

混合 Petri 网的流体随机 Petri 网模型

黄镇谨^{1,2} 陆阳^{1,3} 杨娟¹ 方欢¹

(合肥工业大学计算机与信息学院 合肥 230009)¹ (广西工学院计算机工程系 柳州 545006)²

(安徽省矿山物联网与安全监控技术重点实验室 合肥 230088)³

摘要 混合 Petri 网和流体随机 Petri 网作为混合系统的建模方法,其内在的建模机制和分析方法不尽相同,而且这两种机制远未发展成熟。分析并建立它们之间相互转换的方法有助于机制本身的发展完善,并利用对方的建模原语和分析方法对系统进行多角度的分析。提出了一阶混合 Petri 网转换成流体随机 Petri 网的形式化方法及转换后变迁合并的方法,并给出了转换和合并方法的正确性证明,最后结合实例简要阐述了这两种方法的实现过程。

关键词 Petri 网,模型转换,混合 Petri 网,流体随机 Petri 网,变迁合并

中图分类号 TP393 **文献标识码** A

Fluid Stochastic Petri Net Model for Hybrid Petri Net

HUANG Zhen-jin^{1,2} LU Yang^{1,3} YANG Juan¹ FANG Huan¹

(School of Computer & Information, Hefei University of Technology, Hefei 230009, China)¹

(Department of Computer Engineering, Guangxi University of Technology, Liuzhou 545006, China)²

(The Anhui Provincial Key Laboratory of Mine IoT and Mine Safety Supervisory Control, Hefei 230088, China)³

Abstract Hybrid Petri net and fluid stochastic Petri are two modeling methods for hybrid system. Their modeling mechanism and analysis method are different and the two modeling formalisms are still evolving. Through converting for each other, one modeling method can use the others modeling primitives and analytical method for analyzing system, which is helpful for the future development of them. The formal conversion from hybrid Petri net to fluid stochastic Petri net and a transition merge method after conversion were proposed. Also, the effectiveness of proposed conversion and merge was proofed. At the end of paper, a case was illustrated for describing the process of conversion and merge.

Keywords Petri net, Model conversion, Hybrid petri net, Fluid stochastic petri net, Transition merge

1 引言

现实中很多系统都具有离散特征和连续特征,而且离散部分和连续部分相互作用。对这类系统的建模既需要离散状态变量,又需要连续状态变量。扩充了时间因素的时间 Petri 网,可用于对复杂系统的性能和可靠性进行定性和定量分析,然而传统的时间 Petri 网由于其离散的本质很难用于混合系统的建模。随着系统规模的增大,表征系统标识可达集的状态空间会呈指数增长,导致状态空间爆炸问题,而通过把具有大状态空间的小粒度参数流体化,可以降低系统的状态空间。为了解决混合系统的建模和状态空间爆炸问题,流体随机 Petri 网(Flow Stochastic Petri Net, FSPN)和混合 Petri 网(Hybrid Petri Net, HPN)先后被提出,并得到了成功应用^[1]。

K. S. Trivedi 和 V. G. Kulkarni 首先在文献[2]中引入了流体随机 Petri 网的概念。文献[3]和文献[4]对 FSPN 的基

本语义进行了扩展,添加了使能函数和跳跃弧,使 FSPN 具备了模拟 HPN 的能力。H. A11a 和 R. David 等提出连续 Petri 网,并将其拓展为混合 Petri 网,混合 Petri 网由连续 Petri 网和离散 Petri 网两部分组成^[5]。从形式上来看,这两种模型都能够对混合系统进行建模,并用于解决状态空间爆炸问题。从性能分析的角度来看,HPN 只能通过仿真的方法获得系统的性能指标,当连续库所较少时,FSPN 可以通过建立动态方程,采用数值方法求解系统性能指标,同时 FSPN 可以用于非马尔科夫随机系统然的建模。文献[1,6]对这两类模型的不同点、优缺点和适用范围进行了分析比较。M. Becher 和 T. Bessey 首先在文献[7]指出基本的 FSPN 模型可以转换成 HPN,扩展了模型语义的 FSPN 也可以用来模拟 HPN,然而文中并没有给出转换的形式化描述。在文献[7]的基础上,廖伟志等在文献[8]给出了由 FSPN 转换成 HPN 的形式化方法,并给出了正确性证明。采用文献[9]的转换方法,可以用混合

收稿日期:2011-09-13 返修日期:2011-12-29 本文受国家自然科学基金项目(60873195, 61070220),高等学校博士点基金项目(20090111110002)资助。

