

时间 Petri 网的可调度分析及在 FMS 中的应用

翟正利^{1,2} 丁志军²

(青岛理工大学计算机工程学院 青岛 266520)¹ (同济大学电子与信息工程学院 上海 201804)²

摘 要 在实时系统中,检查任务执行的计划是否满足要求的时间约束称为可调度分析。通过把时间特性与其他行为特性分离,提出了一种以时间 Petri 网建模的实时系统调度分析方法。如果特定任务的执行是可调度的,则可以计算任务执行的时间跨度,否则确定出不可调度的变迁以便于调整时间约束和纠正设计错误。提出了一种通过把复杂的任务序列分解成一些子序列来进行可调度性分析的综合时序分析技术,它不仅提高了效率,也有助于关于调度的可达性问题的讨论。讨论了柔性制造系统 FMS 中的车间装配子系统的可调度性。

关键词 可调度分析,时间 Petri 网,实时系统,柔性制造系统

中图分类号 TP301 **文献标识码** A **DOI** 10.11896/j.issn.1002-137X.2015.1.003

Schedulability Analysis of Time Petri Net and its Application in FMS

ZHAI Zheng-li^{1,2} DING Zhi-jun²

(College of Information Engineering, Qingdao Technological University, Qingdao 266520, China)¹

(College of Electronics and Information Engineering, Tongji University, Shanghai 201804, China)²

Abstract In a real-time system, the process of verifying whether a schedule of task execution meets the imposed timing constraints is referred to as scheduling analysis. This paper presented an approach to analyze the scheduling of real time systems modeled in time Petri nets by separating timing properties from other behavioral properties. If a specific task execution is schedulable, we could calculate the time span of the task execution, otherwise pinpointed out non-schedulable transitions to help adjust timing constraints and correct design error. A technique for compositional timing analysis was also proposed to deal with complex task sequences by decomposing it into some simple sub-task sequences, which not only improves efficiency but also facilitates the discussion of reachability issue with regard to scheduling. Finally, scheduling analysis of an assembly system in flexible manufacturing systems (FMS) was discussed.

Keywords Schedulability analysis, Time Petri nets, Real-time systems, Flexible manufacturing system

1 引言

在实时系统中,检查任务执行的计划是否满足要求的时间约束称为可调度分析,很多学者通过实时系统的执行或规范(如时序逻辑技术)来解决这一问题^[1-3]。

作为一种虚拟模型,时间 Petri 网(TPN)^[4]已被证明对表达时间依赖系统的时间约束非常方便。可达图(树)^[5,6]提供了 TPN 的完整动态行为的表示方法。结点为状态类,边标以发生的变迁和反映时间约束的发生域。可调度性与可达性密切相关,但主要涉及变迁序列而不是标识或状态。为验证动态行为而构造的状态图对可调度分析并不十分有效。

本文主要研究 TPN 的可调度分析,主要结果包括:1)确定一个特定变迁序列是否可调度的方法,并可计算一个可调度任务执行的时间跨度,或者指出不可调度变迁,以帮助调整时间约束和纠正设计错误。2)处理复杂任务序列的一个分解

合成方法。

本文第 2 节介绍了实时系统可调度分析的相关工作;第 3 节给出 TPN 及其可调度性的主要介绍;第 4 节说明如何把时间特性从其它行为特性分开以便进行任务执行的可调度分析的时间验证;第 5 节描述了如何把基础 Petri 网(称为源网)的发生序列分解成一些子序列来进行可调度性分析;第 6 节讨论了柔性制造系统(FMS)中的车间装配子系统的可调度性;最后总结全文。

2 相关工作

许多研究人员对实时系统中的可调度性分析问题进行了研究。Tsai 等人^[3]给出了将实时系统的调度模型转换为时间约束 Petri 网(TCPN)的方法,TCPN 对 Petri 网的每个变迁和库所关联了最小/最大时序约束,对每个变迁的引发关联了一个持续时间约束,TCPN 使用弱引发规则,表达能力更

到稿日期:2014-06-05 返修日期:2014-07-22 本文受国家重点基础研究发展计划(973 计划)(2010CB328101),国家自然科学基金(61173042,61173181),上海市自然科学基金(13ZR1443100),山东省优秀中青年科学家科研奖励基金(BS2010DX009),青岛市科技计划项目(12-1-4-4-(7)-JCH)资助。

翟正利(1972—),男,博士,副教授,CCF 会员,主要研究方向为 Petri 网、信息服务、网格/云计算、智能信息处理,E-mail: zhaizhl@163.com; 丁志军(1974—),男,博士,副教授,主要研究方向为信息服务、语义 Web、Petri 网、并发处理。

