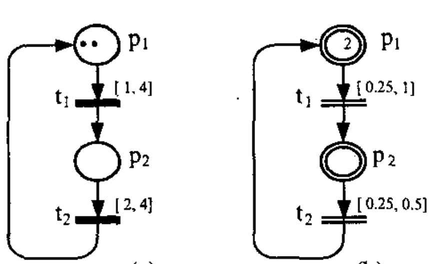
Petri net with time places and vice versa. For our purpose, a time Petri net with time transitions is adopted. Normally, a time interval is specified by two values of time – maximum and minimal times.

Consider the time Petri net in Figure 1(a). Time intervals [1,4] and [2,4] are associated with transitions t_1 and t_2 respectively. The marking in Figure 1(a) corresponds to time $\tau = 0$. Obviously, transition t_1 is enabled at time $\tau = 0$, then a token is reserved in p_1 in order to fire the transition. Transition t_1 can be fired after time $\tau = 1$, and must be fired before time $\tau = 4$. Supposed that each transition works in the earliest firing mode, i.e., each transition is fired as soon as it can be fired. Then, at time $\tau = 1$ transition t_1 is fired, and the reserved token in p_1 is taken away, and non-reserved token is put into p_2. At time $\tau = 1$ transitions t_1 and t_2 are enabled. At time $\tau = 2$ transitions t_1 is fired again, and one reserved token in p_1 is taken away, and one non-reserved token is put into p_2. At time $\tau = 3$ transitions t_2 is fired, and one reserved token in p_2 is taken away, and one non-reserved token is put into p_1. At time $\tau = 3$, transitions t_1 and t_2 are enabled again, and they will be fired at times $\tau = 4$ and $\tau = 5$ respectively, and so on. The corresponding markings m_1 and m_2 are illustrated by dash lines in Figure 1(c). After time $\tau = 1$, a periodical behaviour with the period $\mu = 2$ is reached. Similarly, the markings m_1 and m_2 for different firing modes of transitions t_1 and t_2 can be derived respectively.

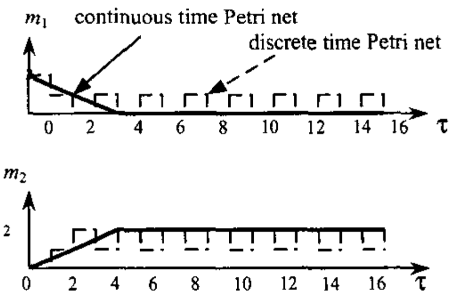
From the time Petri net of Figure 1(a), it is possible to construct a continuous model by replacing time values d_j and d_j' with maximum firing speeds $V_j = 1/\ d_j$ and minimal firing speeds $V_j' = 1/\ d_j'$. This gives the continuous time Petri net of Figure 1(b). In the model, transitions t_1 and t_2 will be fired at the instantaneous firing speeds $v_1(\tau)$ and $v_2(\tau)$ after time τ respectively. At the initial time $\tau = 0$, transition t_1 is strongly enabled, and transition t_2 weakly enabled [2][4]. The evolution of the markings in two places is governed by the following equations in the interval $[0, \tau_1)$ (in this interval, $v_1(\tau) = 1$ and $v_2(\tau) = 0.5$):

$$m_1(\tau) = 2 + (1 - 0.5)\ \tau,$$

$$m_2(\tau) = (1 - 0.5)\tau.$$



(a) (b)



(c)

Figure 1. Approximating time Perti net

These equations remain true until place p_1 becomes empty at time $\tau_1 = 2/0.5 = 4$. At all times $\tau \geq \tau_1$ (= 4), transitions t_1 and t_2 are fired at the speeds $v_1(\tau) = v_2(\tau) = 0.5$ since transition t_1 is weakly enabled and t_2 strongly enabled. Thus, a steady state is reached from time $\tau = \tau_1$, and the evolution of markings is illustrated in Figure 1(c). According to the firing and enabling semantics of time Perti nets, the continuous time Petri net may have different behaviours shown in Figure 1(d)(e)(f), and so on.

If we change the times specification of time Petri net in Figure 1(a), an unexpected situation occurs. Consider the time Petri net in Figure 2(a), and the corresponding continuous time Petri net is shown in Figure 2(b). The evolution of markings m_1 and m_2 of the time Petri net is illustrated by dash lines in Figure 2(c), supposed that the earliest firing mode for transitions t_1 and t_2 is adopted. After time $\tau = 1$, a periodical behaviour with the period $\mu = 3$ is reached.