黄镇谨(1975-),男,博士生,主要研究方向为计算机控制、形式化技术,E-mail:schzj@163.com;陆阳(1967-),教授,博士生导师,主要研究方向为人工智能、计算机控制和传感器网络;杨娟(1983-),女,博士生,主要研究方向为人工智能、神经网络;方欢(1982-),女,博士生,主要研究方向为 Petri 网、形式化方法。

自动机的验证技术和工具来实现 FSPN 的验证。利用 FSPN 的扩展定义,基于以上文献,本文给出混合 Petri 网转换成流体随机 Petri 网的形式化方法,同时提出了转换后模型的变迁合并算法,并对转换方法和合并算法的正确性进行了证明。

本文第 2 节介绍扩展 FSPN 和 HPN 的形式化定义;第 3 节详细讨论 HPN 转换成 FSPN 的形式化描述,给出变迁合并的方法,并对转换方法和合并算法的正确性进行证明;第 4 节结合实例简要阐述这两种方法的实现过程;最后是结论。

2 HPN 与 FSPN 的形式化定义

下面的定义中以 N, R, R_0 分别表示非负整数、实数和非负实数。

定义 1^[10] 流体随机 Petri 网为十元组 $(P, T, A, B, g, h, w, \lambda, f, M_0)$, 各元素含义如下:

$P = P_D \cup P_C$ 表示库所集,其中 P_D 为离散库所集, P_C 为连续库所集。离散库所 $p \in P_D$ 用单圆圈表示并且包含非负整数的托肯数 $m_p \in N$, 连续标识 $q \in P_C$ 用双圆圈表示且包含流体 $x_q \in R$, FSPN 的完整标识由矢量对 $(m, x) \in S$ 表示, S 表示标识空间, S_D 表示标识空间的离散部分。标识随时间变化,形式上用二维随机过程 $\{(m(t), x(t)); t \geq 0\}$ 表示。

$T = T_I \cup T_E \cup T_G$ 是变迁集。其中 T_I 是瞬时变迁集, T_E 是指数变迁集,瞬时变迁用细实线表示,其发射时间为 0,指数变迁用小矩形表示,发射时间是符合指数分布的随机变量。 T_G 为发射时间符合任意分布的变迁集。

$A = A_d \cup A_c \cup A_h$ 是有向弧集。 $A_d: (P_D \times T) \cup (T \times P_D) \times S_D \rightarrow N$ 是离散弧, $A_c: (P_C \times T_E) \cup (T_E \times P_C) \times S \rightarrow R$ 是连续弧集,每个连续弧均有一个可依赖于标识的流体流动速度。 $A_h \subseteq (P_C \times T) \cup (T \times P_C)$ 是流体跳跃弧集,跳跃弧连接连续库所和变迁。

B 定义连续库所标识的上界,连续库所标识的下界隐含为 0。

$g: T \times S \rightarrow \{0, 1\}$ 为每个变迁的使能条件函数。

$h: A_h \rightarrow R \cup \{*\}$ 是跳跃弧的跳跃高度,跳跃高度密度函数为 $b(\cdot)$ 。 $\{*\}$ 标识清空弧。

$w: T_I \times S_D \rightarrow R$ 表示瞬时变迁的权重函数。

$\lambda: T \times S \rightarrow R \cup \{\infty\}$ 定义变迁的瞬时发射率,瞬时变迁的瞬时发射率为无穷大,指数变迁的发射率可以是离散和连续标识的函数。

$f: ((P_C \times T_E) \cup (T_E \times P_C)) \times S \rightarrow R_0$ 表示连续弧中依赖于标识的流体流动速度。

$M_0 = (m_0, x_0)$ 为模型的初始标识。

定义 2^[11] 混合 Petri 网是六元组 $(P, T, pre, post, D, C)$, 各元组定义如下:

$P = P_D \cup P_C$ 由离散库所 P_D 和连续库所 P_C 组成,其中离散库所用单圆圈表示,连续库所用双圆圈表示。

$T = T_d \cup T_c$ 分成离散变迁 T_d 和连续变迁 T_c (用白盒表示)。离散变迁 $T_d = T_I \cup T_D \cup T_E$ 由 3 部分组成,其中 T_I 为瞬时变迁(单横线表示), T_D 为常数延时变迁(双盒子表示), T_E 为指数分布延时变迁(黑盒表示)。

$pre: (P_D \times T) \cup (P_C \times T) \rightarrow N$ 为输入关联映射; $post: (P_D \times T) \cup (P_C \times T) \rightarrow R_0$ 为输出关联映射。对于 $t \in T_c \wedge p \in P_D$, 有 $pre(p, t) = post(p, t)$ 。

