

含有效冲突的恒定连续 Petri 网演变图及构造算法

赵义军 张小轩

(山东科技大学数学与系统科学学院 青岛 266590)

摘 要 最大速度恒定的连续 Petri 网(CCPN)是由 David 首先提出的一类时延连续 Petri 网模型,构造其演变图是对其性质进行分析的一种有效方法。而对于含有效冲突的最大速度恒定的连续 Petri 网,由于有效冲突所引起的变迁激发的不确定性,使得构造演变图变得较为困难。基于有效冲突的两种解决方式——确定优先级或按比例分配流量,给出了计算各变迁瞬发速度的算法 2;进而对含有效冲突且有界的 CCPN,提出了演变图构造算法 3,利用其演变图,可以对含有效冲突的 CCPN 进行性能分析。

关键词 最大速度恒定的连续 Petri 网,强(弱)使能,有效冲突,瞬发速度,运行区间,演变图,算法

中图分类号 TP301

文献标识码 A

DOI 10.11896/j.issn.1002-137X.2016.11.018

Developing Graph and Generating Algorithm for Constant Continuous Petri Net with Effective Conflict

ZHAO Yi-jun ZHANG Xiao-xuan

(College of Mathematics and Systems Science, Shandong University of Science and Technology, Qingdao 266590, China)

Abstract Maximum constant continuous Petri net(CCPN), one of the timed continuous Petri net models, was first put forward by David. An efficient way for analysis of CCPN properties is developing graph of CCPN. However, for CCPN with effective conflict, it makes the construction of developing graph become more difficult because of the uncertainty of transition excitation caused by effective conflict. Based on the two kinds of solutions for effective conflict-determining priority or distributing flow in proportion, algorithm 2 for calculating the velocity of every transition was presented, and generating algorithm 3 of developing graph was further put forward for bounded CCPN with effective conflict. By means of which, analysis can be made for analyzing the properties of CCPN with effective conflict.

Keywords Maximum constant continuous Petri net, Strongly(Weakly) enabled, Effective conflict, Instantaneous firing velocity, Functioning interval, Developing graph, Algorithm

最大速度恒定的连续 Petri 网(CCPN)是由 David 提出的一类连续 Petri 网模型。应用 CCPN 的概念和理论,可以对许多与时间和连续量有关的系统进行建模,也可对实时离散事件动态系统近似建模,因而在理论及应用上有较广泛的发展前景。David 等^[1]给出了 CCPN 的定义及使能变迁(包括强、弱两种使能)、可供给位置的定义,并提出了计算使能变迁集和使能变迁瞬发速度的算法 1 和算法 2。为了分析 CCPN 的性质,同时给出了 CCPN 的基本关系、运行区间和演变图构造的具体例子。赵义军等^[2,3]进一步提出了强、弱供给位置的定义及运行区间的严格定义和划分,在此基础上提出了不含有效冲突 CCPN 的演变图构造算法。而对于含有效冲突的 CCPN,在有效冲突中各变迁的瞬发速度的计算取决于有效冲突的解决策略。本文基于有效冲突的两种解决方式——确定优先级或按比例分配流量,给出了计算各变迁瞬发速度的算法 2,进而对含有效冲突且有界的 CCPN,提出了演变图的构造算法 3。利用其演变图,可以对含有效冲突的有界 CCPN 进行性能分析。

1 CCPN 的基本概念、术语及有关结论

定义 1^[1] 一个最大速度恒定的连续 Petri 网(CCPN)是

一个五元组: $\Sigma_c(t) = (P, T; F, M(t), V)$, 其中: 1) $N = (P, T; F)$ 为一个网; 2) $M(t): P \rightarrow R_0$ (非负实数集) 为 t 时刻的标识; 3) $V: T \rightarrow R^+$ (正实数集), $V(T_j) = V_j$ 为对应于变迁 T_j 的最大激发速度。

$m_i(t) = M(t)(p_i)$ 表示 t 时刻位置 p_i 的标记, 从而 $M(t) = [m_1(t), m_2(t), \dots, m_m(t)]^T$, 其中 $m = |P|$; 用 $v_j(t)$ 表示 t 时刻变迁 T_j 的瞬时激发速度, 从而 $v(t) = [v_1(t), v_2(t), \dots, v_n(t)]^T$ 表示 t 时刻变迁集 T 的瞬时激发速度向量, 其中 $n = |T|$ 。