In the continuous time Petri net, at the initial time $\tau = 0$ transition t_1 is strongly enabled, and transition t_2 weakly enabled [6][8]. But we couldn't give the instantaneous firing speed (IFS) $v_2(\tau)$ of transition t_2 because $v_2(\tau)$ must be in the interval [0.5, 1], and its input place p_2 can receive the supply flow only at the speed $v_1(\tau) = 1/3$. However, after any time delay δ, place p_2 will not be empty, and transition t_2 will be strongly enabled. Thus, there will exist feasible value for the instantaneous firing speed of $v_2(\tau)$ of transition t_2. For this situation, we could assume that weakly enabled transition t_2 has a time delay (ex. $\delta = d_j = 1/\ V_j$). Then, at time $\tau_1 = \delta$, transition t_2 will be strongly enabled, and we can give he instantaneous firing speed $v_2(\delta) = 1$, and so on. By this way, the evolution of markings in this continuous time Petri net can be derived, and shown in Figure 2(c).

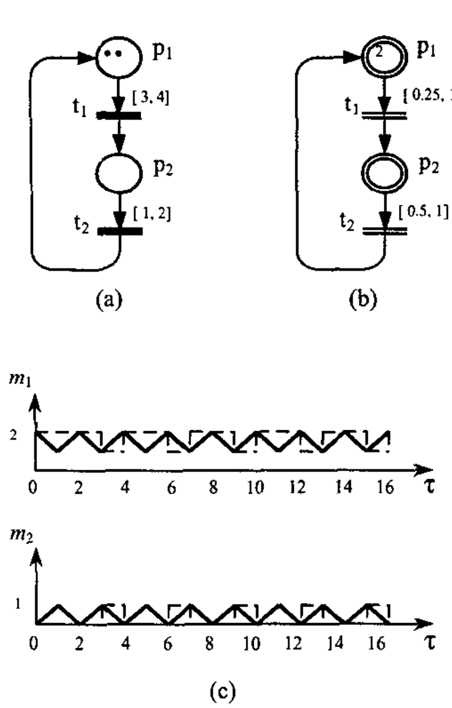


Figure 2. An unexpected situation

2.2 Formalism of interval speed continuous PN

This section formalizes some concepts, which were presented intuitively in the previous section, and Interval speed continuous Petri nets (ICPNs) is defined to approximate time Petri nets. The common formalism and notation of PNs and CPNs are used, and comprehensive introduction can be found in [1][3][6].

Definition 1: An interval speed continuous Petri net (ICPN) is a 5-tuple: N = (P, T, *Pre*, *Post*, *F*), where

$P = \{p_1, p_2, p_3, \ldots, p_n\}$ is a set of continuous places;

$T = \{t_1, t_2, t_3, \ldots, t_m\}$ is a set of continuous transitions;

$P \cap T = \varnothing$, i.e. the sets P and T are disjointed;

Pre: $T \times P \rightarrow R^+$ (or $P \times T \rightarrow R^+$) ($R^+$ is a set of non-negative real number), is the transition (or place) input incidence mapping;

Post: $T \times P \rightarrow R^+$ (or $P \times T \rightarrow R^+$), is the transition (or place) output incidence mapping;

F: $T \rightarrow R^+ \times (R^+ \cup \{\infty\})$, is the flow or speed interval mapping.

In ICPNs, the mapping *F* specifies the firing speeds associated to continuous transitions. For any continuous transition $t_j \in T$, let $F(t_i) = [V_j, V_j']$, with $V_j \leq V_j'$, where V_j represents the minimal speed, and V_j' represents the maximum speed. Here, speed specification $F(t_i) = [V_j, V_j']$ of each continuous transition approximates the time specification $D(t_i) = [d_j, d_j']$ of its discrete version, with $V_j = 1/d_j$ and $V_j' = 1/d_j'$.