$D: T_d \setminus T_I \rightarrow R_0$ 为时间延时函数。对于 $t_i \in T_D$, 其引发延时 $\delta_i = D(t_i)$, 对于 $t_i \in T_E$, 其平均引发速率 $\lambda_i = D(t_i)$ 。

$C: T_c \rightarrow R_0 \times (R \cup \{\infty\})$ 为连续变迁的引发速率区间。 $\forall t_i \in T_c, C(t_i) = (V_i^l, V_i^u)$, 其中 $V_i^l \leq V_i^u$, V_i^l 为最小引发速率, V_i^u 为最大引发速率。最小引发速率隐含条件下为 0。

3 混合 Petri 网的 FSPN 变换

扩充了语法和语义的 FSPN 可以对混合 Petri 网进行模拟,下面给出其形式化的转换方法。设 $P_D |_{HPN}, P_C |_{HPN}$ 分别表示混合 Petri 网的离散库所和连续库所,并令 $T_C |_{HPN}, T_D |_{HPN} = T_I |_{HPN} + T_d |_{HPN} + T_e |_{HPN}$ 表示连续变迁和离散变迁,其中 $T_I |_{HPN}, T_d |_{HPN}, T_e |_{HPN}$ 分别为离散变迁中的瞬时变迁、固定时间变迁和指数变迁。同样,令 $P_D |_{FSPN}, P_C |_{FSPN}$ 为流体随机 Petri 网的离散库所和连续库所, $T_I |_{FSPN}, T_e |_{FSPN}, T_g |_{FSPN}$ 分别为瞬时变迁、指数变迁和任意分布变迁集。

3.1 转换过程

步骤 1 库所、变迁变换

$$\bullet P_D |_{FSPN} \leftarrow P_D |_{HPN}, P_C |_{FSPN} \leftarrow P_C |_{HPN}$$

$$\bullet T_I |_{FSPN} \leftarrow T_I |_{HPN}, T_e |_{FSPN} \leftarrow T_e |_{HPN}, T_g |_{FSPN} \leftarrow T_d |_{HPN}, T_e |_{FSPN} \leftarrow T_c |_{HPN}$$

步骤 2 建立变迁和库所之间的弧与权重

设流体随机 Petri 网中跳跃弧的跳跃高度密度函数为 $b(\cdot)$, $F(A_c)$ 为连续弧 A_c 在 FSPN 中流体流动的速度。 $G(t_i)$ 为变迁 t_i 的使能函数,默认情况下为 0。

(1) $(\forall t_i \in T_I |_{FSPN}) \vee (\forall t_i \in T_e |_{FSPN} \wedge t_i \in T_E |_{HPN}) \vee (\forall t_i \in T_g |_{FSPN})$ 则:

$t_i |_{FSPN} \leftarrow t_i |_{HPN}$, 若 $\forall \rho \in (t_i |_{HPN} \cap P_D |_{HPN})$, 令 $A_d = \rho \times t_i, W(\rho, t_i |_{FSPN}) \leftarrow pre(\rho, t_i |_{HPN}), \forall \rho \in \cap P_C |_{HPN}$, 令 $A_h = \rho \times t_i, b(A_h) = pre(\rho, t_i |_{HPN}); t_i' |_{FSPN} \leftarrow t_i' |_{HPN}$, 若 $\forall \rho \in (t_i' |_{HPN} \cap P_D |_{HPN})$, 令 $A_d = t_i \times \rho, W(t_i |_{FSPN}, \rho) \leftarrow post(\rho, t_i |_{HPN}), t_i' |_{FSPN} \leftarrow (t_i' |_{HPN} \cap P_C |_{HPN}), A_h = t_i \times \rho, b(A_h) = post(\rho, t_i |_{HPN})$ 。

(2) $\forall t_i \in T_e |_{FSPN}$ 且 $t_i \in T_C |_{HPN}$ 则:

$t_i |_{FSPN} \leftarrow t_i |_{HPN}$, 若 $\forall \rho \in (t_i |_{HPN} \cap P_D |_{HPN})$, 令 $A_d = \rho \times t_i, W(\rho, t_i |_{FSPN}) \leftarrow pre(\rho, t_i |_{HPN}), \forall \rho \in (t_i |_{HPN} \cap P_C |_{HPN}), A_c = \rho \times t_i, F(A_c) = C(t_i |_{HPN}); t_i' |_{FSPN} \leftarrow (t_i' |_{HPN} \cap P_C |_{HPN}), \forall \rho \in t_i' |_{FSPN}$, 令 $A_c = t_i \times \rho, F(A_c) = C(t_i |_{HPN})$ 。