强,但当需要分析行为和时间特性时,TCPN难以解决一般的可达性问题。因为不同的引发规则和对时间约束的解释不同,TCPN的调度分析并不适用于TPN,并且没有一个软件工具实现。Stoyenko等人^[7]利用硬件相关的信息提供了准确的最差情况的时间界以及其他调度信息,提出了一套基于程序实现信息的语言独立的调度技术。Haban和Shin^[2]提出了一个监控和验证任务执行的方法,利用实时监控器测量的纯执行时间和资源共享延迟来降低最差执行时间和估计执行时间的差异,以确定最早可能时间是否满足其截止期限。Jahanian和Mok^[8]基于实时逻辑,把安全断言和系统的形式规范关联起来,提出了实时系统的时间特性的形式化安全分析方法,该方法尤其适于系统的时间行为的推理。可达图(树)基于交错语义提供了TPN完整的动态行为的表示。节点是状态类,边上标记反映时序约束的引发变迁及发生域。但是,实时系统中的一个重要问题——执行任务中的端到端延迟,不能直接从状态类的发生域导出。因此,文献^[9,10]中开发的技术对可达性分析有用而不适用于可调度性分析。

Furfaro和Negro^[11]也给出了可调度性分析的TPN模型,然后将TPN模型翻译成时间自动机模型,并用建模工具UPPAAL验证^[12]。转换TPN到相应的时间自动机模型的形式化机制由Gu和Shin^[13]提出。然而,时间自动机模型只能处理抢占或非抢占式执行,不能直接应用于实时系统的可调度性分析。

3 时间Petri网及可调度性

由于篇幅关系,这里只给出同本文密切相关的概念,Petri网的其它概念可参考文献^[14-16]。

定义1 一个四元组 $\Sigma = (P, T; F, M)$ 称为Petri网,其中:

P 和 T 是两不相交集,分别称为库所集和变迁集;

F 是 P 和 T 之间的弧集,称为网的流关系;

$M: P \rightarrow \{0, 1, 2, \dots\}$ 是 Σ 的一个标识。

定义2 设Petri网 $\Sigma = (P, T; F, M)$, 标识 M 下使能的变迁集合定义为 $Enabled(M) = \{t \mid t^- \leq M\}$, 其中 t^- 为变迁 t 的前置诱导标识。

在含时间因素的Petri网模型中,时间的概念可依据所描述问题的需要关联到Petri网的不同元素上,变迁的使能和发生规则及时间限制也有所不同。

本文研究的TPN最早由Merlin^[4]提出,后来由Berthomieu、Diaz^[9]和Popova等人^[17]对其时间区间的含义进行了修改。

定义3 一个五元组 $Z = (P, T; F, M_0, I)$ 称作时间Petri网(TPN)当且仅当:

① $\Sigma = (P, T; F, M_0)$ 是一个Petri网,称为 Z 的源网;

② $I: T \rightarrow Q^+ \times (Q^+ \cup \{\infty\})$ 。

其中, Q^+ 表示非负有理数, $\forall t \in T, I(t) = [EFT(t), LFT(t)]$, $EFT(t) \leq LFT(t)$ 。

I 称作 Z 的时间函数, $EFT(t)$ 和 $LFT(t)$ 分别称作 t 的最早发生时间和最晚发生时间。

定义4 设 $I_1 = [a_1, b_1]$, $I_2 = [a_2, b_2]$, 其中 $0 \leq a_i \leq b_i < \infty, k \in N$, 定义:

$$I_1 + I_2 = [a_1 + a_2, b_1 + b_2]$$

$$I_1 - I_2 = [a_1 - a_2, b_1 - b_2]$$

$$k * I_1 = [k * a_1, k * b_1]$$

定义5(强发生语义) 如果变迁 t (在时刻 τ 使能) 未发生且未被其它变迁的发生取消使能, 则 t 至少必在 $\tau + LFT(t)$ 时刻发生。

定义6 在TPN中, 如果使能变迁 t 在标识 M 下可作为第一个发生的变迁(即其能在其它使能变迁之前发生), 则称 t 在标识 M 下是可调度的。

图1(a)中, 处于冲突结构中的 t_1 和 t_2 在当前标识下是可调度的, 尽管其中一个的发生使得另一个在新标识下不再使能。图1(b)中, 只有 t_1 是可调度的, t_2 是不可调度的, 因为在 t_2 有机会发生之前 t_1 必须发生。图1(c)中, t_1 和 t_2 都是可调度的, 在其中一个发生后所得到的新标识下另一个仍然是使能和可调度的。

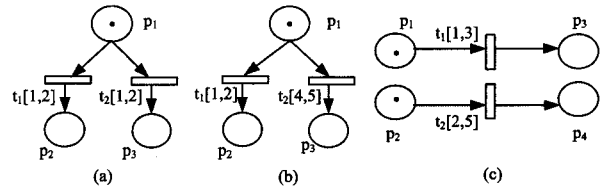


图1 变迁的可调度性

变迁的可调度性与当前标识下使能变迁的时间约束有关, 若 $EFT(t) \leq \min\{LFT(t') \mid t' \in Enabled(M)\}$, 则变迁 t 在标识 M 下是可调度的。

