

时间约束 Petri 网及其可调度性分析与验证^{*}

宋 巍^{1,2}, 窦万春^{1,2+}, 刘茜萍^{1,2}

¹(南京大学 计算机科学与技术系,江苏 南京 210093)

²(计算机软件新技术国家重点实验室(南京大学),江苏 南京 210093)

Timing Constraint Petri Nets and Their Schedulability Analysis and Verification

SONG Wei^{1,2}, DOU Wan-Chun^{1,2+}, LIU Xi-Ping^{1,2}

¹(Department of Computer Science and Technology, Nanjing University, Nanjing 210093, China)

²(State Key Laboratory for Novel Software Technology (Nanjing University), Nanjing 210093, China)

+ Corresponding author: Phn: +86-25-83686399, E-mail: douwc@nju.edu.cn, <http://graphics.nju.edu.cn/douwanchun/index.htm>

Song W, Dou WC, Liu XP. Timing constraint Petri nets and their schedulability analysis and verification. *Journal of Software*, 2007,18(1):11–21. <http://www.jos.org.cn/1000-9825/18/11.htm>

Abstract: TCPN (timing constraint Petri nets) is an important kind of time-related Petri nets. Some basic concepts are redefined to enrich and perfect corresponding the theory of TCPN. Firstly, the rigorous definitions for weak and strong schedulability of a transition are redefined to correct the irrationality of the original ones. The verification method for the strong schedulability is presented in the form of a theorem. Secondly, the definition of schedulability of TCPN is given and the corresponding necessary and sufficient condition is proposed. Finally, the characteristic of TCPN along with Petri nets and other time-related Petri nets is analysed. It is of certain illumination and reference, both in research of time-related Petri nets and in modeling real-time systems.

Key words: time-related Petri nets; TCPN (timing constraint Petri nets); weak schedulability; strong schedulability; schedulability analysis

摘 要: 时间约束 Petri 网(timing constraint Petri nets,简称 TCPN)是一类重要的时间 Petri 网系统.对相关定义重新界定,丰富并完善了 TCPN 理论.针对变迁可调度原始定义的不足,首先给出了变迁的弱/强可调度的新定义及强可调度判定定理;然后对变迁的强可调度进行了拓展,提出了 TCPN 时间可调度的概念,并结合 Petri 网结构给出了 TCPN 时间可调度判定定理;最后对 TCPN 的相关特性进行了研究.TCPN 对时间 Petri 网的研究工作将起到启发作用,并对实时系统的描述和建模工作具有一定的参考价值.

关键词: 时间相关 Petri 网;TCPN;弱可调度;强可调度;可调度性分析

中图法分类号: TP301 文献标识码: A

Petri 网在 1962 年被 Carl Adam Petri 作为一种过程建模和分析工具提了出来^[1,2],随着 Petri 网理论的发展和完善,出现了各种各样的 Petri 网系统^[2-4].Petri 网直观的图形表示和坚实的数学基础使其特别适合描述异步

* Supported by the National Natural Science Foundation of China under Grant Nos.60303025, 60673017 (国家自然科学基金); the Jiangsu Provincial NSF Research Funds under Grant Nos.BK2004411, BK2005208 (江苏省基础研究计划)

Received 2005-12-09; Accepted 2006-04-03

并发系统^[1,2]。一般而言,在进行并发系统(特别是实时系统)的建模和分析时,时间往往是一个非常重要的因素^[4-7]。然而,在传统的 Petri 网理论中,时间的概念并没有显式地体现在 Petri 网原型中,这约束了 Petri 网在一些实时系统中的应用。因此,将时间概念引入到 Petri 网中,得到时间扩展的 Petri 网,可以显著地提高 Petri 网的建模和分析能力,尤其体现在各种实时系统的时间行为分析和性能评估等方面^[4-10]。

随着时间因素在 Petri 网理论和应用分析中越来越受到重视,不同的研究团队相继提出了各种与时间有关的 Petri 网方案,在 Petri 网的理论研究中出现了时间 Petri 网这一分支。普通 Petri 网只有局部时间的概念,而现阶段与时间相关的 Petri 网(下文简称时间 Petri 网)方面的研究,往往既对应局部时间概念,又假设系统存在一个全局时间坐标。在具体的时间 Petri 网分析过程中,往往通过托肯的流动将局部时间转化为全局时间,这是时间 Petri 网和普通 Petri 网的不同之处,与普通 Petri 网的以时钟读数处理局部时间的观点不甚一致^[2]。本文讨论的时间约束 Petri 网(timing constraint Petri nets,简称 TCPN)即是一种特殊的时间 Petri 网,特别适合于描述实时系统^[7]。

1 时间 Petri 网相关工作

在具体的时间 Petri 网研究中,往往通过在库所、变迁以及连接库所和变迁的有向弧上添加时间信息,从而形成各种时间 Petri 网模型^[4-11]。例如,在利用变迁模拟事件时,事件的发生需要时间,因此,通过在变迁上添加时间约束,对传统的 Petri 网进行扩展。几种常见的 Petri 网系统,如 timed Petri nets, stochastic Petri nets, time Petri nets 与 timing constraint Petri nets,往往是在基本网系统或者库所/变迁网系统的变迁上添加相应的时间约束衍生而成^[4,6,7]。在库所或者在有向弧上添加时间约束,与在相应的变迁上添加时间约束有相似的语义,它们之间往往可以相互转化^[7]。

Timed Petri nets 由 Ramchandani 提出^[6,8],它在库所/变迁网中的每一个变迁上添加一个时间延迟 T_{del} 。它遵循强触发规则,即一个变迁 t_j 对应着一个时间延迟 T_{del} ,如果在时间 T_0, t_j 的所有输入库所的 Token 均准备就绪,那么变迁 t_j 立即执行。在时间 $(T_0, T_0 + T_{del})$ 内,所有输入库所的 Token 都归 t_j 所使用,其余变迁不能使用。在时刻 $T_0 + T_{del}$ 到来时, t_j 输入库所的 Token 就被转移到相应的输出库所中去。Timed Petri nets 及其相关模型主要用于并发系统的性能分析^[6,8]。Stochastic Petri nets 是 timed Petri nets 的改进,由于在变迁发生之前,我们无法对每一个变迁给出一个准确的时间延迟,但通过以前的经验或者期望可以知道变迁的执行延迟符合某种概率分布,因此,可以利用变迁执行的平均时间代表变迁的延迟。与 timed Petri nets 一样, stochastic Petri nets 主要用于性能评估^[4,9]。清华大学学者林闯在这方面做了许多工作。Time Petri net (TPN) 由 Merlin 等人首先提出^[6]。TPN 通过将 timed Petri nets 中变迁的时间延迟替换为一个使能区间 (TC_{min}, TC_{max}) 而得到,其中, TC_{min} 表示相应变迁使能前所必须流逝的最小时间;而 TC_{max} 表示相应变迁触发前可以经历的最长时间。例如一个变迁 t_j ,如果在 T_0 时刻使能,那么,只有在时间区间 $(T_0 + TC_{min}, T_0 + TC_{max})$ 内才能触发。由于 TPN 遵循强触发规则,如果在区间 $(T_0 + TC_{min}, T_0 + TC_{max})$ 内不能触发,并且它的触发条件没有因其他变迁触发而改变,那么 t_j 将在时刻 $T_0 + TC_{max}$ 触发。注意:Timed Petri nets 和 stochastic Petri nets 中变迁的执行均需要一段时间,而 TPN 中变迁的执行瞬间即可完成,但它们都遵循强触发规则。TPN 是 timed Petri nets 的一般化,即 timed Petri nets 可以看成是 TPN 的特例^[5]。TPN 已被证明可比较方便地表示并发系统中的时间约束,因为在许多场合很难通过一个确定的时间延迟来表示相应变迁的执行时间。因此,TPN 的应用面比较广。TPN 的应用领域主要集中在各种实时系统的时间行为分析和性能评估等方面^[6,7]。