步骤 3 确定非瞬时变迁的引发时间

$\forall t_i \in T_g |_{FSPN}, \lambda(t_i) = 1/D(t_i |_{HPN}), (\forall t_i \in T_e |_{FSPN} \wedge t_i \in T_E |_{HPN}), \lambda(t_i |_{FSPN}) = D(t_i |_{HPN})$ 。

步骤 4 确定 FSPN 中变迁的使能函数

$\forall t_i \in T_I |_{FSPN}, G(t_i) = (\forall p \in (t_i \cap P_D |_{FSPN}), m(p) \geq \lambda(\forall p \in (t_i \cap P_C |_{FSPN}), m(p) \geq b(p, t_i)); (\forall t_i \in T_e |_{FSPN} \wedge t_i \in T_E |_{HPN}) \vee (\forall t_i \in T_g |_{FSPN}), G(t_i) = (\forall p \in (t_i \cap P_D |_{FSPN}), m(p) \geq W(p, t_i)) \wedge (\forall p \in (t_i \cap P_C |_{FSPN}), m(p) \geq b(p, t_i)); (\forall t_i \in T_e |_{FSPN}) \wedge (t_i \in T_C |_{HPN}), G(t_i) = (\forall p \in (t_i \cap P_C |_{FSPN}), m(p) > 0) \wedge (\forall p \in (t_i \cap P_D |_{FSPN}), m(p) \geq W(p, t_i))$ 。

3.2 变迁合并

经过以上步骤变换后的 FSPN 可以进一步简化。设 $Del(\cdot)$ 表示删除操作, $A(t_i \times \rho)$ 表示连接库所和变迁的弧, $T_\alpha = \{t_i | t_i \in T_e |_{FSPN} \wedge t_i \in T_C |_{HPN}\}$ $T_{ad} = T |_{FSPN} - T_\alpha - T_I |_{FSPN}$ 。过

程如下:

(1) 若 $T_{\alpha} \neq \emptyset$, 则从 T_{α} 中选择变迁 t_i , $T_{\alpha} = T_{\alpha} - t_i$, 否则转(6)。

(2) 从 T_{ad} 中选取变迁 $t_j = \{t_k | ((t_k \cap P_D |_{FSPN}) = (t_i \cap P_D |_{FSPN})) \wedge (((t_k \cap P_D |_{FSPN}) \cap (t_i \cap P_D |_{FSPN})) = \emptyset)\}$, 若 $t_j = \emptyset$, 转(1)。

(3) $t_i' \leftarrow t_j'$, $A(t_i \times t_j') \leftarrow A(t_j \times t_j')$, $W(t_i \times t_j') \leftarrow W(t_j \times t_j')$, $\lambda(t_i) = \lambda(t_j)$, $\text{Del}(A(t_j \times t_j'))$, $\text{Del}(A(t_i' \times t_j))$, $\text{Del}(t_j)$, $T_{ad} = T_{ad} - t_j$ 。

(4) 若 $T_{ad} \neq \emptyset$, 从 T_{ad} 中选取变迁 $t_j = \{t_k | (((t_k \cap P_D |_{FSPN}) = (t_i \cap P_D |_{FSPN})) \wedge (((t_k \cap P_D |_{FSPN}) \cap (t_i \cap P_D |_{FSPN})) = \emptyset)) \wedge (\lambda(t_k) = \lambda(t_i))\}$, 若 $t_j = \emptyset$, 转(1)。

(5) $t_i' \leftarrow t_j'$, $A(t_i \times t_j') \leftarrow A(t_j \times t_j')$, $W(t_i \times t_j') \leftarrow W(t_j \times t_j')$, $\text{Del}(A(t_j \times t_j'))$, $\text{Del}(A(t_i' \times t_j))$, $\text{Del}(t_j)$, $T_{ad} = T_{ad} - t_j$ 。转(1)。

(6) 结束。

3.3 正确性证明

下面首先定义 HPN 和 FSPN 的模型语义, 然后给出模型转换的正确性证明和变迁合并的正确性证明。

对于 HPN, 由于采用了与文献[8]一样的定义, 因此其模型语义也与文献[8]类似:

$\text{HPN}_1: \forall t_i \in T_I |_{HPN}, (\forall p \in (m(p) \geq \text{pre}(p, t_i)) \rightarrow t_i \text{ 使能 } \wedge t_i \text{ 引发 } \wedge \forall p \in (t_i' - t_i), m(p) = m(p) - \text{pre}(p, t_i) \wedge \forall p \in (t_i' - t_i), m(p) = m(p) + \text{post}(p, t_i)); \forall t_i \in (T_E |_{HPN} \vee T_G |_{HPN}) (\forall p \in (m(p) \geq \text{pre}(p, t_i)) \rightarrow t_i \text{ 使能 } \wedge \text{经过 } \lambda(t_i) \text{ 时间 } t_i \text{ 引发 } \wedge \forall p \in (t_i' - t_i), m(p) = m(p) - \text{pre}(p, t_i) \wedge \forall p \in (t_i' - t_i), m(p) = m(p) + \text{post}(p, t_i))$ 。

$\text{HPN}_2: \forall t_i \in T_C |_{HPN}, (\forall p \in (m(p) \geq \text{pre}(p, t_i)) \rightarrow t_i \text{ 使能 } \wedge t_i \text{ 以 } F(A_c) \text{ 速率引发 } \wedge \forall p \in ((t_i' - t_i) \cap P_C |_{HPN}), m(p) = m(p) + C(t_i) \cdot \delta \wedge \forall p \in ((t_i' - t_i) \cap P_C |_{HPN}), m(p) = m(p) - C(t_i) \cdot \delta \wedge \forall p \in ((t_i' - t_i) \cap P_D |_{HPN}), m(p) = m(p) + \text{pre}(p, t_i) \wedge \forall p \in ((t_i' - t_i) \cap P_D |_{HPN}), m(p) = m(p) - \text{post}(p, t_i))$ 。

由于本文采用了扩展定义 FSPN 来模拟 HPN, 因此其模型语义需重新定义。令 $T_d |_{FSPN} = \{t_i | \forall p \in t_i, \forall q \in t_i', A(p, t_i) \notin A_c \wedge A(t_i, q) \notin A_c\}$, $T |_{FSPN} - T_d |_{FSPN} = \{t_i | (\exists p \in t_i) \vee (\exists q \in t_i'), A(p, t_i) \in A_c \vee A(t_i, q) \in A_c\}$ 。

$\text{FSPN}_1: \forall t_i \in (T_d |_{FSPN} \cap T_i |_{FSPN}), (((\forall p \in (t_i' \cap P_D |_{FSPN}), (m(p) \geq W(p, t_i))) \wedge (\forall p \in (t_i' \cap P_C |_{FSPN}), \wedge m(p) > 0) \rightarrow t_i \text{ 使能 } \wedge t_i \text{ 引发 } \wedge \forall p \in ((t_i' - t_i) \cap P_D |_{FSPN}), m(p) = m(p) + W(p, t_i) \wedge \forall p \in ((t_i' - t_i) \cap P_D |_{FSPN}), m(p) = m(p) - W(p, t_i) \wedge \forall p \in ((t_i' - t_i) \cap P_C |_{FSPN}), m(p) = m(p) + b(p, t_i) \wedge \forall p \in ((t_i' - t_i) \cap P_C |_{FSPN}), m(p) = m(p) - b(p, t_i); \forall t_i \in (T_d |_{FSPN} - T_i |_{FSPN}), (((\forall p \in (t_i' \cap P_D |_{FSPN}), (m(p) \geq W(p, t_i))) \wedge (\forall p \in (t_i' \cap P_C |_{FSPN}), m(p) > 0) \rightarrow t_i \text{ 使能 } \wedge \text{经过 } \lambda(t_i) \text{ 时间 } t_i \text{ 引发 } \wedge \forall p \in ((t_i' - t_i) \cap P_D |_{FSPN}), m(p) = m(p) + W(p, t_i) \wedge \forall p \in ((t_i' - t_i) \cap P_D |_{FSPN}), m(p) = m(p) - W(p, t_i) \wedge \forall p \in ((t_i' - t_i) \cap P_C |_{FSPN}), m(p) = m(p) + b(p, t_i) \wedge \forall p \in ((t_i' - t_i) \cap P_C |_{FSPN}), m(p) = m(p) - b(p, t_i))$ 。

$\text{FSPN}_2: \forall t_i \in T |_{FSPN} - T_d |_{FSPN}, (\forall p \in (t_i \cap P_C |_{FSPN}), m(p) > 0) \wedge (\forall p \in (t_i \cap P_D |_{FSPN}), m(p) \geq W(p, t_i)) \rightarrow t_i \text{ 使能 } \wedge A_c(t_i)$ 以速率 $F(a_c)$ 引发 $\wedge \forall p \in (t_i' - t_i) \cap P_D |_{FSPN}, m(p) = m(p) + W(p, t_i) \wedge ((\forall p \in (t_i' - t_i) \cap P_C |_{FSPN}) \wedge A(p, t_i) \in A_h), m(p) = m(p) + b(t_i, p) \wedge ((\forall p \in (t_i' - t_i) \cap P_C |_{FSPN}) \wedge A(p, t_i) \in A_c), m(p) = m(p) + F(a_c) \cdot \delta \wedge ((\forall p \in (t_i' - t_i) \cap P_C |_{FSPN}) \wedge A(p, t_i) \in A_c), m(p) = m(p) - F(a_c) \cdot \delta$ 。