定义 2 在 t 时刻, 位置集 P 的各位置可划分为 3 类: 强供给位置、弱供给位置、不可供给位置。具体如下:

(1) 若位置 p_i 的标记 $m_i(t) > 0$, 则称 p_i 是强供给的, 所有强供给的位置集记作 $P^s(t)$;

(2) 若位置 p_i 的标记 $m_i(t) = 0$, 但 p_i 中至少有一变迁使能, 则称 p_i 是弱供给的, 所有弱供给的位置集记作 $P^w(t)$;

(3) 位置 p_i 是可供给的, 当且仅当它是强供给的或弱供给的, 所有可供给的位置集记作 $P^{sw}(t)$;

(4) 若位置 p_i 既不是强供给的, 也不是弱供给的, 则称 p_i 是不可供给的, 所有不可供给的位置集记作 $P^m(t)$ 。

定义 3 在 t 时刻, 变迁集 T 的各变迁可划分为 3 类: 强

到稿日期: 2015-07-16 返修日期: 2016-02-07 本文受国家自然科学基金(61304074)资助。

赵义军(1967—), 男, 硕士, 教授, 主要研究方向为 Petri 网理论及应用, E-mail: zyj6702@163.com; 张小轩(1989—), 男, 硕士生, 主要研究方向为计算理论与数据处理。

使能变迁、弱使能变迁、不可使能变迁。具体如下：

(1) 若 $\forall p_i \in \cdot T_j$, 均有 $p_i \in P^s(t)$, 则称变迁 T_j 是强使能的 (若 $\cdot T_j = \emptyset$, 则 T_j 在任何时刻都是强使能的), 所有强使能的变迁集记作 $T^s(t)$;

(2) 若 $\forall p_i \in \cdot T_j$, 均有 $p_i \in P^{sw}(t)$ 且 $\cdot T_j \cap P^w(t) \neq \emptyset$, 则称变迁 T_j 是弱使能的, 所有弱使能的变迁集记作 $T^w(t)$;

(3) 变迁 T_j 是使能的, 当且仅当它是强使能的或弱使能的, 所有使能的变迁集记作 $T^{sw}(t)$;

(4) 若变迁 T_j 既不是强使能的, 也不是弱使能的, 则称 T_j 是不可使能的, 所有不可使能的变迁集记作 $T^e(t)$ 。

由定义 3 可知, 在 t 时刻使能的变迁包括强、弱两种使能, 而由定义 2 和定义 3 的递归性可知, 无法直接得到在时刻 t 一个变迁是否是弱使能, 这需要考察变迁的前置位置是否是可供给, 同样考察一个位置是否是弱供给也存在着类似的情况, 这对大型的 CCPN 来说较复杂。为此给出了确定使能变迁集 $T^{sw}(t)$ 和可供给位置集 $P^{sw}(t)$ 的算法, 如算法 1 所示。

首先定义布尔变量如下：

$$a_i = \begin{cases} 1, & p_i \in P^{sw}(t) \\ 0, & p_i \in P^e(t) \end{cases}, e_j = \begin{cases} 1, & T_j \in T^{sw}(t) \\ 0, & T_j \in T^e(t) \end{cases}$$

于是获得固定点的方程组：

$$a_i = \begin{cases} 1, & p_i \in P^s(t) \\ \max\{e_j \mid T_j \in \cdot p_i\}, & \text{否则} \end{cases}$$

$$e_j = \begin{cases} 1, & T_j \in T^s(t) \\ \min\{a_i \mid p_i \in \cdot T_j\}, & \text{否则} \end{cases}$$