Timing constraint Petri nets (TCPN) 由 Tsai 提出^[7]。鉴于事件的发生条件有时也有相应的时间要求,从而 TCPN 中库所和变迁上均有时间约束信息。TCPN 库所上的时间约束借鉴了 TPN 的思路,而变迁上的时间约束则借鉴了 timed Petri nets 和 TPN。Timed Petri nets, stochastic Petri nets 和 TPN 均使用强触发规则,而 TCPN 采用弱触发规则,因此,与普通 Petri 网一样可以描述冲突结构。TCPN 中变迁的触发不仅受制于变迁的外延,而且与变迁及其输入库所上的时间约束信息密切相关。范玉顺等人所在的 CIMS 中心研究组将 TCPN 应用到 workflow 建模领域,根据 workflow 技术特点,提出变迁局部/全局可调度性的定义^[12,13]。袁崇义教授在 Petri 网原理方面所做的工作^[2]及上述国内外研究团队的工作均对本文有巨大的启发作用。针对 TCPN 应用的特点,我们将只对具有自由选择属性^[2,12]的 TCPN 进行研究,改进其不合理之处^[14],重新给出变迁弱/强可调度的定义,探讨验证和分析方法,

进而总结并分析 TCPN 的特点,并与普通 Petri 网和其他时间 Petri 网的相应特点进行比较.

本文第 2 节对 TCPN 中的时间约束信息进行重新解释,使其更加合理.在新的语义下重新定义变迁的弱可调度性和强可调度性,并给出 TCPN 变迁强可调度性分析方法,基于此,提出 TCPN 可调度的定义和判定定理.第 3 节是 TCPN 的应用实例.最后,对 TCPN 特点进行分析和研究.

2 TCPN 及其可调度性分析

2.1 TCPN 定义

根据相关文献,TCPN 可以从形式上定义如下^[7,12]:

定义 1. TCPN 是一个六元组 $\langle P, T, F, C, D, M \rangle$:

$P = \{p_1, p_2, \dots, p_m\}$ 为库所集合;

$T = \{t_1, t_2, \dots, t_n\}$ 为变迁集合;

F 表示连接库所和变迁的有向弧集合;

C 为关联库所和变迁的实数对 $[t_{\min}, t_{\max}]$ 的集合;

D 为变迁的执行延迟 $t_d(t)$ 的集合;

M 为用来描述系统状态的 m 维向量集合(标识),其中分量 $M(p)$ 表示库所 p 中的托肯数,一般用 M_0 表示初始标识.

TCPN 在实时系统的建模与分析等领域具有很强的表达能力.但是,如何利用 TCPN 中的时间约束条件对实时系统中的变迁可调度性进行分析和验证,现有的 TCPN 理论还需进一步完善.文献[14]曾明确指出:在 TCPN 中,其变迁的可调度性与时间约束的含义在某些条件下不甚一致,并以图 1 为例,结合变迁的弱可调度性分析,对这种不一致性情况进行了解释.

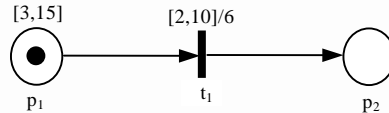


Fig.1 A concrete fragment of TCPN

图 1 一个具体的 TCPN 片段

文献[7]中,变迁 t_j 的弱可调度的定义如下:

$$(1) t_{EF}(t_j) = \max\{t_{\min}(p_j)\} + t_{\min}(t_j);$$

$$(2) t_{LF}(t_j) = \min\{t_{\max}(p_j), t_{\max}(t_j)\}.$$

若 $t_{LF}(t_j) - t_{EF}(t_j) \geq t_d(t_j)$, 则变迁 t_j 是弱可调度的.其中: p_j 为变迁 t_j 的输入库所; $t_{EF}(t_j)/t_{LF}(t_j)$ 为变迁 t_j 的最早/最迟可触发时间.

图 1 对应于一个具体的 TCPN 片段,在该片段中,变迁 t_1 有唯一的输入库所 p_1 .假设托肯到达库所 p_1 的时刻为 T_0 .按照上面的变迁弱可调度的定义,变迁 t_1 的最早和最迟可触发时间分别为:

$$(1) t_{EF}(t_1) = \max\{t_{\min}(p_1)\} + t_{\min}(t_1) = 3 + 2 = 5;$$

$$(2) t_{LF}(t_1) = \min\{t_{\max}(p_1), t_{\max}(t_1)\} = \min\{15, 10\} = 10.$$

由于 $t_{LF}(t_1) - t_{EF}(t_1) = 10 - 5 = 5 < 6$, 所以变迁 t_1 不是弱可调度的.而文献[14]认为,上述结果与时间约束的含义不甚一致.因为按照文献[7]中时间约束的含义,库所上的时间约束表示变迁 t_1 可以在区间 $[T_0 + 3, T_0 + 15]$ 这个时间段内发生,而变迁 t_1 上的时间约束 $[2, 10]$ 表示变迁可以在区间 $[T_0 + 2, T_0 + 10]$ 内发生.因此,

$$(1) t_{EF}(t_1) = \max\{T_0 + 3, T_0 + 2\} = T_0 + 3;$$

$$(2) t_{LF}(t_1) = \min\{T_0 + 15, T_0 + 10\} = T_0 + 10.$$

此时, $t_{LF}(t_1) - t_{EF}(t_1) = T_0 + 10 - (T_0 + 3) = 7 > 6$, 因此,按照时间约束的含义,变迁 t_1 成为可调度的.产生这种矛盾的原

因在于 TCPN 变迁可调度性定义的不合理性.但是,文献[14]没有对这种矛盾的结论进行针对性的深入研究,也没有给出相关的解决方案.针对这种情况,我们将结合 TCPN 的原始定义,对相关定义重新进行界定,使其能够在可调度分析方面更加合理.