能 $\wedge A_c(t_i)$ 以速率 $F(a_c)$ 引发 $\wedge \forall p \in (t_i' - t_i) \cap P_D |_{FSPN}, m(p) = m(p) + W(p, t_i) \wedge ((\forall p \in (t_i' - t_i) \cap P_C |_{FSPN}) \wedge A(t_i, p) \in A_h), m(p) = m(p) + b(t_i, p) \wedge ((\forall p \in (t_i' - t_i) \cap P_C |_{FSPN}) \wedge A(p, t_i) \in A_h), m(p) = m(p) - b(t_i, p) \wedge ((\forall p \in (t_i' - t_i) \cap P_C |_{FSPN}) \wedge A(t_i, p) \in A_c), m(p) = m(p) + F(a_c) \cdot \delta \wedge ((\forall p \in (t_i' - t_i) \cap P_C |_{FSPN}) \wedge A(p, t_i) \in A_c), m(p) = m(p) - F(a_c) \cdot \delta$ 。

定理 1 采用上述方法转换而成的 FSPN 能正确表达原 HPN 的模型语义。

证明: 由转换算法知, 当 $t_i \in T_D |_{HPN}$ 时, 分别有 $T_i |_{FSPN} \leftarrow T_i |_{HPN}$, $T_e |_{FSPN} \leftarrow T_e |_{HPN}$, $T_g |_{FSPN} \leftarrow T_d |_{HPN}$, 且对于前件有 $t_i |_{FSPN} \leftarrow t_i |_{HPN}$, $\forall \rho \in (t_i |_{HPN} \cap P_D |_{HPN})$, 令 $A_d = \rho \times t_i$, $W(\rho, t_i |_{FSPN}) \leftarrow \text{pre}(\rho, t_i |_{HPN})$, $\forall \rho \in (t_i |_{HPN} \cap P_C |_{HPN})$, 令 $A_h = \rho \times t_i$, $b(A_h) = \text{pre}(\rho, t_i |_{HPN})$, 对于后件有 $t_i' |_{FSPN} \leftarrow t_i' |_{HPN}$, 若 $\forall \rho \in (t_i' |_{HPN} \cap P_D |_{HPN})$, 令 $A_d = t_i \times \rho$, $W(t_i |_{FSPN}, \rho) \leftarrow \text{post}(\rho, t_i |_{HPN})$, 否则令 $A_h = t_i \times \rho$, $b(A_h) = \text{post}(\rho, t_i |_{HPN})$ 。因此转换后得到的 FSPN 模型前后件及弧与原 HPN 模型一一对应, 不同之处在于 FSPN 中用跳跃弧模拟 HPN 中连续库所与离散变迁之间的弧。由转换算法中 $W(\rho, t_i |_{FSPN}) \leftarrow \text{pre}(\rho, t_i |_{HPN})$, $b(A_h) = \text{pre}(\rho, t_i |_{HPN})$ 及模型语义知两种模型的使能和引发条件是一致的。由 $W(t_i |_{FSPN}, \rho) \leftarrow \text{post}(\rho, t_i |_{HPN})$, $b(A_h) = \text{post}(\rho, t_i |_{HPN})$ 及模型语义亦可知两种模型中相应变迁引发后其标识的变化是相同的, FSPN 中的使能函数 $G(t_i) = m(P_C) > b(\cdot) \ \&\& \ m(P_D) \geq W(P_D, t_i)$ 确保从 HPN 中映射过来的连续库所能对变迁的使能产生影响。