定义7 在TPN中, 如果变迁序列 $\delta = t_1 \dots t_i \dots t_n$ 中的所有变迁按给定顺序都是可调度的, 即存在标识序列 $M_0, \dots, M_n$, 使得 $M_0 t_1 M_1 \dots t_i M_i \dots t_n M_n$ 在源网中是一个发生序列并且 t_i 在 M_{i-1} 下是可调度的, 则称 δ 是可调度的或 δ 是TPN中的一个调度。如果 δ 中至少有一个变迁是不可调度的, 则 δ 就是不可调度的。

定义8 $L(M_0, M_n)$ 表示 Σ 中从 M_0 到 M_n 的所有可能发生序列的集合。

如果一个变迁序列不是源网 Σ 中的一个出现序列, 则它在TPN中就不是可调度的。然而, Σ 中的出现序列在TPN中不一定是可调度的, 并且 Σ 中的可达标识在TPN中不一定可达。本文主要研究源网 Σ 中的出现序列在TPN中是否是可调度的。

4 可调度分析的时间验证

通常, TPN状态类的相对发生域可用来确定变迁的可调度性, 假设 D_i 是在 M_i 使能变迁的相对发生域。变迁 t 在 D_i 中的动态发生区间 $D_i(t)$ 表示为 $[REFT_i(t), RLFT_i(t)]$, 其中 $REFT_i(t)$ 和 $RLFT_i(t)$ 分别称为相对最早发生时间和相对最晚发生时间。变迁 t 的静态发生区间 $C(t) = [EFT(t), LFT(t)]$ 。

设 $D_0 = \{C(t) \mid t \in Enabled(M_0)\}$, $[RE_i, RL_i]$ 为 t_i 的相对可调度区间。 t_{i+1} ($0 \leq i \leq n-1$) 在发生序列 $M_0 t_1 \dots t_n M_n$ 的可调度性可通过下列步骤来检查:

Step 1 如果 $t_{i+1} \in Enabled(M_i) \wedge REFT_i(t_{i+1}) \leq \min\{RLFT_i(t) \mid t \in Enabled(M_i)\}$, 则 t_{i+1} 在标识 M_i 下, 在区间 $[REFT_i(t_{i+1}), \min\{RLFT_i(t) \mid t \in Enabled(M_i)\}]$ (记为 $[RE_{i+1}, RL_{i+1}]$) 中是可调度的; 否则 t_{i+1} 在标识 M_i 下是不可调度的。

调度的。

Step 2 建立新的相对发生区间:

① $D_{i+1} := \emptyset$;

② $\forall t \in Enabled(M_{i+1}) - Enabled(M_i)$, 把其静态时间区间 $C(t)$ 加入 D_{i+1} ;

③ $\forall t \in Enabled(M_i) \cap Enabled(M_{i+1})$, $t \neq t_{i+1}$, 把区间 $[Max\{0, REFT_i(t) - RL_{i+1}\}, RLFT_i(t) - RE_{i+1}]$ 加入 D_{i+1} ;

④ 如果 $t_{i+1} \in Enabled(M_i) \cap Enabled(M_{i+1})$, 则把其静态区间加入 D_{i+1} 。

下面检查图 1(c) 中 $t_1 t_2$ 的可调度性, 其中, $M_0 = \{p_1, p_3\}$, $D_0 = \{t_1[1, 3], t_2[2, 5]\}$;

(1) 检查 t_1 : t_1 在 $[1, Min\{3, 5\}] = [1, 3]$ 内是可调度的; $//M_1 = \{p_2, p_3\}$;

$D_1 = \{t_2[Max\{2-3, 0\}, 5-1]\} = \{t_2[0, 4]\}$;

(2) 检查 t_2 : t_2 在 $[0, 4]$ 间是可调度的; $//M_2 = \{p_2, p_4\}$,

$D_2 = \emptyset$ 。

根据上述步骤, t_1 在相对于时间 τ (在 M_0) 的 θ_1 ($1 \leq \theta_1 \leq 3$) 发生, t_1 发生到达标识 M_1 ; t_2 在相对于时间 $\tau + \theta_1$ (在 M_1) 的 θ_2 ($0 \leq \theta_2 \leq 4$) 发生, t_2 发生到达标识 M_2 , 时间应是 $\tau + \theta_1 + \theta_2$ 。 $[1, 3] + [0, 4] = [1, 7]$ 是 $\theta_1 + \theta_2$ 的区间吗? 回答是否定的。至少这是不准确的。 $t_1 t_2$ 的正确时间跨度是 $[2, 5]$, 因为 t_2 在 M_0 是使能的, 它必须在 $[2, 5]$ 内的某个时间发生。类似地, $t_2 t_1$ 也是可调度的。 t_2 的相对可调度区间是 $[2, \min\{3, 5\}] = [2, 3]$, t_1 的相对可调度区间是 $[\max\{1-3, 0\}, 3-2] = [0, 1]$ 。 $[2, 3] + [0, 1] = [2, 4]$ 也不是 $t_2 t_1$ 的正确时间跨度。由于 t_1 必须在 $[1, 3]$ 内发生, $t_2 t_1$ 的正确时间跨度应为 $[2, 3]$ 。实际上, 根据相对发生域不能准确地确定一个出现序列的时间跨度。问题是由并发变迁的时间约束引起的, 对于在 M_i 和 M_{i+1} 都使能的变迁 t_i , 相对区间 $[MAX\{0, REFT_i(t) - RL_{i+1}\}, RLFT_i(t) - RE_{i+1}]$ 失去了计算时间跨度的时间信息。例如, 在 M_i 的相对区间不一定等于 t_{i+1} 在 M_i 的可调度区间加上在 M_{i+1} 的相对区间, 即