Definition 2: A marked ICPN is a 6-tuple: (N, $\boldsymbol{m}(\tau_0)$) = (P, T, *Pre*, *Post*, *F*, $\boldsymbol{m}(\tau_0)$), where $\boldsymbol{m}(\tau_0)$ represents the initial marking vector, and $\boldsymbol{m}$: $P \rightarrow R^+$, is a marking function that assigns to each continuous place a nonnegative real number tokens. For any place $p_i \in P$, its tokens at time τ is denoted by $m_i(\tau)$ or m_i.

2.3 Enabling and firing in ICPNs

The enabling of continuous transitions in ICPNs depends on not only the current marking but also the feeding flow of all its input places. We use the similar notation ${}^\bullet x$ ($x^\bullet$) to denote the input (output) set of element $x \in P \cup T$.

Definition 3: A place $p_i \in P$ is supplied or fed if and only if there is at least one of its input transitions $t_j \in {}^\bullet p_i$ is being fired at a positive speed $v_j(\tau)$ (>0).

Definition 4: A transition $t_j \in T$ is enabled at time τ if all input places $p_i \in {}^\bullet t_j$ satisfy that either $m_i(\tau) > 0$, or p_i is supplied, otherwise, the transition is disabled.

Definition 5: A enabled transition $t_j \in T$ is called as strongly enabled or 2-level enabled at time τ if all input places $p_i \in {}^\bullet t_j$ satisfy $m_i(\tau) > 0$.

Definition 6: A enabled transition $t_j \in T$ is called as weakly enabled at time τ if at least one of its input places $p_i \in {}^\bullet t_j$ doesn't satisfy $m_i(\tau) > 0$.

Definition 7: A weakly enabled transition $t_j \in T$ is called as 1-level enabled at time τ if all the supplied place $p_i \in {}^\bullet t_j$ satisfy the following condition: $\sum_{k \neq j} Post(p_i, t_k)\, v_k(\tau) - \sum_k Pre(p_i, t_k)\, v_k(\tau) \geq V_j$.

Definition 8: A weakly enabled transition $t_j \in T$ is called as 0-level enabled at time τ if one of the supplied place $p_i \in {}^\bullet t_j$ satisfy the following condition: $\sum_{k \neq j} Post(p_i, t_k)\, v_k(\tau) - \sum_k Pre(p_i, t_k)\, v_k(\tau) < V_j$.

Clearly, either 2-level or 1-level enabled transition can be fired once it is enabled. However, 0-level enabled transitions can be fired only after certain time delay, since there doesn't exist feasible instantaneous firing speed at the moment.

Property 1: 2-level enabled transition $t_j \in T$ can be fired at the instantaneous firing speed $v_j(\tau) \in [V_j, V_j']$.

Property 2: 1-level enabled transition $t_j \in T$ can be fired at the instantaneous firing speed $v_j(\tau)$ satisfying the following conditions:

$$V_j \leq v_j(\tau) \leq V_j' \quad (1)$$

$$v_j(\tau) \leq \sum_{k \neq j} Post(p_i,t_k)\, v_k(\tau) - \sum_k Pre(p_i,t_k)\, v_k(\tau),$$
$$\text{for all places } p_i \in {}^{\bullet}t_j \quad (2)$$

Property 3: 0-level enabled transition $t_j \in T$ can be fired after the time delay $d_j = 1/V_j$ at the instantaneous firing speed $v_j(\tau) \in [V_j, V_j']$, unless the transition is disabled before time $(\tau + d_j)$.

In ICPNs, a conflict situation can arise if a unique place has to supply two or more transitions. When the unique place holds positive tokens or has proper feed flows, the conflict is not effective.

Definition 9: A conflict occurs when a place $p_i \in P$ has at least two output transitions. We denote a conflict by $K = < p_i, \{t \mid t \in p_i^{\bullet}\}>$. A conflict is effective if the following conditions are met: $m_i(\tau) = 0$; and $\sum_k Post(p_i,t_k) V_j \leq \sum_k Pre(p_i,t_k)\, v_k(\tau) \leq \sum_k Post(p_i,t_k)\, V_j'$.

Some typical conflict situations are shown in Figure 3. Figure 3(a), (b) and (c) are conflicts, but not effective conflicts. Figure 3(d), (e) and (f) are conflicts, and effective conflicts.