方程组形式是 $X=G(X)$, 其中 $X=(a_1, a_2, \dots, a_m, e_1, e_2, \dots, e_n)$ 是 $m+n$ 维的布尔向量, 令 $X=(A, E)$, $A=(a_1, a_2, \dots, a_m)$, $E=(e_1, e_2, \dots, e_n)$, 迭代方式 G 定义如下：

$$A^{k+1}=G(A^k)$$

$$a_i^{k+1} = \begin{cases} 1, & p_i \in P^s(t) \\ \max\{e_j^k \mid T_j \in \cdot p_i\}, & \text{否则} \end{cases}$$

$$E^{k+1}=G(E^k)$$

$$e_j^{k+1} = \begin{cases} 1, & T_j \in T^s(t) \\ \min\{a_i^{k+1} \mid p_i \in \cdot T_j\}, & \text{否则} \end{cases}$$

算法 1 在时刻 t , 一个 CCPN 使能变迁集 $T^{sw}(t)$ 和可供给位置集 $P^{sw}(t)$ 的计算

step 1 初始化 $X^0=(A^0, E^0)=0, k=0$

step 2 $A^{k+1}=G(A^k), E^{k+1}=G(E^k), X^{k+1}=(A^{k+1}, E^{k+1})$

step 3 若 $(A^{k+1}, E^{k+1}) \neq (A^k, E^k), k=k+1$, 转 step 2;

若 $(A^{k+1}, E^{k+1}) = (A^k, E^k)$, 转 step 4

step 4 输出 $X^{k+1}=(A^{k+1}, E^{k+1})$, 结束。

由算法 1 输出的结果, 易得

$$T^{sw}(t) = \{T_j \mid e_j^{k+1} = 1\}, P^{sw}(t) = \{p_i \mid a_i^{k+1} = 1\}$$

$$T^e(t) = \{T_j \mid e_j^{k+1} = 0\}, P^e(t) = \{p_i \mid a_i^{k+1} = 0\}$$

定义 4 冲突 $cf = \langle p_i, \{T_1, T_2, \dots, T_k\}, M(t) \rangle$ 是一个有效冲突, 如果下列 3 个条件被满足：

(1) $\langle p_i, \{T_1, T_2, \dots, T_k\} \rangle$ 是一个结构冲突, 且变迁 $T_1, T_2, \dots, T_k$ 由 $M(t)$ 使能;

(2) $m_i(t)=0$;

(3) $0 < \sum_{T_j \in \cdot p_i} v_j(t) < v_1' + v_2' + \dots + v_k'$, 其中: 若 $\cdot T_u = \{p_i\}$, 则 $V_u' = V_u, u=1, 2, \dots, k$; 若 $\cdot T_u = \{p_i, p_{i1}, p_{i2}, \dots, p_{is}\}$ 且存在某些 $p_{iv} \in \cdot T_u, m_{iv}(t)=0$, 则 $V_u' \leq V_u, u=1, 2, \dots, k$ 。

定义 5 设 $\Sigma_c(t) = (P, T; F, M(t), V)$ 为一个 CCPN, C 为其关联矩阵, $v(t)$ 表示 t 时刻变迁集 T 的瞬发速度向量, 则有：

$$\frac{dM(t)}{dt} = Cv(t) \quad (1)$$

式(1)称为 CCPN 的基本关系。 t 时刻以后的某段时间内, CCPN 的运行由标识 $M(t)$ 、瞬发速度向量 $v(t)$ 和基本关系所确定。

2 含有效冲突 CCPN 中变迁的瞬发速度计算

在 CCPN 中, t 时刻的有效冲突会造成不确定性, 解决有效冲突的方法是必须确定某种分配原则。本文基于有效冲突的两种解决方式——确定优先级或按比例分配流量, 给出计算各变迁瞬发速度的算法。

性质 1 若 $T_j = T^s(t)$, 则 $v_j(t) = V_j$ 。

定义 6 记 $Pre(p_i) = \sum_{k=1}^n C_{ik}^+ v_k(t)$,

$$Post(p_i) = \sum_{k=1}^n C_{ik}^- v_k(t), i=1, 2, \dots, m$$

其中, $C^+ = [C_{ij}^+]_{m \times n}$, $C^- = [C_{ij}^-]_{m \times n}$ 分别为输出关联矩阵和输入关联矩阵, $C = [C_{ij}]_{m \times n}$ 为关联矩阵, 且 $C_{ij} = C_{ij}^+ - C_{ij}^-$ 。令

$$B_i(t) = Pre(p_i) - Post(p_i)$$

$$B(t) = [B_1(t), B_2(t), \dots, B_m(t)]^T$$

则称 $B_i(t)$ 为时刻 t 位置 p_i 的动态平衡量, $B(t)$ 为时刻 t 位置集 P 的动态平衡向量。显然有 $\frac{dM(t)}{dt} = Cv(t) = B(t)$ 。