$$[REFT_i(t), RLFT_i(t)] \neq [RL_{i+1}, RE_{i+1}] + [MAX\{0, REFT_i(t) - RL_{i+1}\}, RLFT_i(t) - RE_{i+1}]$$

为解决上述问题, 在绝对区间(相对于 M_0 的 τ)基础上对使能变迁引入绝对发生域。对可达标识引入全局时间戳。

在发生一个变迁后, 可达标识被打上时间戳, 并构造新使能和仍保持使能变迁的绝对发生域。假设 $TS_i = [AE_i, AL_i]$ 是 M_i 的时间戳, 即 M_i 在区间 $[AE_i, AL_i]$ 内的某个时刻到达。 AD_i ($1 \leq i \leq n$) 是在 M_i 使能变迁的绝对发生域, 变迁 t 在 AD_i 中的区间记为 $AD_i(t) = [AEFT_i(t), ALFT_i(t)]$, 其中 $AEFT_i(t)$ 和 $ALFT_i(t)$ 分别称为绝对最早发生时间和绝对最晚发生时间。初始值 $AD_0(t) = \{C(t) | t \in Enabled(M_0)\}$, 令 M_0 的时间戳 $TS_0 = [0, 0]$, Σ 中的一个发生序列 $\sigma = M_0 t_1 M_1 \dots t_i M_i \dots t_n M_n$ 在 TPN 中是否可调度可以通过如下步骤检查每个变迁 t_{i+1} ($0 \leq i \leq n-1$) 来确定:

① 若 $t_{i+1} \notin Enabled(M_i)$, 则 σ 不是 Σ 中的一个出现序列, 因此它是不可调度的。

② 若 $REFT_i(t_{i+1}) \leq \min\{RLFT_i(t) | t \in Enabled(M_i)\}$, 则 t_{i+1} 在标识 M_i 是可调度的, 转③; 否则 t_{i+1} 在 M_i 是不可调度的。

③ 计算 t_{i+1} 的相对可调度区间: $[RE_{i+1}, RL_{i+1}] = [REF-$

$$T_i(t_{i+1}), \min\{RLFT_i(t) | t \in Enabled(M_i)\}]$$

计算 t_{i+1} 的绝对可调度区间即时间戳:

$$TS_{i+1} = [AE_{i+1}, AL_{i+1}]$$

$$= [AEFT_i(t_{i+1}), \min\{ALFT_i(t) | t \in Enabled(M_i)\}]$$

④ 从 D_i 建立新的相对发生域 D_{i+1} :

$$D_{i+1} := \emptyset$$

$$D_{i+1} := D_{i+1} \cup \{[EFT(t_{i+1}), LFT(t_{i+1})] | t \notin Enabled(M_i) \wedge t \in Enabled(M_{i+1})\}$$

$$D_{i+1} := D_{i+1} \cup \{[MAX\{0, REFT_i(t) - RL_{i+1}\}, RLFT_i(t) - RE_{i+1}] | t \in Enabled(M_i) \cap Enabled(M_{i+1}) \wedge t \neq t_{i+1}\}$$

$$D_{i+1} := D_{i+1} \cup \{[EFT(t_{i+1}), LFT(t_{i+1})] | t_{i+1} \in Enabled(M_{i+1})\}$$

⑤ 从 AD_i 建立新的绝对发生域:

$$AD_{i+1} := \emptyset$$

$$AD_{i+1} := AD_{i+1} \cup \{[EFT(t_{i+1}), LFT(t_{i+1})] + TS_{i+1} | t \notin Enabled(M_i) \wedge t \in Enabled(M_{i+1})\}$$

$$AD_{i+1} := AD_{i+1} \cup \{[MAX\{AEFT_i(t), AE_{i+1}\}, ALFT_i(t)] | t \in Enabled(M_i) \cap Enabled(M_{i+1}) \wedge t \neq t_{i+1}\}$$

$$AD_{i+1} := AD_{i+1} \cup \{[EFT(t_{i+1}), LFT(t_{i+1})] + TS_{i+1} | t_{i+1} \in Enabled(M_{i+1})\}$$