2.2 TCPN中时间约束语义的形式化约定

结合 TCPN 的原始定义,图 2 对泛化的 TCPN 片段中的库所和变迁上的时间约束条件说明如下:时间约束的单位可以根据具体情况设定为分、小时等.其中: $[t_{\min}(p_1), t_{\max}(p_1)]$ 和 $[t_{\min}(p_2), t_{\max}(p_2)]$ 分别是库所 p_1 和 p_2 上的局部时间约束,表示库所支持后续变迁发生的使能区间.例如,库所 p_1 在 T_0 时刻获得一个托肯,则在区间 $[T_0+t_{\min}(p_1), T_0+t_{\max}(p_1)]$ 内, t_1 被使能. $[t_{\min}(t_1), t_{\max}(t_1)]/t_d(t_1)$ 是变迁 t_1 上的局部时间约束,其中:前者表示变迁 t_1 的可触发区间;后者表示变迁 t_1 的执行延迟时间.假设变迁 t_1 在 T_1 时刻使能,由于变迁的局部时间约束,使得变迁 t_1 只能在区间 $[T_1+t_{\min}(t_1), T_1+t_{\max}(t_1)]$ 内触发.如果 $t_{\max}(t_1)-t_{\min}(t_1) \geq t_d(t_1)$ 和 $t_{\max}(p_1)-t_{\min}(p_1)-t_{\min}(t_1) \geq t_d(t_1)$ 同时成立,变迁 t_1 能够顺利触发并结束.因此,在 TCPN 中变迁的触发受变迁和变迁的输入库所集上的时间共同约束.下文将进一步考虑 Token 到达库所的时间,从而引入变迁弱可调度和强可调度的概念.

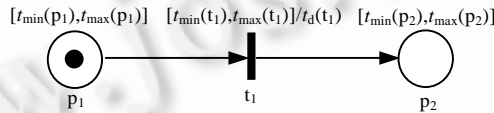


Fig.2 A generalized fragment of TCPN

图 2 一个泛化的 TCPN 片段

本文用 $I_p(t)/O_p(t)$ 表示变迁 t 的输入/输出库所集合; $I_i(p)/O_i(p)$ 表示库所 p 的输入/输出变迁集合; $\delta=(M_0t_1M_1\dots t_nM_n)$ 表示系统从状态 M_0 到达状态 M_n ; $\delta_k(M_n)$ 表示从 M_0 到 M_n 的第 k 条路径上的除第 1 个变迁之外的所有库所和变迁的序列, n 为这个序列中最后一个变迁的序号; $t_{EE}(t)/t_{LE}(t)$ 表示变迁 t 的最早/最迟使能时间; $t_{EF}(t)/t_{LF}(t)$ 表示变迁 t 的最早/最迟可触发时间; $t_{EB}(t)/t_{FE}(t)$ 表示变迁 t 的触发的开始/结束时间; $Token_a(p)/Token_l(p)$ 分别表示托肯到达和离开库所 p 的时间.这些概念的详细定义读者可以参考文献[7,12].

很明显,有如下关系式成立:

$$[t_{EB}(t), t_{FE}(t)] \subseteq [t_{EF}(t), t_{LF}(t)] \subseteq [t_{EE}(t), t_{LE}(t)] \quad (1)$$

$$Token_l(p) = \min\{t_{FE}(t_i)\}, t_i \in O_i(p) \quad (2)$$

$$Token_a(p) = \min\{t_{EB}(t_i)\}, t_i \in I_i(p) \quad (3)$$

$$t_{FE}(t) - t_{EB}(t) = t_d(t) \quad (4)$$

TCPN 的状态可达问题与变迁的可调度性问题密切相关,例如有如下状态转换序列: $\delta=(M_0t_1M_1\dots t_nM_n)$,判断状态 M_n 的可达问题转化为判断变迁序列 $t_1t_2\dots t_n$ 中的每一个变迁是否可调度的问题.在普通的 Petri 网中,变迁的触发规则只需考虑变迁的外延条件即可,而 TCPN 的变迁可调度性分析还必须考虑相应的时间约束是否能够得到满足.在 TCPN 中去掉所有时间约束所得到的网系统,称为该 TCPN 的基网.在基网下可达的状态,由于时间约束条件,在 TCPN 下并不一定可达.因此,我们必须从时间约束的角度来考虑变迁序列的可调度问题.

2.3 TCPN可调度性分析

TCPN 中的变迁可调度性分为弱可调度性和强可调度性^[7]两类.结合第 2.2 节中所给出的 TCPN 中时间约束语义的形式化约定,我们可以对 TCPN 的变迁弱可调度性和强可调度性重新定义如下:

定义 2. 如果一个变迁 t_j 的所有输入库所中都至少有一个托肯,若不需要考虑托肯到达库所的时间(或者假设它们到达相应库所的时间相同),则称变迁 t_j 是弱可调度的当且仅当条件 $t_{LF}(t_j)-t_{EF}(t_j) \geq t_d(t_j)$ 成立,其中

$$t_{EF}(t_j) = t_{EE}(t_j) + t_{\min}(t_j) = \max\{t_{\min}(p_j)\} + t_{\min}(t_j),$$

$$t_{LF}(t_j) = t_{LE}(t_j) = \min\{\min\{t_{\max}(p_j)\}, \max\{t_{\min}(p_j)\} + t_{\max}(t_j)\}.$$

式中: $p_j \in I_p(t_j)$, $t_{EE}(t_j)/t_{LE}(t_j)$ 表示变迁 t_j 的最早/最迟使能时间; $t_{EF}(t_j)/t_{LF}(t_j)$ 表示变迁 t_j 的最早/最迟可触发时间. 例如, 可以按照这里给出的变迁弱可调度的定义, 对如图 1 所示的 TCPN 片段进行弱可调度的判断.

$$(1) t_{EF}(t_1) = t_{EE}(t_1) + t_{\min}(t_1) = \max\{3\} + 2 = 5;$$

$$(2) t_{LF}(t_1) = t_{LE}(t_1) = \min\{\min\{t_{\max}(p_1), t_{\min}(p_1), \max\{t_{\min}(p_1)\} + t_{\max}(t_1)\}\} = \min\{15, 3+10\} = 13.$$

$t_{LF}(t_1) - t_{EF}(t_1) = 13 - 5 = 8 > 6$, 变迁 t_1 弱可调度. 因此, 在我们的弱可调度性定义条件下, 第 2.1 节提到的 TCPN 时间约束语义和弱可调度性分析产生的矛盾得以解决. 这种弱可调度性由于不必考虑托肯的到达时间, 变迁的触发与全局状态无关, 因而具有局部确定性.

定义 3. TCPN 的状态转换序列 $\delta = (M_0 t_1 M_1 \dots t_i M_i \dots t_n M_n)$ 中一个弱可调度的变迁 t_j , 如果考虑它的每个输入库中所托肯的到达时间, 则它在当前状态 M_{j-1} 下为强可调度的当且仅当 $t_{LF}(t_j) - t_{EF}(t_j) \geq t_d(t_j)$ 成立, 其中,

$$t_{EF}(t_j) = t_{EE}(t_j) + t_{\min}(t_j) = \max\{\min\{Token_a(p_j)\} + t_{\min}(p_j)\} + t_{\min}(t_j),$$

$$t_{LF}(t_j) = t_{LE}(t_j) = \min\{\min\{\max\{Token_a(p_j)\} + t_{\max}(p_j)\}, \max\{\max\{Token_a(p_j)\} + t_{\min}(p_j)\} + t_{\max}(t_j)\}.$$

式中: $p_j \in I_p(t_j)$, $t_{EE}(t_j)/t_{LE}(t_j)$ 表示变迁 t_j 的最早/最迟使能时间; $t_{EF}(t_j)/t_{LF}(t_j)$ 表示变迁 t_j 的最早/最迟可触发时间.