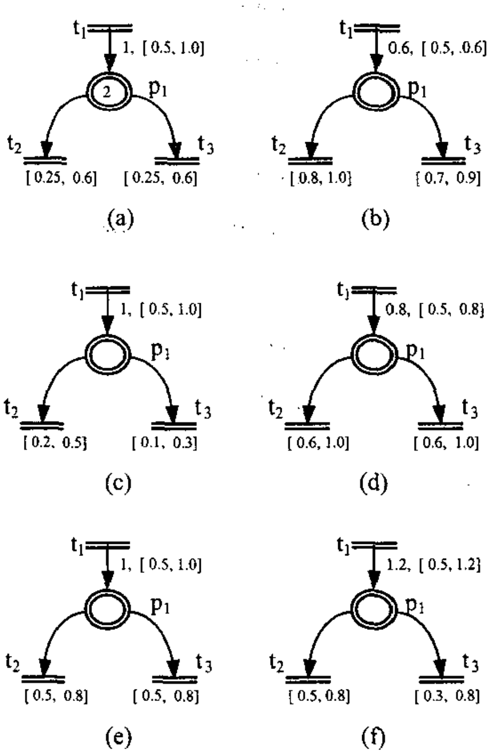


Figure 3. conflict situations

Property 4: Effective conflict $K = < p_i, \{t \mid t \in p_i^{\bullet}\}>$ can be resolved by either one of the following policies:

(priority policy)

$$v_j(\tau) = \begin{cases} \min\left(V_j',\ \left(\sum_k Pre(p_i,t_k)\, v_k(\tau) - \sum_{r>j} V_r'\right)\right), \\ \quad \text{if } \left(\sum_k Pre(p_i,t_k)\, v_k(\tau) - \sum_{r>j} V_r'\right) \geq V_j; \\ 0, \quad \text{otherwise.} \end{cases}$$

(proportionally sharing policy)

$$v_j(\tau) = \min\left(V_j',\ V_j' \sum_k Pre(p_i,t_k)\, v_k(\tau) / \sum_r V_r'\right).$$

(note: $r>j$ represents for all the transitions that are in $p_i^{\bullet}$ and have priority over t_j, and $\sum_r$ for all transitions $t_r \in p_i^{\bullet}$).

Proposition 1: If there exist feasible IFSs for an effective conflict $K = < p_i, \{t \mid t \in p_i^{\bullet}\}>$ by proportionally sharing policy, then $\sum_k Pre(p_i,t_k)\, v_k(\tau) \geq \sum_r V_r$.

Proof: Any feasible IFSs must satisfy that for $\forall t_r \in p_i^{\bullet}$, $V_r' \geq v_r(\tau) \geq V_r$. Thus, $\sum_r v_r(\tau) \geq \sum_r V_r$. From the proportionally sharing policy, $\sum_r v_r(\tau) = \min\left(\sum_r V_r', (\sum_r V_r')(\sum_k Pre(p_i,t_k) v_k(\tau) / \sum_r V_r')\right) = \min\left(\sum_r V_r', \sum_k Pre(p_i,t_k)\, v_k(\tau)\right)$. Therefore, $\sum_r V_r \leq \min\left(\sum_r V_r', \sum_k Pre(p_i,t_k) v_k(\tau)\right) \leq \sum_k Pre(p_i,t_k)\, v_k(\tau)$.

The conflict situations in Figure 3 are revisited. Transitions t_2 and t_3 in Figure 3(a) and (c) can be fired in maximal speed mode, i.e., their IFSs can be set as $v_2(\tau) = v_3(\tau) = 0.6$ and $v_2(\tau) = 0.5$, $v_3(\tau) = 0.3$ respectively. In Figure 3(b), either one of transitions t_2 and t_3 can not be fired at the moment, i.e., $v_2(\tau) = v_3(\tau) = 0.0$, since there do not have feasible IFSs for them.

The conflict in Figure 3(d) can be resolved only by priority policy, and the IFSs be set as $v_2(\tau) = 0.8$, $v_3(\tau) = 0.0$ if transition t_2 has the priority over transition t_3.

In Figure 3(e) and (f), each resolution policy can be adopted, resulting in different IFSs. In terms of the priority policy, we have $v_2(\tau) = 0.8$ and $v_3(\tau) = 0.0$, $v_2(\tau) = 0.8$ and $v_3(\tau) = 0.4$ respectively. Using the proportionally sharing policy, the IFSs are $v_2(\tau) = v_3(\tau) = 0.5$, $v_2(\tau) = v_3(\tau) = 0.6$ respectively.