在上述算法中, $Enabled(M_{i+1})$ ($0 \leq i \leq n-1$) 可直接通过源网 Σ 的可达树得到, 相对发生域 D_i 用于确定变迁的可调度性, 绝对发生域 AD_i 用于计算变迁发生和得到新标识的时间, AD_i 和 D_i 的计算与步骤①、②的前置条件无关。时间戳 TS_{i+1} 是变迁 t_{i+1} 到达 M_{i+1} 的绝对可调度区间, 若该序列是可调度的, 则执行该序列的时间跨度为 TS_n , 这可通过下面的归纳来说明:

1) 根据可调度性的定义, 变迁 t_1 在 $[EFT(t_1), \min\{LFT(t) | t \in Enabled(M_0)\}]$ 内是可调度的, 由于对 $\forall t \in Enabled(M_0)$, 有 $[EFT(t), LFT(t)] = [AEFT_0(t), ALFT_0(t)]$, t_1 的可调度区间等于 $TS_1 = [AE_1, AL_1] = [AEFT_0(t_1), \min\{ALFT_0(t) | t \in Enabled(M_0)\}]$, TS_1 是到达 M_1 的时间戳。此外, AD_1 包括在 M_1 使能的所有变迁的正确的绝对区间: 对任何新使能的变迁 t , t 可发生的绝对区间为静态区间加上 t_1 的绝对可调度区间(即 TS_1); 对所有在 M_0 和 M_1 使能的变迁 $t \neq t_1$, 因为 t_1 在 AE_1 或在 AE_1 之后发生并且 t 在 t_1 之后发生, 所以 t 在 AE_1 之前不会发生, 因此, $MAX\{AEFT_0(t), AE_1\}$ 是 t 能发生的最早绝对时间; 如果 t_1 在 M_1 下仍然使能, 则 t_1 在 AD_1 中的新绝对区间为其静态区间加上 TS_1 (就像一个新使能变迁一样)。

2) 假设 TS_i 是变迁 t_i 发生的时间跨度, AD_i 包括在 M_i 使能的所有变迁的正确绝对区间, 很明显, t_{i+1} 不会在 $AEFT_i(t_{i+1})$ 之前发生, 并且 t_{i+1} 必须在其余使能变迁强制发生之前发生, 即 t_{i+1} 的发生区间是 $[AEFT_i(t_{i+1}), \min\{ALFT_i(t) | t \in Enabled(M_i)\}]$, 这正是 $t_1 t_2 \dots t_{i+1}$ 的发生时间跨度, 这个区间正好等于 TS_{i+1} 。类似地, 容易证明 AD_{i+1} 也包括在 M_{i+1} 下所有使能变迁的正确绝对区间。

进一步地, 可很容易地知道一个可调度出现序列中的两

个变迁或两个标识之间的时间跨度。实际上,序列 $t_i \cdots t_j$ 的时间跨度是 $TS_j - TS_i$ 。这有助于变迁序列的合成,因为一个不是从 M_0 开始的序列也可以分析。

例如,图 2(a)中 $\delta = (t_1 t_2 t_3 t_4 t_5)$ 的可调度性检查如下。

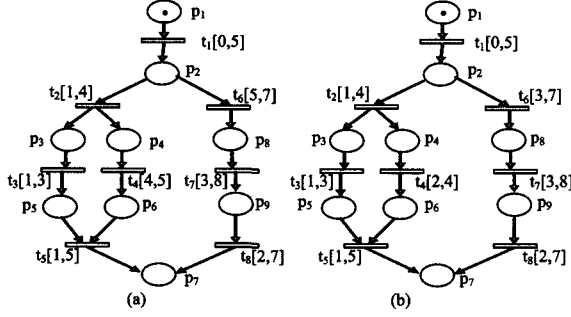


图 2 变迁序列的可调度性

1) 初始时间戳 $TS_0 = [0, 0]$; $//M_0 = \{p_1\}$

$D_0 = \{t_1[0, 5]\}$

$AD_0 = \{t_1(0, 5)\}$

2) 检查 t_1 :

t_1 在动态相对区间 $[0, 5]$ 内可调度;

$TS_1 = [0, 5]$; $//M_1 = \{p_2\}$

$D_1 = \{t_2[1, 4], t_6[5, 7]\}$

$AD_1 = \{t_2: [1, 4] + [0, 5] = [1, 9], t_6: [5, 7] + [0, 5] = [5, 12]\}$

这是因为 t_2, t_6 皆为新使能变迁。

3) 检查 t_2 :

$\because REFT_1(t_2) = 1$

$\text{Min}\{RLFT_1(t_2) = 4, RLFT_1(t_6) = 7\} = 4$

$REFT_1(t_2) \leq \text{Min}\{RLFT_1(t_2) = 4, RLFT_1(t_6) = 7\}$

$\therefore t_2$ 在 $[1, 4]$ 中是可调度的。 $//M_2 = \{p_3, p_4\}$

$TS_2 = [1, 9]$

$D_2 = \{t_3[1, 3], t_4[4, 5]\}$

$AD_2 = \{t_3: [1, 3] + [1, 9] = [2, 12],$

$t_4: [4, 5] + [1, 9] = [5, 14]\}$

4) 检查 t_3 :