这种强可调度性, 变迁的成功触发不仅与系统的当前状态 M_{j-1} 有关, 并且因为需要考虑托肯到达库所的时间, 还与系统的全局状态有关, 不具有局部确定性, 状态 M_{j-1} 可达说明之前的变迁序列均为强可调度的. 我们研究的 TCPN 都具有自由选择特性, 如果事先对 TCPN 的冲突结构进行分解, 可以得到一组没有冲突结构的 T_网^[2,11]. 对每一个 T_网, 利用下面给出的定理 1, 可以从初始变迁触发结束后系统的状态 M_1 来判断变迁 t_j 是否强可调度.

定义 4. 如果一个变迁在 Petri 网的初始状态 M_0 下可以发生, 称这样的变迁为初始变迁.

定义 5. 如果一个变迁 t_j 的输入库所个数大于 1, 即 $|I_p(t_j)| > 1$, 则称变迁 t_j 为同步变迁.

我们假设初始变迁总是可调度的. TCPN 就是利用初始变迁的触发结束时间作为坐标原点, 将库所和变迁上的局部时间转化为系统全局时间的.

定理 1. TCPN 的初始状态 M_0 下某一弱可调度的变迁是强可调度的当且仅当条件 $t_{LF}(t_j) - t_{EF}(t_j) \geq t_d(t_j)$ 成立, 其中,

$$t_{EF}(t_j) = t_{EE}(t_j) + t_{\min}(t_j) = \max\{Token_a(p_j) + t_{\min}(p_j)\} + t_{\min}(t_j) \\ = t_{FE}(t_1) + \max_k \{\sum t_{\min}(t_{nk}) + \sum t_d(t_{nk}) + \sum t_{\min}(p_{mk})\} - t_d(t_j),$$

$$t_{LF}(t_j) = t_{LE}(t_j) = \min\{\min\{\max\{Token_a(p_j)\} + t_{\max}(p_j)\}, \max\{\max\{Token_a(p_j)\} + t_{\min}(p_j)\} + t_{\max}(t_j)\} \\ = t_{LF}(t_s) + \{\sum_{\min} \{t_{\max}(p_{mk}), t_{\min}(p_{mk}) + t_{\max}(t_{nk})\}\}.$$

式中 $p_j \in I_p(t_j)$, $t_{nk}, p_{mk} \in \delta_k(M_j)$, 变迁 t_{nk} 是库所 p_{mk} 的直接后继. 其中, $\delta_k(M_j)$ 表示从 M_0 到 M_j 的第 k 条路径上的除第一个变迁之外的所有库所和变迁组成的序列. 变迁 t_s 为变迁 t_1 到 t_j 的路径上离 t_j 最近的同步变迁, 特别地, 若在此之前没有同步变迁, t_1 即为同步变迁 t_s ; 若 t_j 即为同步变迁 t_s , 则 $t_{LF}(t_j) = t_{LF}(t_s)$, $t_{LF}(t_s)$ 根据定义 3 求得.

证明: 由定义 3 可知: 确定一个变迁是否强可调度, 除了考虑变迁上的时间约束和变迁的输入库所上的时间约束以外, 必须考虑变迁输入库所中 Token 的到达时间. 要求出 $t_{EF}(t_j)/t_{LF}(t_j)$, 必须求出 Token 到达时间 $Token_a(p_j)$ 的上下界. 如果从 $\delta_k(M_j)$ 的角度考虑, 式(3)可以简化为 $Token_a(p) = t_{FE}(t_j)$, 其中 $t_j \in I_p(p)$. 于是, 求 $Token_a(p_j)$ 的上下界问题转化为求 $t_{FE}(t_{j-1})$ 的上下界问题, 而 $t_{FE}(t_{j-1})$ 的上界和下界分别对应于 $t_{EF}(t_{j-1}) + t_d(t_j)$ 和 $t_{LF}(t_j)$. 以上分析说明: 求 $t_{EF}(t_j)/t_{LF}(t_j)$ 的问题包含求 $t_{EF}(t_{j-1}) + t_d(t_j)/t_{LF}(t_j)$ 的问题, $t_d(t_j)$ 是常量, 因此, 该问题的求解是一个递归过程. 定理 1 将变迁强可调度定义由递归转化为非递归, 具体过程如下:

由于只考虑一条路径, 定义 3 中变迁最迟可触发时间可以简化, 由式(1)可知:

$$\min\{Token_a(p_j)\} + t_{\min}(p_j) + t_{\min}(t_j) \leq t_{FB}(t_j) \quad (5)$$

$$t_{EE}(t_j) \leq \max\{Token_a(p_j)\} + \min\{t_{\max}(p_j), t_{\min}(p_j) + t_{\max}(t_j)\} \quad (6)$$

而由式(4)可知: $t_{FE}(t_j) - t_d(t_j) = t_{FB}(t_j)$, 由式(4)和式(5)可得如下不等式:

$$\min\{Token_a(p_j)\} + t_{\min}(p_j) + t_{\min}(t_j) + t_d(t_j) \leq t_{FE}(t_j) \quad (7)$$

由式(3): $Token_a(p_j) = t_{FE}(t_{j-1})$, 将式(3)代入式(7)和式(6)有,

$$\min\{t_{FE}(t_{j-1})\}+t_{\min}(p_j)+t_{\min}(t_j)+t_d(t_j)\leq t_{FE}(t_j) \quad (8)$$

$$t_{EE}(t_j)\leq t_{FE}(t_{j-1})+\min\{t_{\max}(p_j),t_{\min}(p_j)+t_{\max}(t_j)\} \quad (9)$$

所以,我们可以通过 $t_{FE}(t_{j-1})$ 的上下界来确定 $Token_a(p_j)$ 的上下界.而 $t_{FE}(t_{j-1})$ 的上下界有如下不等式:

$$\min\{t_{FE}(t_{j-2})\}+t_{\min}(p_{j-1})+t_{\min}(t_{j-1})+t_d(t_{j-1})\leq t_{FE}(t_{j-1}) \quad (10)$$

$$t_{FE}(t_{j-1})\leq \max\{t_{FE}(t_{j-2})\}+\min\{t_{\max}(p_{j-1}),t_{\min}(p_{j-1})+t_{\max}(t_{j-1})\} \quad (11)$$

于是,联立式(8)~式(11)可得:

$$\min\{t_{FE}(t_{j-2})\}+t_{\min}(p_{j-1})+t_{\min}(t_{j-1})+t_d(t_{j-1})+t_{\min}(p_j)+t_{\min}(t_j)+t_d(t_j)\leq t_{FE}(t_j) \quad (12)$$

$$t_{FE}(t_j)\leq \max\{t_{FE}(t_{j-2})\}+\min\{t_{\max}(p_j),t_{\min}(p_j)+t_{\max}(t_j)\}+\min\{t_{\max}(p_{j-1}),t_{\min}(p_{j-1})+t_{\max}(t_{j-1})\} \quad (13)$$

依此类推,直到化归到该路径上离 t_j 最近的同步变迁 t_s 的最早/最迟可触发时间为止,可得:

$$t_{EF}(t_j)=t_{EF}(t_s)+\{\sum t_{\min}(t_{nk})+\sum t_d(t_{nk})+\sum t_{\min}(p_{mk})\}-t_d(t_j) \quad (14)$$

$$t_{LF}(t_j)=t_{LF}(t_s)+\{\sum \min\{t_{\max}(p_{mk}),t_{\min}(p_{mk})+t_{\max}(t_{nk})\}\} \quad (15)$$

无论变迁 t_j 是否是同步变迁,变迁 t_s 的最早可触发时间可以继续前递归,直至化归到初始变迁的触发结束时间为止,当遍历完从 M_0 到 M_j 的所有路径之后可得:

$$t_{EF}(t_j)=t_{EF}(t_1)+\max_k\{\sum t_{\min}(t_{nk})+\sum t_d(t_{nk})+\sum t_{\min}(p_{mk})\}-t_d(t_j) \quad (16)$$

由式(15)、式(16)可知,定理成立.定理 1 实际上给出了 TCPN 变迁强可调度性的非递归验证方法.