性质 2 若 $T_j \in T^w(t)$, 且 T_j 不在有效冲突中, 则 $v_j(t) = \min\{V_j, \min_i [B_i(t) + v_j(t)]\}$, 其中 $p_i \in \cdot T_j \cap P^w(t)$ 。

性质 3 若 $T_j \in T^e(t)$, 则 $v_j(t) = 0$ 且 $v_j(t + \Delta t) \equiv 0, \forall \Delta t > 0$ 。

性质 4 设 $cf = \langle p_i, \{T_1, T_2, \dots, T_k\}, M(t) \rangle$ 为一个有效冲突。

(1) 若变迁 $T_1, T_2, \dots, T_k$ 是按优先级从高到低排序, 则 $v_j(t) = \min\{V_j, \min_i [B_i(t) + v_j(t)]\}$, 其中 $p_i \in \cdot T_j \cap P^w(t) (j=1, 2, \dots, k)$, 计算时优先级为从高到低。

(2) 若变迁 $T_1, T_2, \dots, T_k$ 是按比例分配流量, 则 $\frac{v_1(t)}{V_1} =$

$$\frac{v_2(t)}{V_2} = \dots = \frac{v_k(t)}{V_k}, \text{且满足 } v_1(t) + v_2(t) + \dots + v_k(t) = \sum_{T_j \in \cdot p_i} v_j(t)。$$

设在时刻 t , 由算法 1 计算得到的使能变迁集为 $T^{sw}(t) = \{T_1, T_2, \dots, T_s\}$, 其中 $s = |T^{sw}(t)|$, $v(t) = [v_1(t), v_2(t), \dots, v_s(t)]^T$ 是使能变迁的瞬发速度向量, 方程组形式是 $v = G'(v)$ 。迭代方式 G' 定义如下：

$$v_j^{r+1} = \begin{cases} V_j, & T_j \in T^s(t); \\ \min\{V_j, \min_i [\sum_{k=1}^{j-1} C_{ik} v_k^{r+1} + \sum_{k=j}^s C_{ik} v_k^r + v_j^r]\}, & T_j \in T^w(t) \end{cases}$$

其中, $p_i \in \cdot T_j \cap P^w(t)$, v_j^r 是第 r 次迭代得到的速度 $v_j(t)$ 的值, $v^r = (v_1^r, v_2^r, \dots, v_s^r)$ 是第 r 次迭代得到的速度向量 $v(t)$ 的值。

需要特别说明的是, 当变迁 T_j 处于有效冲突时, 应根据有效冲突的解决策略——优先级或按比例分配流量, 由性质 4 计算相应的迭代值。

算法2 t 时刻使能变迁的瞬发速度计算

step 1 初始化 $v^0 = (0, 0, \dots, 0)$, $r = 0$
 step 2 由迭代方式 G' 计算 $v^{r+1} = (v_1^{r+1}, v_2^{r+1}, \dots, v_s^{r+1})$
 step 3 若 $v^{r+1} \neq v^r$, 则 $r = r + 1$, 转 step 2; 若 $v^{r+1} = v^r$, 转 step 4
 step 4 输出 $v^{r+1} = (v_1^{r+1}, v_2^{r+1}, \dots, v_s^{r+1})$

从而对于 $T_j \in T^{sw}(t)$, $v_j(t) = v_j^{r+1}$ ($j = 1, 2, \dots, s$); 而由性质3知, 若 $T_j \in T^w(t)$, 则 $v_j(t) = 0$, 因而任何变迁的瞬发速度都可求出。

3 含有效冲突 CCPN 的运行区间及其划分

定义7 时间区间 $[t_{k-1}, t_k]$ 称为含有效冲突 CCPN 的一个运行区间, 当且仅当满足下面的两个条件:

(1) $\forall t \in [t_{k-1}, t_k]$, 均有 $v(t) = v(t_{k-1})$, 即隐含有 $B(t) = B(t_{k-1})$;