$\because REFT_2(t_3) = 1$

$\text{Min}\{RLFT_2(t_3) = 3, RLFT_2(t_4) = 5\} = 3$

$REFT_2(t_3) \leq \text{Min}\{RLFT_2(t_3) = 3, RLFT_2(t_4) = 5\}$

$\therefore t_3$ 在 $[1, 3]$ 内是可调度的。 $//M_3 = \{p_5, p_4\}$

$TS_3 = [2, 12]$

$D_3 = \{t_4[1, 4]\}$

$\because t_4$ 不是新使能变迁,

$\therefore AD_3 = \{t_4[5, 14]\}$ 。

5) 检查 t_4 : t_4 在 $[1, 4]$ 内可调度。 $//M_4 = \{p_5, p_6\}$

$TS_4 = [5, 14]$

$D_4 = \{t_5[1, 5]\}$

$\because t_5$ 是新使能变迁, $\therefore AD_4 = \{t_5: [1, 5] + [5, 14] = [6,$

$19]\}$ 。

6) 检查 t_5 : t_5 在 $[1, 5]$ 内可调度。 $//M_5 = \{p_7\}$

$TS_5 = [6, 19]$

$D_5 = \emptyset, AD_5 = \emptyset$

因此 δ 是可调度的, 且 δ 的时间跨度为 $TS_5 = [6, 19]$ 。

其中一个子序列 $t_3 t_4$ 的时间跨度是 $TS_4 - TS_2 = [5, 14] - [1, 9] = [4, 5]$, 即发生 t_3 和 t_4 需占用 4 到 5 个时间单元, 这符合施加在并发结构上的时间约束。注意, 绝对发生域不能用来确定变迁的可调度性。在上述例子中, 发生 t_1 之后的绝对发生域 $AD_1 = \{t_2[1, 9], t_6[5, 12]\}$, 从 AD_1 来看, 似乎冲突结构中的 t_2 或 t_6 是可调度的。但根据 t_2 和 t_6 的时间约束, t_6 是不可调度的。类似地, 从 $AD_2 = \{t_3[2, 12], t_4[5, 14]\}$ 来看, 似乎并发结构中的 t_3 和 t_4 能按 $t_3 t_4$ 或 $t_4 t_3$ 两种顺序发生, 但根据时间约束, t_3 必须在 t_4 前发生。这就是为什么要使用相对发生域和绝对发生域来分别确定各个变迁的可调度性和变迁序列的时间跨度的原因。

类似地, 图 2(a)中 $\delta_1 = (t_1 t_2 t_4 t_3 t_5)$ 是不可调度的。根据 $M_2 = \{p_3, p_4\}, D_2 = \{t_3[1, 3], t_4[4, 5]\}$ 和 $REFT_2(t_4) = 4 > \text{Min}\{RLFT_2(t_3) = 3, RLFT_2(t_4) = 5\} = 3$, t_4 不能在 t_3 之前发生。现在考虑 $\delta_2 = (t_1 t_6 t_7 t_8)$, t_1 可调度, 其发生得到标识 $M_1 = \{p_2\}, D_1 = \{t_2[1, 4], t_6[5, 7]\}$ 。很明显 t_2 可调度而 t_6 不可调度。因此 δ_2 是不可调度的。注意, $L(M_0, M_n) = \{\delta, \delta_1, \delta_2\}$, 即 δ, δ_1 和 δ_2 正好是到达标识 $M_n = \{p_7\}$ 的 3 个可能发生序列。然而 M_n 只能通过 δ 可达。在图 2(b)中, 其源网同图 2(a)的源网相同, 但 t_4 和 t_6 的静态区间分别替换成了 $[2, 4]$ 和 $[3, 7]$ 。在这种情形下, δ, δ_1 和 δ_2 都是可调度的, 其时间跨度分别为 $[4, 18], [4, 17]$ 和 $[8, 24]$ 。 $t_3 t_4$ 的跨度为 $[2, 4]$, 然而 $t_4 t_3$ 的跨度为 $[2, 3]$ 。应该注意到, δ_2 中 t_6 的动态区间是 $[3, 4]$, 因此 t_6 的绝对发生区间为 $[0, 5] + [3, 4] = [3, 9]$ 。

5 可调度性的综合分析

本节研究如何把源网中的一个发生序列分解为一些子序列来进行可调度性分析。正是由于分解、合成, 一些子序列的分析结果在检查其他序列时可以重用。包含重复子序列的发生序列的分析可被简化, 这不仅降低了复杂度, 而且有助于讨论可达性问题。