定义 6. 一个 TCPN 是时间可调度的,当且仅当网中全部变迁是强可调度的.即如果网中有一个变迁不是强可调度的,那么该 TCPN 不可调度.

定理 2. 如果一个 TCPN 中除初始变迁外的所有同步变迁强可调度,其余变迁弱可调度,则该 TCPN 是时间可调度的.

证明:根据定义 3 和 6,必要性成立.

充分性证明如下:已知所有同步变迁均为强可调度的,我们只需考虑非同步变迁是否强可调度即可.非同步变迁的输入库所只有一个,因此对于任意一个非同步变迁 t_j ,设 p_j 是其唯一的输入库所.而已知非同步变迁都是弱可调度的,根据定义 3 有,

$$\begin{aligned} t_{LF}(t_j)-t_{EF}(t_j) &= \min\{\min\{\max\{Token_a(p_j)\}+t_{\max}(p_j)\},\max\{\max\{Token_a(p_j)\}+t_{\min}(p_j)\}+t_{\max}(t_j)\}- \\ &\quad \max\{\min\{Token_a(p_j)\}+t_{\min}(p_j)\}-t_{\min}(t_j)\} \\ &= \{\max\{Token_a(p_j)\}+\min\{t_{\max}(p_j),t_{\min}(p_j)+t_{\max}(t_j)\}\}-\{\min\{Token_a(p_j)\}+t_{\min}(p_j)\}-t_{\min}(t_j)\} \\ &= \max\{Token_a(p_j)\}-\min\{Token_a(p_j)\}+\min\{t_{\max}(p_j),t_{\min}(p_j)+t_{\max}(t_j)\}-t_{\min}(p_j)-t_{\min}(t_j). \end{aligned}$$

已知非同步变迁都是弱可调度的,即有 $\min\{t_{\max}(p_j),t_{\min}(p_j)+t_{\max}(t_j)\}-t_{\min}(p_j)-t_{\min}(t_j)\geq t_d(t_j)$,并且 $\max\{Token_a(p_j)\}-\min\{Token_a(p_j)\}\geq 0$,所以 $t_{LF}(t_j)-t_{EF}(t_j)\geq t_d(t_j)$.表明弱可调度的非同步变迁是强可调度的.于是,网中的所有变迁都是强可调度的.因此根据定义 6,该 TCPN 是可调度的.

实际上,定理 2 给出了判定 TCPN 强可调度的分析和验证方法.

最后需要指出的是:并不是满足弱/强可调度的变迁一定都能触发并顺利完成,但一个弱/强可调度的变迁在 $[t_{EF}(t_j),t_{LF}(t_j)-t_d(t_j)]$ 内触发一定能顺利完成,这个区间称为变迁的可调度决策范围^[12,13].

3 实例分析

Petri 网是一种图形化描述过程的强有力工具,它可以直观地描述一个过程.荷兰的 Aaslt 将 Petri 网的相关概念映射到工作流中,得到工作流网(WF-net)^[15].当我们需要对工作流的预期性能进行判断时,就可以利用时间扩展的 Petri 网进行建模,而 TCPN 可以建模冲突结构,因此特别适合工作流等实时系统的建模.

下面利用 TCPN 对工作流过程进行建模,并对工作流的性能进行分析.图 3 所示的 TCPN 对保险索赔过程进行了建模,它是自由选择的,变迁对应的操作语义附在表 1 中.库所和变迁上的时间约束信息通过保险索赔业务过程的要求或者保险公司经验给出,单位为分,如果相应库所或变迁无特殊的时间约束,则默认为 $[0,\infty]$.初始变迁为 t_1 ,变迁 t_2 和 t_4 处在一个冲突结构中,冲突的解决可以通过案例的属性由系统环境作出决策.为了验证

TCPN的时间约束信息是否妥当,可以先将相应的 TCPN 进行分解,然后就可以按照定理 1 对每一个变迁的强可调度性进行分析.

假设开始的时候库所 p_1 中有一个 Token, p_1 中的 Token 代表索赔案例.初始变迁 t_1 的触发没有时间约束,我们以变迁 t_1 的触发结束时间 $t_{FE}(t_1)$ 为坐标原点对后续变迁的可调度性进行分析. $t_{FE}(t_1)$ 可以是参照案例到达时间而得到的相对时间,也可以是绝对时间.而变迁 t_2 和 t_4 处在一个冲突结构中,即根据案例属性来确定是给予赔偿还是拒绝赔偿.根据定义 2 不难验证网中变迁均是弱可调度的.为了利用定理 1 来判定网的强可调度性,我们对如图 3 所示的 TCPN 冲突结构进行分解,可以得到两个 T_网,如图 4 所示.