3 Behavioural analysis of ICPNs

3.1 Enabled transitions and IFSs

The behavioral evolution of ICPNs is both enabled transitions and IFSs dependent. Due to the recursive definitions, it is not trivial to know whether or not a transition is enabled and to calculate the IFSs. The algorithm to determine enabled transitions and IFSs is presented as follows. It is assumed that 2-level enabled transitions function in maximal speed mode, and that the effective conflicts are resolved by the proportionally sharing policy.

Algorithm 1: (*Calculation of enabled transitions and IFSs at time* τ)

1° Initialization $ET_0 = ET_1 = ET_2 = \varnothing$; $time(t_j) = 0$ for $\forall t_j \in T$;

2° For $p_i \in P$ with $m_i(\tau) > 0$, find all 2-level enabled transitions $t_j \in p_i^{\bullet}$; let $ET_2 := ET_2 \cup \{t_j\}$, $v_j(\tau) = V_j'$, $time(t_j) := \tau$, and $T := T - \{t_j\}$;

3° For $p_i \in P$ with $m_i(\tau) = 0$, find all disabled transitions $t_j \in T \cap p_i^{\bullet}$; let $v_j(\tau) = 0$, $time(t_j) := \tau$, and $T := T - \{t_j\}$;

4° Find all 0-level enabled transitions t_j in T; let $v_j(\tau) = 0$, $time(t_j) := \tau + 1/V_j$, $ET_0 := ET_0 \cup \{t_j\}$ and $T := T - \{t_j\}$;

5° By proportionally sharing policy and equation (1) (2), calculate the IFSs of transition $t_j \in T$; let $time(t_j) := \tau$, $ET_1 := ET_1 \cup \{t_j\}$ and $T := T - \{t_j\}$.

3.2 Behavioural analysis

Similar to CCPN, the marking of a place in ICPNs is a time continuous function. A characteristic quantity of dynamic evolution of ICPNs is the IFS vector, which remains constant in a regional state.

Definition 10: A regional state is defined as (M, V, $[\tau_1, \tau_2]$), where M is the marking vector of all continuous places, and V is the IFS vector of all continuous transitions which keeps unchanged during time interval $[\tau_1, \tau_2]$.

The behavioral evolution of ICPNs is driven by discrete events of empting continuous place. However, a regional state occurs when: 1) a continuous place becomes empty, or 2) a 0-level enabled transition is fired after its delay. Thus, the duration of a time interval $[\tau_1, \tau_2\}$ in a regional state is determined by the first place whose marking becomes zero, or the first 0-level enabled transitions which will be fired, i.e., $\Delta_k = \tau_k - \tau_{k-1}$ is given by

$$\Delta_k = \mathrm{Min}\ \{\ \mathrm{Min}_i\ \{m_i(\tau_{k-1}) / \textstyle\sum_r Post(p_i,t_r)\ v_r(\tau_{k-1})\},$$
$$\mathrm{Min}_j\ \{1/V_j \mid t_j \in ET_0\}\}.$$

Algorithm 2: (*Behavioral analysis of an ICPN*)

1° Initialization $k = 1$, $\tau_k = 0$, $M(\tau_k)$;

2° If $V(\tau_{k+1}) = V(\tau_k)$, then stop; else, using *algorithm 1*, calculate ET_0, ET_1, ET_2, IFSs vector V, and time vector *time*;

3° Calculate $\Delta_k = \mathrm{Min}_i\ \{m_i(\tau_{k-1}) / \sum_r Post(p_i,t_r)\ v_r(\tau_{k-1})\}$;

4° If $ET_0 = \varnothing$, then update $\tau_k := \tau_k + \Delta_k$, $m_i(\tau_k) := m_i(\tau_k) + \sum_r Pre(p_i,t_r)\ v_r(\tau_k) - \sum_r Post(p_i,t_r)\ v_r(\tau_k)$, and goto 2°;

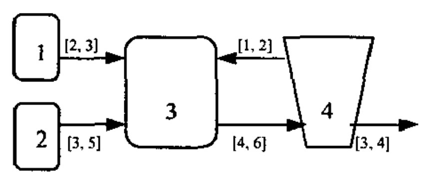
5° Find the transition $t_j \in ET_0$ satisfying $1/V_j = \mathrm{Min}_r \{1/V_r \mid t_r \in ET_0\}$;