如上所述, 一个调度中的任何一个标识都标记了一个绝对时间区间(相对于初始标识)。很容易从一个给定调度得到任意两个标识或变迁之间的时间跨度。为了方便分解、合成, 在这里用发生序列而不是出现序列, 并把可调度性分析扩展到更一般的情况。一个序列 δ 可从任意一个可达标识而不一定从 M_0 开始, 即 δ 允许是一个发生序列的子序列。检查 δ 可调度性的算法被表示为映射 $\phi: S^T \rightarrow Q^* \times Q^*$, 其中 S^T 为源网中所有发生(子)序列的集合, 如果 δ 不可调度, 则 $\phi(\delta) = [0, 0]$, 否则 $\phi(\delta) = TS_n$, TS_n 是相对于开始时间的的时间跨度。

定义 9 设 $\delta_1 = (M_{10} t_{11} M_{11} \cdots t_{1m} M_{1m}), \delta_2 = (M_{20} t_{21} M_{21} \cdots t_{2n} M_{2n})$ 为源网中的两个序列($m \geq 1, n > 1$), 其中 M_{10} 和 M_{20} 都从 M_0 可达。 δ_2 是同 δ_1 可合成的(记为 $\delta_1 + \delta_2$)当且仅当 $M_{1m} = M_{20}$ 并且 $\text{Enabled}(M_{1m}) \cap \text{Enabled}(M_{1m-1}) - \{t_{1m}\} = \emptyset$ 。

$\text{Enabled}(M_{1m}) \cap \text{Enabled}(M_{1m-1}) - \{t_{1m}\} = \emptyset$ 说明 t_{21} 为新使能的, 意味着 M_{1m} 和 M_{1m-1} 除了 t_{1m} 之外不会有其它共同使能的变迁, 即 M_{1m} 使能的所有变迁都是新使能的。如果 t_{1m} 在 M_{1m} 仍然使能, 则认为它是一个新使能的。

例如, 在图 2(a)中, 因为, $\text{Enabled}(M_2) = \{t_3, t_4\}, \text{Enabled}(M_1) = \{t_2, t_6\}, \text{Enabled}(M_2) \cap \text{Enabled}(M_1) = \emptyset$, 所以 $(M_2 t_3 M_3 t_4 M_4 t_5 M_5)$ 同 $(M_0 t_1 M_1 t_2 M_2)$ 可合成。

$(M_3 t_4 M_4 t_5 M_5)$ 同 $(M_0 t_1 M_1 t_2 M_2 t_3 M_3)$ 不可合成, 因为 $Enabled(M_3) = \{t_4\}$, $Enabled(M_2) = \{t_3, t_4\}$, 其交集为 $\{t_4\} \neq \emptyset$, 即 $(t_1 t_2 t_3 t_4 t_5)$ 不能分解为 $(t_1 t_2 t_3)$ 和 $(t_4 t_5)$ 。

一般情况下, 当分解一个序列时把并发变迁分开是不正确的。

显然, 序列的合成是可结合的, 即 $(\delta_1 + \delta_2) + \delta_3 = \delta_1 + (\delta_2 + \delta_3)$ 。后面将 $\delta_1 + \delta_2 + \dots + \delta_k$ 简记为 $\delta_1 \delta_2 \dots \delta_k$ 。

定理 1 设 δ_2 同 δ_1 可合成。 $\delta_1 \delta_2$ 可调度 $\Leftrightarrow \delta_1$ 和 δ_2 都可调度。且若可调度, 则 $\psi(\delta_1 \delta_2) = \psi(\delta_1) + \psi(\delta_2)$ 。

证明: 设 $\delta_1 = (M_{10} t_{11} M_{11} \dots t_{1m} M_{1m})$

$\delta_2 = (M_{20} t_{21} M_{21} \dots t_{2n} M_{2n})$,

$\delta_1 \delta_2 = (M_{10} t_{11} M_{11} \dots t_{1m} M_{1m} t_{21} M_{21} \dots t_{2n} M_{2n})$

1) 假设 δ_1 和 δ_2 皆可调度, 则存在检查 δ_1 和 δ_2 可调度的两个相对发生域序列 $(D_{10} D_{11} \dots D_{1m})$ 和 $(D_{20} D_{21} \dots D_{2n})$ 。由于 δ_2 同 δ_1 可合成, 则有 $M_{1m} = M_{20}$, 并且 $Enabled(M_{1m}) \cap Enabled(M_{1m-1}) - \{t_{1m}\} = \emptyset$ 。因此, $D_{1m} = \{[EFT(t), LFT(t)] | t \in Enabled(M_{1m})\}$, 即对于 M_{1m} 使能的任何变迁, 其动态区间正好就等于其静态区间。

另外, $D_{20} = \{[EFT(t), LFT(t)] | t \in Enabled(M_{20})\}$, 因此 $D_{1m} = D_{20}$ 并且 $(D_{10} D_{11} \dots D_{1m} D_{21} \dots D_{2n})$ 恰是检查 $\delta_1 \delta_2$ 可调度性的相对发生域的序列。所以 $\delta_1 \delta_2$ 是可调度的。