当 $t_i \in T_C |_{HPN}$ 时, $T_e |_{FSPN} \leftarrow T_c |_{HPN}$, $t_i |_{FSPN} \leftarrow t_i |_{HPN}$, 前件由两部分组成: $\forall \rho \in (t_i |_{HPN} \cap P_D |_{HPN})$ 为离散库所集, 建立离散弧 $A_d = \rho \times t_i$, $W(\rho, t_i |_{FSPN}) \leftarrow \text{pre}(\rho, t_i |_{HPN})$; $\forall \rho \in (t_i |_{HPN} \cap P_C |_{HPN})$ 为连续库所, 建立连续弧 $A_c = \rho \times t_i$, $F(a_c) = C(t_i |_{HPN})$ 。对于 HPN 来说, 连续库所标识: $\forall p \in ((t_i - t_i') \cap P_C |_{HPN}), m(p) = m(p) - C(t_i) \cdot \delta \wedge \forall p \in ((t_i' - t_i) \cap P_C |_{HPN}), m(p) = m(p) - C(t_i) \cdot \delta$, 而在 FSPN 中, $((\forall p \in (t_i' - t_i) \cap P_C |_{FSPN}) \wedge A(t_i, p) \in A_c), m(p) = m(p) + F(a_c) \cdot \delta \wedge ((\forall p \in (t_i' - t_i) \cap P_C |_{FSPN}) \wedge A(p, t_i) \in A_c), m(p) = m(p) - F(a_c) \cdot \delta$, 由于 $C(t_i) = F(a_c)$, 因此经过变换后连续标识的变换是相同的。由于在 HPN 中 $\forall p \in P_D |_{HPN}$ 与 $t_i \in T_C |_{HPN}$, 要求 $\text{pre}(p, t_i) = \text{post}(p, t_i)$, 因此 HPN 中连续变迁的引发不改变离散库所的标识, 转换算法中 $\forall \rho \in (t_i |_{HPN} \cap P_D |_{HPN})$, 令 $A_d = \rho \times t_i$, 而无 $A_d = t_i \times \rho$, 因此转换成 FSPN 后连续变迁的引发同样不改变同一输入库所的标识, 反之亦然。纯粹的 HPN 连续网络(即只包含连续库所和连续变迁)转换成 FSPN 模型后, 变迁的使能通过使能函数 $G(t_i) = m(P_D) \geq W(P_D, t_i) \vee m(P_C) > 0$ 实现, 当 $m(P_C) > 0$ 时变迁使能。

定理 2 经过变迁合并后的模型与原模型具有相同的模型语义。

证明: 由 $t_j = t_i'$ 及 $\forall p \in (t_i = t_j)$, $W(p, t_i) = W(p, t_j)$ 可知, 合并前后变迁的前件及引发条件一致。再由 $t_i' \leftarrow t_j'$, $A(t_i \times t_j') \leftarrow A(t_j \times t_j')$, $W(t_i \times t_j') \leftarrow W(t_j \times t_j')$ 可知合并前后变迁的后件及权重相同。对于离散库所来说, 由于前后件及

其弧的权重相同,因此变迁引发后库所标识的变化是一样的。对于连续库所来说,合并前 $m(p) = m(p) \pm F(A_c) / \lambda(t_j)$, 合并后 $m(p) = m(p) \pm F(A_c) / \lambda(t_j)$, 由于合并前后有 $\lambda(t_j) = \lambda(t_j)$, 因此合并后连续库所标识的变化与合并前相同。

4 实例

下面以文献[5]的 HPN 模型为例,简要介绍模型转换过程。该模型用以描述一个简单的批到达处理系统。每一到达批次包括的部件数量为 60, 输入缓冲区容量为 400, 只有机器处于正常状态下才能加工工件, 当输出缓冲区中工件的数量超过 500 个时停止加工, 系统 HPN 模型如图 1 所示。采用前述的模型转换方法可以把该模型转换成 FSPN 模型, 结果如图 2 所示。