6° If $1/V_j \geq \Delta_k$, then update $\tau_k := \tau_k + \Delta_k$, $m_i(\tau_k) := m_i(\tau_k) + \sum_r Pre(p_i,t_r)\ v_r(\tau_k) - \sum_r Post(p_i,t_r)\ v_r(\tau_k)$, and goto 2°;

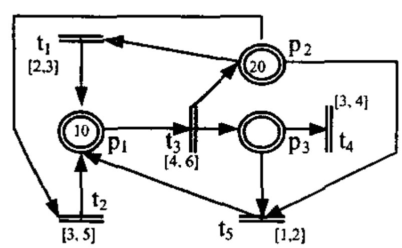
7° Update $\tau_k := \tau_k + 1/V_j$, $m_i(\tau_k) := m_i(\tau_k) + \sum_r Pre(p_i,t_r)\ v_r(\tau_k) - \sum_r Post(p_i,t_r)\ v_r(\tau_k)$, $V(\tau_k) := V(\tau_k)$, $v_j(\tau_k) := V_j$, and goto 2°.

4 Application

A chemical process with 4 units and 4 operations is shown in Figure 4(a). Two kinds of materials are processed in unit 1 (operation 1) and unit 2 (operation 2) respectively, and then fed to unit 3, where operation 3 is undertaken. The feed flow from unit 1 to unit 3 is limited within [2, 3], and The feed flow from unit 2 to unit 3 within [3, 5]. Intermediate product is fed from unit 3 to unit 4 at a flow of [4,6]. There are two output flows of unit 4, one is the final product flow at speed of [3, 4], and the other is the recycled flow to unit 3 at speed of [1,2]. The capacity of unit 3 is limited by 30, and its initial volume is 10.



(a)



(b)

Figure 4. one chemical process

This process can be modeled as an ICPN shown in Figure 4(b). From the ICPN model, we can analyze the dynamic behavior as follows.

At the initial time $\tau = 0$, $m_1(0) = 10$, $m_2(0) = 20$ and $m_3(0) = 0$. Then, $ET_2 = \{t_1, t_2, t_3\}$, $ET_1 = \{t_4, t_5\}$, and $v_1(0) = 3$, $v_2(0) = 5$, $v_3(0) = 6$, $v_4(0) = 4$, $v_5(0) = 2$. From time τ = 0, ICPN's behaviour is governed by the following equations: $m_1(\tau) = 10 + 4\tau$, $m_2(\tau) = 20 - 4\tau$, $m_3(\tau) = 0$.

At time $\tau = 5$, $m_1(5) = 30$, $m_2(5) = 0$ and $m_3(5) = 0$. Transition t_3 is still strongly enabled, and $ET_2 = \{t_3\}$, $ET_1 = \{t_2, t_4, t_5\}$, $ET_0 = \{t_1\}$, $v_1(5) = 0$, $v_2(5) = 3$, $v_3(5) = 6$, $v_4(5) = 4$, $v_5(5) = 1.2$. From time $\tau = 5$, ICPN's behavior is governed by the following equations: $m_1(\tau) = 39 - 1.8\tau$, $m_2(\tau) = 1.8\tau - 9$, $m_3(\tau) = 0.8\tau - 4$.

At time $\tau = 5.5$, 0-level enabled transition t_1 is fired at $v_1(5.5) = 2$. The IFSs of other transitions remain unchanged, and $m_1(5.5) = 29.1$, $m_2(5.5) = 0.9$, $m_3(5.5) = 0.4$. Thus, from time $\tau = 5.5$, the equations governing the ICPN's behavior are changed to: $m_1(\tau) = 28 + 0.2\tau$, $m_2(\tau) = 2 - 0.2\tau$, $m_3(\tau) = 0.8\tau - 4$.

At time $\tau = 10$, $m_1(10) = 30$, $m_2(10) = 0$ and $m_3(10) = 4$. Then, $ET_2 = \{t_3, t_4\}$, $ET_1 = \{t_2, t_5\}$, $ET_0 = \{t_1\}$, and $v_1(10) = 0$, $v_2(10) = 3$, $v_3(10) = 6$, $v_4(10) = 4$, $v_5(10) = 1.2$. Thus, from time $\tau = 10$, the marking equations of the ICPN are: $m_1(\tau) = 48 - 1.8\tau$, $m_2(\tau) = 1.8\tau - 18$, $m_3(\tau) = 0.8\tau - 4$.