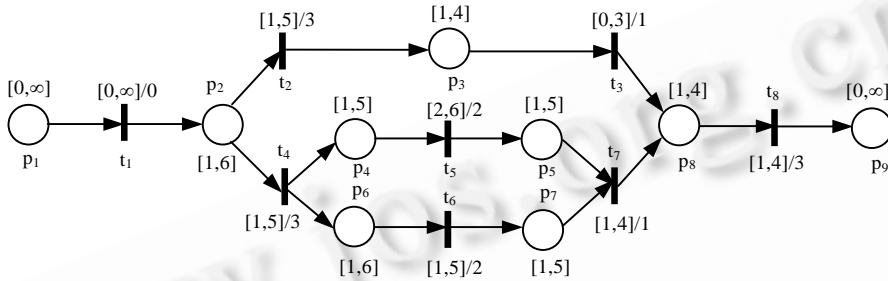


Fig.3 Using TCPN to model insurance claim
图 3 TCPN 对保险索赔过程的建模

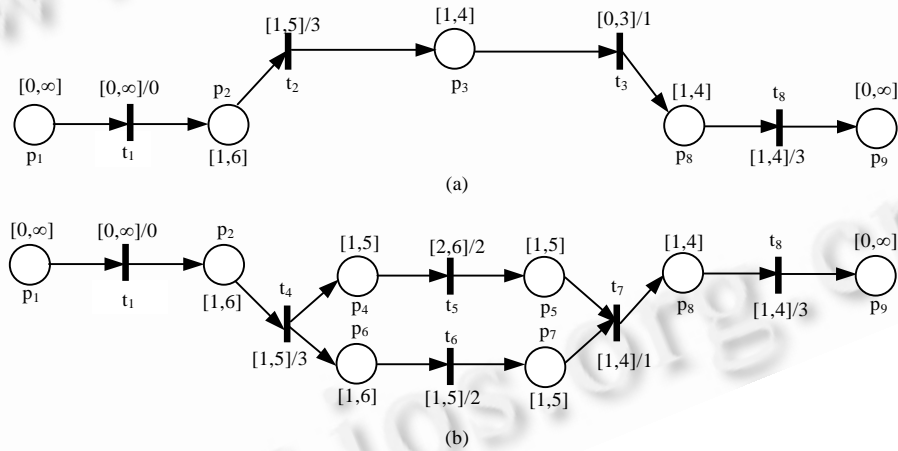


Fig.4 Two T_nets are gained after decomposing Fig.3
图 4 图 3 冲突结构分解后得到的两个 T_网

在图 4(a)中,考虑变迁 t_2 的可调度性的时候应该考虑 Token 到达库所 p_2 的时间.根据定理 1,验证变迁 t_2 的强可调度性的过程如下:

$$t_{EF}(t_2) = t_{EE}(t_2) + t_{\min}(t_2) = t_{FE}(t_1) + \max\{t_{\min}(p_2) + t_{\min}(t_2) + t_d(t_2)\} - t_d(t_2) = (t_{FE}(t_1) + 1 + 1 + 2) - 2 = t_{FE}(t_1) + 2,$$

$$t_{LF}(t_2) = t_{LE}(t_2) = t_{FE}(t_1) + \min\{t_{\max}(p_2), t_{\min}(p_2) + t_{\max}(t_2)\} = t_{FE}(t_1) + \min\{6, 1 + 5\} = t_{FE}(t_1) + 6,$$

$$t_{LF}(t_2) - t_{EF}(t_2) = (t_{FE}(t_1) + 6) - (t_{FE}(t_1) + 2) = 4 \geq 3 = t_d(t_2).$$

因此,变迁 t_2 是强可调度的.另一种判定变迁 t_2 强可调度性的方法只需分析变迁 t_2 的弱可调度性即可.根据定义 2,有

$$t_{EF}(t_2) = t_{EE}(t_2) + t_{\min}(t_2) = \max\{t_{\min}(p_2)\} + t_{\min}(t_2) = t_{\min}(p_2) + t_{\min}(t_2) = 1 + 1 = 2,$$

$$t_{LF}(t_2) = t_{LE}(t_2) = \min\{\min\{t_{\max}(p_2), \max\{t_{\min}(p_2)\} + t_{\max}(t_2)\}\} = \min\{t_{\max}(p_2), t_{\min}(p_2) + t_{\max}(t_2)\} = \min\{6, 1 + 5\} = 6,$$

$$t_{LF}(t_2) - t_{EF}(t_2) = 6 - 2 = 4 \geq 3 = t_d(t_2).$$

所以,变迁 t_2 是弱可调度的,并且注意到变迁 t_2 不是同步变迁,根据定理 2 可知,变迁 t_2 也是强可调度的.利用这两种方法,可以验证图 4(a)中变迁 t_3 和变迁 t_8 也是强可调度的.按照同样的思路,可以验证图 4(b)中变迁 t_4, t_5, t_6 和 t_8 的强可调度性.下面具体分析同步变迁 t_7 的强可调度性,利用已求得的 t_5, t_6 的可调度区间,根据定义 3 有

$$\begin{aligned} t_{EF}(t_7) &= t_{EE}(t_7) + t_{\min}(t_7) = \max\{\min\{Token_d(p_5)\} + t_{\min}(p_5), \min\{Token_d(p_7)\} + t_{\min}(p_7)\} + t_{\min}(t_7) \\ &= \max\{t_{FE}(t_1) + 8 + 2 + 1, t_{FE}(t_1) + 7 + 2 + 1\} + 1 = t_{FE}(t_1) + 12, \\ t_{LF}(t_7) &= t_{LE}(t_7) = \min\{\min\{t_{FE}(t_1) + 11 + 5, t_{FE}(t_1) + 12 + 5\}, \max\{t_{FE}(t_1) + 11 + 1, t_{FE}(t_1) + 12 + 1\} + 4\} = t_{FE}(t_1) + 16, \\ t_{LF}(t_7) - t_{EF}(t_7) &= (t_{FE}(t_1) + 16) - (t_{FE}(t_1) + 12) = 4 \geq 1 = t_d(t_7). \end{aligned}$$

因此,同步变迁 t_7 也是强可调度的.由于变迁 t_7 是同步变迁,因此,通过弱可调度性验证的方法不再成立.

利用定理 1 对上述图 3 中变迁的强可调度性进行验证,并求出相应的可调度区间,将计算结果汇总得到表 1.表 1 说明:图 3 中的 8 个变迁均强可调度,从而,这个 TCPN 是时间可调度的.这说明,在事先期望的时间约束条件下,只要每个变迁所对应的活动在相应的可调度决策范围内开始执行,案例索赔处理就能正常运行并顺利结束.注意到,变迁 t_8 在图 4(a)和图 4(b)中均是强可调度的,但相应的计算结果并不相同.图 4(a)和图 4(b)分别对应拒绝赔偿和给予赔偿两种情况下案例的处理流程,处理流程不同,最终计算结果也就不同.这就是为什么要对冲突结构进行分解的原因.到达库所 p_9 的 Token 表示相应的索赔案例处理结束.如果开始时有多个案例等待在库所 p_1 处,这些案例可以按照同样的流程进行处理,如果不断有新的案例出现,则该系统很像一个循环系统.最后需要指明的是:图 4(a)与图 4(b)中变迁 t_8 的最早/最迟可触发时间,分别对应了拒绝索赔和给予索赔两种情况下整个案例处理时间的上下界.如果再知道案例符合索赔的统计概率情况,就可以对这个索赔系统案例处理周期等性能指标进行分析.

Table 1 Schedulability analysis of the transitions

表 1 各个变迁可调度性分析

Transition t_i	Operation	$t_{LF}(t_i)$	$t_{EF}(t_i)$	$t_{LF}(t_i) - t_{EF}(t_i)$	$t_d(t_i)$	Schedulability	Decision span
t_1	Register	0	∞	∞	0	Strong	$(0, \infty)$
t_2	Send rejection	$t_{FE}(t_1) + 6$	$t_{FE}(t_1) + 2$	4	3	Strong	$[t_{FE}(t_1) + 2, t_{FE}(t_1) + 3]$
t_3	Record rejection	$t_{FE}(t_1) + 10$	$t_{FE}(t_1) + 6$	4	1	Strong	$[t_{FE}(t_1) + 6, t_{FE}(t_1) + 9]$
t_4	Count compensation	$t_{FE}(t_1) + 6$	$t_{FE}(t_1) + 2$	4	3	Strong	$[t_{FE}(t_1) + 2, t_{FE}(t_1) + 3]$
t_5	Send compensation	$t_{FE}(t_1) + 11$	$t_{FE}(t_1) + 8$	3	2	Strong	$[t_{FE}(t_1) + 8, t_{FE}(t_1) + 9]$
t_6	Send letter	$t_{FE}(t_1) + 12$	$t_{FE}(t_1) + 7$	5	2	Strong	$[t_{FE}(t_1) + 7, t_{FE}(t_1) + 10]$
t_7	Record compensation	$t_{FE}(t_1) + 16$	$t_{FE}(t_1) + 12$	4	1	Strong	$[t_{FE}(t_1) + 12, t_{FE}(t_1) + 15]$
$t_8(a)$	Archive rejection	$t_{FE}(t_1) + 14$	$t_{FE}(t_1) + 9$	5	3	Strong	$[t_{FE}(t_1) + 9, t_{FE}(t_1) + 11]$
$t_8(b)$	Archive compensation	$t_{FE}(t_1) + 20$	$t_{FE}(t_1) + 15$	5	3	Strong	$[t_{FE}(t_1) + 15, t_{FE}(t_1) + 17]$