(2) $\Delta_k = t_k - t_{k-1}$

$$= \begin{cases} \min_i \left\{ \frac{m_i(t_{k-1})}{-B_i(t_{k-1})} \right\}, & \text{当 } \{p_i | p_i \in P^s(t_{k-1}) \text{ 且 } B_i(t_{k-1}) < 0\} \neq \emptyset \\ +\infty, & \text{当 } \{p_i | p_i \in P^s(t_{k-1}) \text{ 且 } B_i(t_{k-1}) < 0\} = \emptyset \end{cases}$$

由定义7可知, 若存在某个运行区间 $[t_{k-1}, t_k]$, 使得 $\{p_i | p_i \in P^s(t_{k-1}) \text{ 且 } B_i(t_{k-1}) < 0\} = \emptyset$, 则有 $[t_{k-1}, t_k] = [t_{k-1}, +\infty)$ 为含有效冲突 CCPN 的一个无限运行区间, 且 $\forall t \in [t_{k-1}, +\infty)$, 均有 $v(t) = v(t_{k-1})$, 这说明在 t_{k-1} 时刻以后的所有时间内, 瞬发速度向量将不再变化, 称该 $v(t_{k-1})$ 为含有效冲突 CCPN 的稳定速度状态, 特别记作 $v(t_{end}) = v(t_{k-1})$, 并称 $[t_{end}, +\infty)$ 为其稳定运行区间; 若对任何一个运行区间 $[t_{k-1}, t_k]$, 均有 $\{p_i | p_i \in P^s(t_{k-1}) \text{ 且 } B_i(t_{k-1}) < 0\} \neq \emptyset$, 则含有效冲突 CCPN 不存在稳定速度状态和稳定运行区间。

性质5^[2] 一切不存在有效冲突的 CCPN 在有限时间内必收敛到一个稳定速度状态 $v(t_{end})$ 。

引理1 若 $v(t_{end})$ 为 CCPN 的稳定速度状态, 则 $\forall p_i \in P$, 均有 $B_i(t_{end}) \geq 0$, 从而 $B(t_{end}) \geq 0$ 。

证明: 因 $v(t_{end})$ 为 CCPN 的稳定速度状态, 所以 $[t_{end}, +\infty)$ 为 CCPN 的稳定运行区间, 由定义7可知, 显然有 $\{p_i | p_i \in P^s(t_{end}) \text{ 且 } B_i(t_{end}) < 0\} = \emptyset$, 即 $\forall p_i \in P^s(t_{end})$, 均有 $B_i(t_{end}) \geq 0$; 而 $\forall p_i \in P^w(t_{end})$, 显然有 $B_i(t_{end}) \geq 0$; 对 $\forall p_i \in P^{rs}(t_{end})$, 有 $B_i(t_{end}) = 0$ 。又因在时刻 t_{end} , 有 $P = P^s(t_{end}) + P^w(t_{end}) + P^{rs}(t_{end})$, 故 $\forall p_i \in P$, 均有 $B_i(t_{end}) \geq 0$, 从而有

$$\begin{aligned} t: [0, +\infty) &= [0, t_1) + [t_1, t_2) + \dots + [t_{k-1}, t_k) + [t_k, t_{k+1}) + \dots \\ &\quad \downarrow \quad \quad \quad \downarrow \\ DG[\sum_c(t)]: & \quad T^{(0)} \leftrightarrow M(0), t_0 = 0 \quad T^{(1)} \leftrightarrow M(t_1), t_1 \\ & \quad p^{(0)} \leftrightarrow v(t) = v(0) \quad p^{(1)} \leftrightarrow v(t) = v(t_1) \\ & \quad \text{弧 } T^{(0)} \rightarrow p^{(0)} \quad \text{弧 } p^{(0)} \rightarrow T^{(1)} \\ & \quad \quad \quad \quad \quad \quad \quad T^{(1)} \rightarrow p^{(1)} \end{aligned}$$