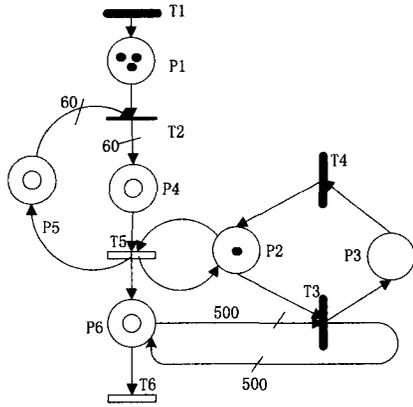


图 1 批到达处理系统的 HPN 模型

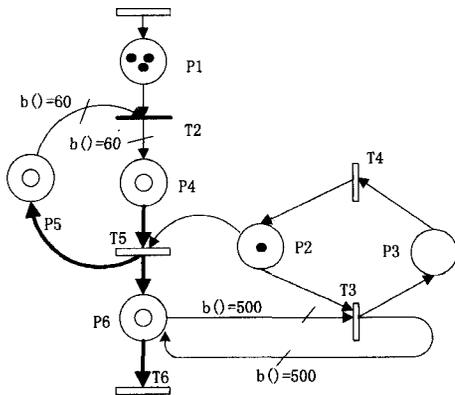


图 2 HPN 的流体随机 Petri 网模型

变换过程中, HPN 中连续库所与连续变迁 T_5 相连的弧在 FSPN 中用连续弧来替代, 弧中流体流动的速度等于 T_5 的引发速率, T_5 的使能由 P_2, P_4 决定, 当 T_3 引发时机器停止工作, 由于转换前后 T_3 的发射率相同, 因此转换前后流体库所标识的变化是一致的。HPN 中连续库所与离散变迁的弧用 FSPN 中跳跃高度密度函数为常数的跳跃弧模拟。由于 T_3 和 T_5 的前件不相同, 转换后的 FSPN 模型中这两者不能合并。图 3 和图 4 分别给出变迁合并前和变迁合并后的 FSPN 模型。图中假设未标注的弧权重相同, T_1 与 T_2 合并, T_3 与 T_4 合并。合并前 T_1 和 T_4 的引发将终止 T_2 和 T_3 的使能, 因此合并后的 T_2 和 T_3 的发射率分别等于 T_1 和 T_4 的发射率。

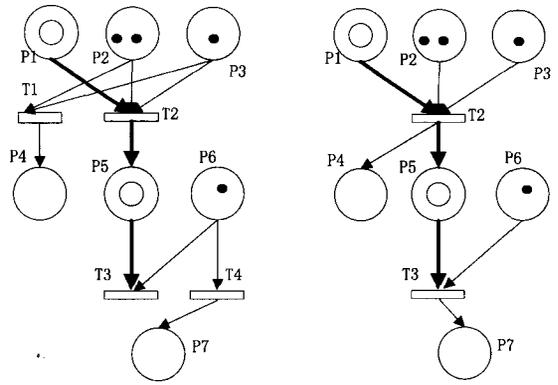


图 3 变迁合并前 FSPN 模型 图 4 变迁合并后 FSPN 模型

结束语 混合 Petri 网和流体随机 Petri 网都可用于混合系统的建模, 这两类建模方法目前都处在发展完善的阶段。二者内在的建模机制不同、各自有其相应的分析方法, 因此分析和比较这两类建模方法, 建立两者之间的转换机制, 有助于机制本身的进一步发展完善并从不同的角度对系统进行分析。本文给出了混合 Petri 网转换成流体随机 Petri 网的形式化描述, 通过转换可以借助 FSPN 的建模原语与其分析方法对 HPN 进行分析。进一步, 还可以利用 FSPN 作为桥梁沟通 HPN 与其他建模方法的联系。由于 FSPN 中与离散变迁相连接的跳跃弧的存在限制了现有的一些分析方法的使用^[10], 因此寻找更为有效的分析方法或对现有方法进行改进是下一步有待解决的问题。

参考文献

- [1] 卢光松, 葛运建. 流体随机 Petri 网与混合 Petri 网比较分析[J]. 小型微型计算机系统, 2005, 26(12): 2144-2146
- [2] Trivedi K, Kulkarni V. FSPNs; Fluid Stochastic Petri nets[C]// Application and Theory of Petri nets, 1993. Proc. 14th Intern. Conference. 1993, 691: 24-31
- [3] Horton G, Kulkarni V G. Fluid Stochastic Petri nets; Theory, Applications and Solution[J]. European Journal of Operation Research, 1998, 105(1): 184-201
- [4] Wolter K. Jump transitions in second order FSPNs[C]//Proc. of MASCOTS. Washington, DC, Oct. 1999: 156-163
- [5] David R, Alla H. Discrete, continuous, and hybrid petri nets [M]. Springer, 2010
- [6] Gribaudo M, Telek M. Fluid Model in Performance Analysis [C]// 7th International School on Formal Methods. 2007
- [7] Bechke M, Bessey T. Comparison of the Modeling Power of Fluid Stochastic Petri nets and Hybrid Petri Nets[C]//IEEE International Conference on Systems, Man and Cybernetics (SMC). 2002
- [8] 廖伟志, 古天龙, 董荣胜. 流体随机 Petri 网的一阶混杂 Petri 网模型[J]. 计算机科学, 2008, 35(1): 294-297
- [9] Tuffin B, Chen D S, Trivedi K. Comparison of Hybrid Systems and Fluid Stochastic Petri Nets[J]. Discrete Event Dynamic Systems, 2001, 11(1): 77-95
- [10] Wolter K. Performance and Dependability Modelling with Second Order Fluid Stochastic Petri Nets[D]. Technical University of Berlin, 1999
- [11] Balduzzi F, Giua A, Menga G. First-order Hybrid Petri Nets; a Model for Optimization and Control[J]. IEEE Transactions on Robotics And Automation, 2000, 16(12): 2144-2146