利用图 4(a)中求出的变迁 t_8 的最早/最迟可触发时间,可以求出被拒绝赔付的案例从提出索赔申请到处理完毕这个过程所需要的时间在 $[t_{EF}(t_8) + t_d(t_8), t_{LF}(t_8)]$ 之内.同理,图 4(b)的变迁 t_8 的完成所需时间亦在 $[t_{EF}(t_8) + t_d(t_8), t_{LF}(t_8)]$ 之内.如果用完成时间的中值代表相应案例的完成时间,并且知道一个案例给予索赔的统计概率是 p ,则可以估算出这个索赔系统案例处理周期 T .

$$\begin{aligned} T &= 0.5 \times \{ [t_{FE}(t_1) + 15 + 3 + t_{FE}(t_1) + 20] \times p + [t_{FE}(t_1) + 9 + 3 + t_{FE}(t_1) + 14] \times (1 - p) \} \\ &= t_{FE}(t_1) + 13 + 6p. \end{aligned}$$

4 TCPN 特点分析

下面,我们通过对比 TCPN 与其他时间 Petri 网(主要以 TPN 为例进行说明)的异同,分析 TCPN 的一些特点.主要从局部/全局时间、局部/全局确定性、强/弱触发规则等方面进行分析.

TCPN 与其他时间 Petri 网一样,除了具有局部时间以外,还假定系统有一个全局时间^[2],局部时间通过 Token 的流动可以转化为全局时间.在考察 TCPN 的强可调度性时,系统的全局时间通常以初始变迁的触发结束时间作为坐标原点.当然,如果不考虑 Token 到达库所的时间,例如在考虑变迁的弱可调度性时, TCPN 和普通 Petri 网一样,只考虑局部时间.

普通 Petri 网和 TPN 等时间 Petri 网一样具有局部确定性,即一个变迁能否发生仅仅取决于变迁的输入集和

输出集,即变迁的外延,而与全局状态无关.若考虑 TCPN 的弱可调度性,变迁的调度不用考虑 Token 的到达时间,因此也是局部确定的.但是,在考虑 TCPN 变迁强可调度性时,特别是考虑同步变迁的强可调度性时,必须考虑库所中 Token 的到达时间,而这个时间是全局的.因此,仅仅通过变迁及其外延上的时间信息无法确定变迁是否强可调度,从而丧失了局部确定性.但是通过定理 2 可知:在考虑非同步变迁的强可调度性的时候,变迁的触发仍然具有局部确定性.

不强迫一个被使能的变迁触发的变迁触发方式称为弱触发规则,即一个被使能的变迁可以触发也可以不触发;强迫一个被使能的变迁触发的变迁触发方式称为强触发规则^[7].TCPN 可以遵循强和弱两种触发规则,而 TPN 等时间 Petri 网只遵循强触发规则.TPN 的强触发规则不仅与 Petri 网的弱触发规则不一致,而且不能对冲突结构进行建模^[7].结合图 5 具体分析如下:

图 5(a)是 Petri 网的一个冲突结构片段,而 Petri 网遵循弱触发规则,它不强迫变迁 t_1 或者 t_2 的发生, t_1 或者 t_2 的发生需要系统环境的决策;而图 5(b)是 TPN 的一个冲突结构片段,相应变迁的静态时间信息如图 5(b)所示.假设在 T_0 时刻一个 Token 到达库所 p_0 ,则变迁 t_1 与 t_2 分别在区间 (T_0+1, T_0+3) 和 (T_0+3, T_0+5) 内触发.如果变迁 t_1 在区间 (T_0+1, T_0+3) 内不能发生,而 TPN 遵循强触发规则^[6,7],所以,变迁 t_1 必须在时刻 T_0+3 触发,导致变迁 t_2 永远不能触发.这与冲突结构的定义相矛盾,并且变迁 t_2 是死变迁;而图 5(c)所示的 TCPN 可以对分支结构进行建模,按照前面介绍的内容易知变迁 t_1 与 t_2 都强可调度.TCPN 可以遵循弱触发规则,并不强迫变迁 t_1 发生,因此,变迁 t_2 有机会获得触发权.变迁 t_1 与 t_2 的触发权由系统环境的决策和相应的时间约束确定.

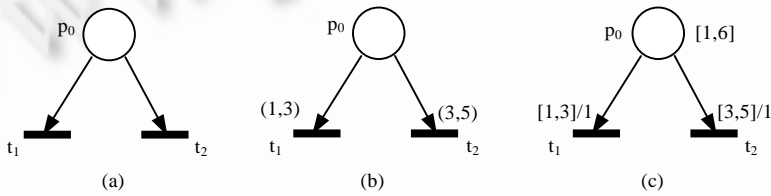


Fig.5 Conflict structure of Petri nets, TPN and TCPN

图 5 Petri 网,TPN,TCPN 描述冲突结构片段

表 2 对上面的分析进行了总结^[1,2,4-7].

Table 2 Comparison between Petri nets and time-related Petri nets

表 2 Petri 网和时间 Petri 网特点比较

	Petri nets	Time-Related Petri nets			
		Timed	Stochastic	Time	Timing constraint Petri nets
Transition timing constraint	Null	Constant	Mean	Interval	Time pair and duration
Place timing constraint	Null	Null			Time pair
Time modes	Local	Local and global			Local and global
Local determinability	Yes	Yes			No
Firing modes	Weak	Strong			Weak or strong
Conflict structure	Support	Nonsupport			Support

最后,分析 TCPN 的有界性、活性和可逆性.

定理 3. 一个可调度的 TCPN,并且所有变迁均在可调度决策范围内触发,如果去掉所有时间约束得到的普通 Petri 网是有界的、活的,那么该 TCPN 也是有界的、活的.