由引理1可知, 一旦 $B(t_k) \geq 0$, 即可得演变图的终端结点 $p^{(k)} \leftrightarrow v(t_k)$, 此时 $DG[\sum_c(t)]$ 的终端结点对应着 CCPN 的稳定速度状态 $v(t_{end})$; 如果不存在 $B(t_k) \geq 0$, 则 $DG[\sum_c(t)]$ 无终端结点, 但因 $\sum_c(t)$ 的有界性, 必存在某个 $M(t_i)$ ($i = 0, 1, \dots, k-1$) 使得 $M(t_k) = M(t_i)$, 从而 $\sum_c(t)$ 进入周期性的运行。

算法3 含有效冲突且有界 CCPN 的演变图构造算法

输入: $\sum_c(t) = (P, T; F, M(0), V)$

$$B(t_{end}) = [B_1(t_{end}), B_2(t_{end}), \dots, B_m(t_{end})]^T \geq 0.$$

设 $\sum_c(t) = (P, T; F, M(0), V)$ 为一个 CCPN, $M(0)$ 是在初始时刻 $t = 0$ 的初始标识, 则在初始时刻 $t = t_0 = 0$, 可由 $M(0)$ 确定 $P^s(0)$, 由算法1和算法2计算出 $v(0)$, 进而根据基本关系 $\frac{dM(t)}{dt} = Cv(t) = B(t)$ 得到 $B(0)$, 这样由定义7可确定第一个运行区间 $[t_0, t_1) = [0, t_1)$ 。若 $t_1 = +\infty$, 则得稳定的运行区间为 $[0, +\infty)$ 及稳定的速度状态 $v(t_{end}) = v(0)$; 若 $t_1 < +\infty$, 则可得一个有限的运行区间 $[0, t_1)$, 由基本关系决定在运行区间 $[0, t_1)$ 内标识的变化, 然后由 $t = t_1$ 时刻的标识 $M(t_1)$ 重复上述过程得第二个运行区间为 $[t_1, t_2)$ 等。对于含有效冲突的 CCPN, 如果有效冲突预先解决, 最后可能收敛到稳定的速度状态 $v(t_{end})$, 进而得稳定的运行区间 $[t_{end}, +\infty)$, 也可能不存在稳定速度状态和稳定运行区间, 但可以按照定义7计算得到各个运行区间。这样含有效冲突 CCPN 的运行就被运行区间所划分:

$$\begin{aligned} t \in [0, +\infty) &= [0, t_1) + [t_1, t_2) + \dots + [t_{k-1}, t_k) + [t_k, t_{k+1}) + \dots \\ &\quad \downarrow \quad \quad \quad \downarrow \quad \quad \quad \downarrow \quad \quad \quad \downarrow \quad \quad \quad \downarrow \\ &\quad v(t) \quad v(0) \quad v(t_1) \quad v(t_{k-1}) \quad v(t_k) \end{aligned}$$

在每一个运行区间 $[t_{k-1}, t_k]$ 上, 瞬发速度向量保持恒定, 而标识的变化由 $\frac{dM(t)}{dt} = Cv(t) = B(t)$ ($t_{k-1} \leq t < t_k$) 确定, 即得 $\frac{dM(t)}{dt} = Cv(t_{k-1}) = B(t_{k-1})$, 从而 $M(t_k) = M(t_{k-1}) + (t_k - t_{k-1})B(t_{k-1})$ 。

4 含有效冲突且有界 CCPN 的演变图及其构造算法

在一个含有效冲突且有界 CCPN $\sum_c(t) = (P, T; F, M(0), V)$ 中, 一般来说, 瞬发速度向量 $v(t)$ 是不连续的, 但标识 $M(t)$ 却是时间的连续函数, 因而在一般情况下难以构造其可达标识图。然而 CCPN 的运行特征向量是瞬发速度向量, 这一向量在某一运行区间上保持不变, 因而由运行区间的划分可以构造其演变图, 通过演变图可以分析 CCPN 的标识和速度状态的变化, 从而得知其有关性质。