At time $\tau = 10.5$, 0-level enabled transition t_1 is fired at $v_1(10.5) = 2$. The IFSs of other transitions remain unchanged, and $m_1(10.5) = 29.1$, $m_2(10.5) = 0.9$, $m_3(10.5) = 4.4$. Thus, from time $\tau = 5.5$, the equations governing the ICPN's behavior are changed to: $m_1(\tau) = 28 + 0.2\tau$, $m_2(\tau) = 2 - 0.2\tau$, $m_3(\tau) = 0.8\tau - 4$.

It is clear that after time $\tau = 5.5$ the ICPN reaches following periodical behavior:

$$\begin{cases} v_1(\tau) = 0,\ v_2(\tau) = 3,\ v_3(\tau) = 6,\ v_4(\tau) = 4,\ v_5(\tau) = 1.2 \\ \quad \text{when } \tau \in [5k, 5k+0.5],\ k = 1,2,\ldots \\ v_1(\tau) = 2,\ v_2(\tau) = 3,\ v_3(\tau) = 6,\ v_4(\tau) = 4,\ v_5(\tau) = 1.2 \\ \quad \text{when } \tau \in [5k+0.5, 5(k+1)],\ k = 1,2,\ldots \end{cases}$$

5 Conclusions

The continuous flows with maximal and minimal limits are important characteristic quantity in either approximating discrete event systems or describing continuous processes. On the view of approximating time Peti net, Interval Speed Continuous Petri Net (ICPN) is developed in this paper. ICPN can be considered as a general formalism of continuous processes. When the minimal speed limit is relaxed, an ICPN reduces to a CCPN.Associating maximal and minimal firing speeds with continuous transitions renders that dynamics and properties in ICPNs are much more complicated than in traditional CCPN. Further efforts are required to establish more theoretical foundation regarding net dynamics and structure properties for ICPNs. In addition, the optimization and control of continuous and hybrid processes via ICPN model are under way.

***Acknowledgments*:** This work has been supported by National Natural Science Foundation of China (Grant No. 60243002).

References

[1] J. Bail Le, H. Alla, and R. David, "Asymptotic Continuous Petri Nets: An Efficient Approximation of Discrete Event Systems", Pro. 1992 IEEE International Conference on Robotics and Automation, Nice, France, pp. 1050-1056, 1992

[2] F. Balduzzi, A. Giua, and G. Menga, "First-order hybrid Petri nets: a model for optimization and control", *IEEE Trans. on Robotics and Automation*, 16(4), pp. 382-398, 2000

[3] R. David, and H. Alla, Continuous Petri Nets. Proceedings of the 8th European Workshop on Application and Theory of Petri Nets, Zaragoza, pp. 275-294, 1987

[4] R. David, and H. Alla, "On Hybrid Petri Nets", *Discrete Event Dynamic Systems: Theory and Applications*, 11(1), pp. 9-40, 2001

[5] E. Dubois, H. Alla, and R. David, "Continuous Petri Net with Maximal Speeds Depending on Time", Pro. the 4th International Conference of Computer Integrated Manufacturing and Automation Technology, Troy, pp. 32-39, 1994

[6] T. Gu, and P. A. Bahri, "A Survey of Petri-Net Applications in Batch Processes", *Computers in Industry*, 47(1), pp. 99-111,2002

[7] T. Gu, P. A. Bahri, and P. L. Lee, "Development of Hybrid Time Petri Nets for Scheduling of Mixed Batch/Continuous Process", Pro. the 15th IFAC World Congress on Automatic Control, Barcelona, Spain, July, 2002

[8] P. M. Merlin, and D. J. Farber, "Recoverability of Communication Protocols --- Implications of a Theoretical Study", *IEEE Trans. on Communications*, 24(9), pp. 1036-1043, 1976

[9] T. Murata, "Petri nets: properties, analysis and applications", *Proceedings of the IEEE*, 77(1), pp. 541-580, 1989

[10] C. Ramchandani, "Analysis of asynchronous concurrent systems by timed Petri nets", Massachusetts Inst. Technology, Technique Rep. 120, 1974