证明:一个可调度的 TCPN(简记为 N),去掉时间约束后所得到的普通 Petri 网记为 N_0 .如果不考虑时间因素, N 和 N_0 是同构的.假设 N_0 下有一个状态转换序列:

$$\delta=(M_0t_1M_1\dots t_iM_i\dots t_nM_n),$$

在 N 下必存在一个对应的状态转换序列

$$\delta'=(M'_0t'_1M'_1\dots t'_iM'_i\dots t'_nM'_n),$$

使得

$$M'_i=M_i, t'_i=t_i(i=1,2,\dots,n).$$

由于 TCPN 是可调度的,因此, $t'_i (i=1,2,\dots,n)$ 均强可调度,只要控制 t'_i 在相应的可调度决策范围内触发,就可以保证 t'_i 能够顺利完成,从而 $M'_i (i=1,2,\dots,n)$ 在 N 下也是可达的.这说明,对于 N_0 中的任意可达状态在 N 下也是可达的, N_0 中的任意一个可触发的变迁在 N 下都能触发并顺利完成,不会出现 Token 聚集在库所中的现象.因此, N_0 的活性和有界性都被继承下来.若 N_0 可逆,由于时间不可逆转,因此,可逆性在时间层面上不能得到继承.

5 总结与展望

把时间的概念引入到 Petri 网中就得到时间 Petri 网.时间 Petri 网的应用性比较强,针对不同应用特点,不同研究团队相继提出了各种各样的时间 Petri 网.本文的 TCPN 具有丰富的时间语义,其强大的建模能力和一些优良的特点,使得它可以被广泛地应用.针对 TCPN 原始定义不太完善的地方,本文对其变迁的强/弱可调度性重新进行了定义,并给出强可调度判定定理.随后,对变迁的弱/强可调度性进行拓展,提出了 TCPN 时间可调度的概念,并结合 Petri 网结构给出了 TCPN 时间可调度的判定定理.实例分析部分给出的 TCPN 在保险索赔申请中的应用实例是对上述内容的说明和应用.最后,本文对 TCPN 的特性进行研究.定理 3 给出的结论使得我们在分析 TCPN 特点的时候,可以利用普通 Petri 网的分析手段对相应的时间可调度的 TCPN 进行分析.

需要指出的是:TCPN 较为复杂,在利用 TCPN 对复杂结构或流程建模时,会使模型变得繁琐且不利于分析.下一步的工作将探讨 TCPN 的简化规则,以克服这一不足.

致谢 在此,我们向对本文提出宝贵意见的评审专家表示感谢.

References:

- [1] Murata T. Petri nets: Properties, analysis and applications. Proc. of the IEEE, 1989,77(4):541-580.
- [2] Yuan CY. Principle and Application of Petri Nets. Beijing: Publishing House of Electronics Industry, 2005 (in Chinese).
- [3] Jensen K. Coloured Petri Nets. 2nd ed. Hong Kong: Springer-Verlag, 1996.
- [4] Lin C. Stochastic Petri Nets and Performance Evaluation for System. 2nd ed. Beijing: Tsinghua University Press, 2005 (in Chinese).
- [5] Ghezzi C, Mandrioli D, Morasca S, Pezze M. A general way to put time in Petri nets. ACM SIGSOFT Software Engineering Notes, 1989,14(3):60-66.
- [6] Berthomieu B, Diaz M. Modeling and verification of time dependent systems using time Petri nets. IEEE Trans. on Software Engineering, 1991,17(3):259-272.
- [7] Tsai JJP, Yang SJ, Chang WH. Timing constraint Petri nets and their applications to schedulability analysis of real-time system specifications. IEEE Trans. on Software Engineering, 1995,21(1):32-49.
- [8] Zuberek WM. Performance evaluation of concurrent systems using timed Petri nets. In: Walker TM, Dominick WD, eds. Proc. of the 1985 ACM 13th Annual Conf. on Computer Science. New York: ACM Press, 1985. 326-329.
- [9] Ciardo G, German R, Lindemann C. A characterization of the stochastic process underlying a stochastic Petri net. IEEE Trans. on Software Engineering, 1994,20(7):506-515.
- [10] Juan YET, Tsai JJP, Murata T, Zhou Y. Reduction methods for real-time systems using delay time Petri nets. IEEE Trans. on Software Engineering, 2001,5(27):422-448.
- [11] Pedrycz W, Camargo H. Fuzzy timed Petri nets. Fuzzy Sets and Systems, 2003,140(2):301-330.
- [12] Li JQ, Fan YS, Zhou MC. Timing constraint workflow nets for workflow analysis. IEEE Trans. on Systems, Man, and Cybernetics, 2003,33(2):179-193.
- [13] Li HF, Fan YS. Workflow model analysis based on time constraint Petri nets. Journal of Software, 2004,15(1):17-26 (in Chinese with English abstract). <http://www.jos.org.cn/1000-9825/15/17.htm>
- [14] Xu DX, He XD, Deng Y. Compositional schedulability analysis of real-time systems using time Petri nets. IEEE Trans. on Software Engineering, 2002,29(10):984-996.
- [15] van der Aalst WMP. The application of Petri nets to workflow management. The Journal of Circuits, Systems and Computers, 1998, 8(1):21-66.

附中文参考文献:

[2] 袁崇义.Petri 网原理与应用.北京:电子工业出版社,2005.
 [4] 林闯.随机 Petri 网和系统性能评价.北京:清华大学出版社,2005.
 [13] 李慧芳,范玉顺.基于时间 Petri 网的工作流模型分析.软件学报,2004,15(1):17-26. <http://www.jos.org.cn/1000-9825/15/17.htm>



宋巍(1981 -),男,山东日照人,硕士生,主要研究领域为 Petri 网应用, workflow 技术, 协同计算.



刘茜萍(1981 -),女,博士生,主要研究领域为协同计算,认知科学.



竇万春(1971 -),男,博士,副教授,CCF 高级会员,主要研究领域为 workflow 技术,协同计算,知识管理.

www.jos.org.cn

第 4 届全国 Web 信息系统及其应用学术会议 (WISA 2007)

征文通知

全国 Web 信息系统及其应用会议 (WISA) 是中国计算机学会暨电子政务与办公自动化专委会主办的系列会议。第 4 届全国 Web 信息系统及其应用学术会议(WISA2007)将于 2007 年 9 月 14~16 日在中国人民大学召开。

一、征文范围 (包括但不限于)

- | | | | |
|------------------|------------------------|---------------------|---------------|
| Web 信息挖掘与检索 | Web2.0 与社会信息 | Web 与数据库技术 | XML 与半结构化数据管理 |
| Web 信息集成 | 语义 Web 与智能 Web | Web 应用框架和体系结构 | 组件与中间件技术 |
| Web 服务、SOA 与网格计算 | 工作流管理 | Web 站点逆向工程与维护技术 | 多媒体数据管理 |
| 决策支持与分析技术 | Deep Web 技术 | Web 测试与 Web 应用的质量保证 | Web 信息系统工程方法 |
| Web 与信息系统安全性 | Web 信息系统实际应用经验 | | |
| Web 系统度量与分析技术 | 基于 Web 的电子政务与电子商务框架及应用 | | |

二、来稿要求和联系信息

详见大会网站: <http://www.ruc.edu.cn/wisa2007/>, <http://www.neu.edu.cn/wisa2007/>

三、重要日期

1. 征文截止日期: 2007 年 4 月 1 日
2. 录用通知发出日期: 2007 年 4 月 20 日
3. 正式论文提交日期: 2007 年 5 月 10 日
4. 会议召开日期: 2007 年 9 月 14-16 日