CCPN 的演变图 $DG[\sum_c(t)]$ 被表示成 Petri 网的形式, 其中 $DG[\sum_c(t)]$ 中变迁 $T^{(k-1)}$ 的设置应表明第 k 个运行区间 $[t_{k-1}, t_k]$ 的开始时刻 t_{k-1} 及 $M(t_{k-1})$, $DG[\sum_c(t)]$ 中的位置 $p^{(k-1)}$ 表明第 k 个运行区间 $[t_{k-1}, t_k]$ 上的速度状态 $v(t) = v(t_{k-1})$ 。具体示意如下:

$$\begin{aligned} & + \dots + [t_{k-1}, t_k) + [t_k, t_{k+1}) + \dots \\ & \quad \downarrow \quad \quad \quad \downarrow \\ & T^{(k-1)} \leftrightarrow M(t_{k-1}), t_{k-1} \quad T^{(k)} \leftrightarrow M(t_k), t_k \\ & p^{(k-1)} \leftrightarrow v(t) = v(t_{k-1}) \quad p^{(k)} \leftrightarrow v(t) = v(t_k) \\ & \text{弧 } p^{(k-2)} \rightarrow T^{(k-1)} \quad \text{弧 } p^{(k-1)} \rightarrow T^{(k)} \\ & T^{(k-1)} \rightarrow p^{(k-1)} \quad T^{(k)} \rightarrow p^{(k)} \end{aligned}$$

输出: $DG[\sum_c(t)]$

初始化: $t_0 = 0, M(t_0) = M(0), k = 1$

step 0 以 $T^{(k-1)}$ 作为 $DG[\sum_c(t)]$ 的一个变迁, 并标明 $M(t_{k-1})$, t_{k-1} 。

step 1 For $M(t_{k-1})$ do

1) 确定 $P^s(t_{k-1}) = \{p_i | m_i(t_{k-1}) > 0\}$

2) 调用算法1计算 $T^{sw}(t_{k-1})$ 及 $T^{ne}(t_{k-1})$

3) 调用算法2计算 $v(t_{k-1})$ 。若 $T_j \in T^{ne}(t_{k-1})$, 则 $v_j(t_{k-1}) = 0$; 若

T_j 是有效冲突中的变迁, 应标明是按优先级解决还是按比例分配流量

4) 以 $p^{(k-1)}$ 作为 $DG[\Sigma_c(t)]$ 的一个位置, 并标明 $v(t) = v(t_{k-1})$, 从 $T^{(k-1)}$ 到 $p^{(k-1)}$ 画一条有向弧

5) 计算 $B(t_{k-1}) = Cv(t_{k-1})$

step 2 If $B(t_{k-1}) \geq 0$, 转 step 6; 否则, 转 step 3

step 3 $t_k = t_{k-1} + \min_i \{ \frac{m_i(t_{k-1})}{-B_i(t_{k-1})} \mid p_i \in P^s(t_{k-1}) \text{ 且 } B_i(t_{k-1}) < 0 \}$

$M(t_k) = M(t_{k-1}) + (t_k - t_{k-1})B(t_{k-1})$

step 4 If $M(t_k) \neq M(t_i), \forall i = 0, 1, \dots, k-1$, 则以 $T^{(k)}$ 为 $DG[\Sigma_c(t)]$ 的一个变迁, 并标明 $M(t_k), t_k$, 从 $p^{(k-1)}$ 到 $T^{(k)}$ 画一条有向弧, 转 step 5; If $M(t_k) = M(t_i), \exists i = 0, 1, \dots, k-1$, 则从 $p^{(k-1)}$ 到 $T^{(i)}$ 画一条有向弧, 转 step 6

step 5 $k = k + 1$, 转 step 1

step 6 输出 $DG[\Sigma_c(t)]$

5 应用举例

对于图 1 所示的有界 $\Sigma_c(t)$, 它含有一个有效冲突 $cf = \langle p_2, \{T_2, T_3\}, M(t) \rangle$, 基于有效冲突的两种解决方式应用算法 3 分别构造其演变图。在初始时刻 $t_0 = 0$ 时, $M(0) = [1, 0, 0, 0]^T$ 。

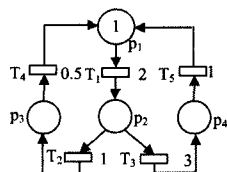


图 1 含有效冲突的 $\Sigma_c(t)$

(1) 采用优先级解决图 1 中的有效冲突, 设变迁 T_2 的优先级高于 T_3 , 应用算法 3, 则得其演变图如图 2 所示。

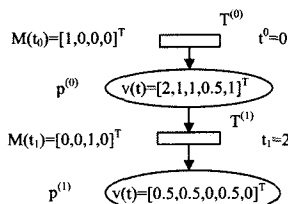


图 2 用优先级解决有效冲突的 $DG[\Sigma_c(t)]$

(2) 采用按比例分配流量解决图 1 中的有效冲突, 应用算法 3, 则得其演变图如图 3 所示。

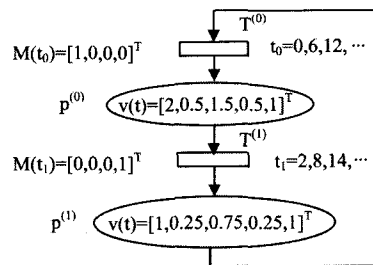


图 3 按比例分配流量解决有效冲突的 $DG[\Sigma_c(t)]$

参考文献

- [1] David R, Alla H. 佩特利网和逻辑控制器图形式工具[M]. 黄建文, 赵不贿, 译. 北京: 机械工业出版社, 1996
- [2] David R, Alla H. Petri Nets and Grafcet: Tools for Modeling Discrete Event System[M]. London: Prentice-Hall, 1992
- [3] Zhao Yi-jun. Properties and Determining Methods of Continuous Petri Nets[D]. Taian: Shandong Mining Institute and Technology, 1998(in Chinese)
赵义军. 连续 Petri 网的性质及判定方法[D]. 泰安: 山东矿业学院, 1998
- [4] Zhao Yi-jun, Wang Pei-liang. Developing Graph and Generating Algorithm of CCPN[J]. Computer Science, 1999, 26(6Suppl): 1-6(in Chinese)
赵义军, 王培良. CCPN 的演变图及其构造算法[J]. 计算机科学, 1999, 26(6 增刊): 1-6
- [5] Peter J L. Petri 网理论与系统模拟[M]. 吴哲辉, 译. 徐州: 中国矿业大学出版社, 1989
- [6] 吴哲辉. Petri 网导论[M]. 北京: 机械工业出版社, 2006
- [7] 袁崇义. Petri 网原理[M]. 北京: 电子工业出版社, 2005
- [8] David R, Alla H. Petri Net for Modeling of Dynamic Systems-A Survey[J]. Automatica, 1994, 30(2): 36-41
- [9] Murata T. Petri Nets: Properties, Analysis and Applications [J]. Proceedings of the IEEE, 1989, 77(4): 541-581
- [10] Li Zhi-jia, Hu Xiang, Jiao Li, et al. Performance Evaluation of Job Scheduling and InfiniBand Network Interconnection in High Performance Computing System Based on Stochastic Petri Nets [J]. Computer Science, 2015, 42(1): 31-37(in Chinese)
李智佳, 胡翔, 焦莉, 等. 基于随机 Petri 网的高性能计算系统作业调度及 InfiniBand 网络互连的性能分析[J]. 计算机科学, 2015, 42(1): 31-37

(上接第 82 页)

- [21] Ramirez-Marquez J E, Coit D W. Composite Importance Measures for Multi-State Systems with Multi-State Components[J]. IEEE Transactions on Reliability, 2005, 54(3): 321-331
- [22] Looney C G. Fuzzy Petri nets and applications[C]// Tzafestas S G, et al. Fuzzy Reasoning in Information. Decision and Control Systems, 1994: 511-527
- [23] Barzegar S, Davoudpour M, Meybodi M R. Formalized learning automata with adaptive fuzzy coloured Petri net; an application specific to managing traffic signals[J]. Scientia Iranica, 2011, 18(3): 554-565

- [24] Wang Xiao-ming, Zhang Li-chen, Dou Wen-yang. Fuzzy colored time Petri net and termination analysis for fuzzy Event-Condition-Action rules[J]. Information Sciences, 2013, 232(20): 225-240
- [25] Hanafi J. Modeling of Collection Strategies for End-of-Life Products using Colored Petri Net[D]. Australian: The University of New South Wales, 2008
- [26] Liu X, Zou J, Zhang X. Investigations on the reliability of storage battery array with high capability[C]// 2008 International Conference on Condition Monitoring and Diagnosis. Beijing, China, 2008: 735-737