类似地, 存在检查 δ_1 和 δ_2 可调度性的两个时间戳序列 $TS_{10} \dots TS_{1m}$ 和 $TS_{20} \dots TS_{2n}$, 其中, $TS_{10} = TS_{20} = [0, 0]$, $\psi(\delta_1) = TS_{1m}$, $\psi(\delta_2) = TS_{2n}$, 也存在检查 δ_1 和 δ_2 可调度性的两个绝对发生域序列 $(AD_{10} AD_{11} \dots AD_{1m})$ 和 $(AD_{20} AD_{21} \dots AD_{2n})$ 。

$AD_{1m} = \{[EFT(t), LFT(t)] + TS_{1m} | t \in Enabled(M_{20})\}$,

$AD_{20} = \{[EFT(t), LFT(t)] | t \in Enabled(M_{20})\}$ 。

令 $AD'_{2j} = \{I + TS_{1m} | I \in AD_{2j}\}$, $TS'_{2j} = TS_{2j} + TS_{1m}$, $(0 < j < n+1)$ 。则 $(AD_{10} AD_{11} \dots AD_{1m} AD'_{21} \dots AD'_{2n})$ 恰为绝对发生域序列, 且 $(TS_{10} TS_{11} \dots TS_{1m} TS'_{21} \dots TS'_{2n})$ 恰为检查 $\delta_1 \delta_2$ 可调度性的时间戳序列。所以 $\psi(\delta_1 \delta_2) = TS'_{2n} = TS_{2n} + TS_{1m} = \psi(\delta_2) + \psi(\delta_1)$ 。

2) 假设 $\delta_1 \delta_2 = (M_{10} t_{11} \dots M_{1m} t_{21} M_{21} \dots D_{2n})$ 是可调度的, 则存在相对发生域的一个序列 $(D_{10} D_{11} \dots D_{1m} D_{21} \dots D_{2n})$ 。显然, $(D_{10} D_{11} \dots D_{1m})$ 和 $(D_{20} D_{21} \dots D_{2n})$ 分别为检查 δ_1 和 δ_2 的相对发生域序列, 所以 δ_1 和 δ_2 都是可调度的。

类似地, 存在一个时间戳序列 $(TS_{10} TS_{11} \dots TS_{1m} TS_{21} \dots TS_{2n})$ 且 $\psi(\delta_1 \delta_2) = TS_{2n}$ 。显然 $(TS_{10} TS_{11} \dots TS_{1m})$ 是检查 δ_1 的时间戳序列, $\psi(\delta_1) = TS_{1m}$ 。设 $TS'_{20} = [0, 0]$, $TS'_{2j} = TS_{2j} - TS_{1m}$, $(TS'_{20} TS'_{21} \dots TS'_{2n})$ 为检查 δ_2 的时间戳, $\psi(\delta_2) = TS'_{2n} = TS_{2n} - TS_{1m}$, 所以 $\psi(\delta_1) + \psi(\delta_2) = TS_{1m} + TS_{2n} - TS_{1m} = TS_{2n} = \psi(\delta_1 \delta_2)$ 。

如果 TS_{1m} 包括无限时间, 则 TS_{2j} $(0 < j < n+1)$ 都包括无限时间。

类似地, 也可证明 $\psi(\delta_1 \delta_2) = \psi(\delta_1) + \psi(\delta_2)$ 。

根据 1) 和 2), 定理成立。

如果 δ_2 同 δ_1 不可合成, 分别检查 δ_1 和 δ_2 并不能提供分析 δ_1 和 δ_2 合成的任何有用信息。原因在于初始相对和绝对发生域中的动态区间总等于静态区间。例如在图 2(a) 中, 设 $\delta_1 = (M_0 t_1 M_1 t_2 M_2 t_3 M_3)$, $\delta_2 = (M_3 t_4 M_4 t_5 M_5)$, $\delta = (M_0 t_1 M_1 \dots t_5 M_5)$ 。这时 δ , δ_1 和 δ_2 都可调度且 $\psi(\delta) = [6, 19]$, $\psi(\delta_1) =$

$[2, 12]$, $\psi(\delta_2) = [5, 15]$, 很明显 $\psi(\delta) \neq \psi(\delta_1) + \psi(\delta_2)$, 因此, 不能由 δ_1 和 δ_2 来分析 δ 的可调度性。

定理 2 设 δ_i $(1 \leq i \leq k)$ 为序列且 δ_i $(2 \leq i \leq k)$ 与 δ_{i-1} 可合成。

$\delta_1 \dots \delta_k$ 是可调度的 $\Leftrightarrow \delta_i$ 都可调度。

如果 $\delta_1 \dots \delta_k$ 可调度, 则 $\psi(\delta_1 \dots \delta_k) = \sum_{i=1}^k \psi(\delta_i)$ 。

证明: $k=2$ 时, 定理 2 即为定理 1 的特例。

如果 $k=3$, 则 $\delta_1 \delta_2 \delta_3 = (\delta_1 \delta_2) \delta_3$ 。令 $\delta = \delta_1 \delta_2$, $\delta_1 \delta_2 \delta_3 = \delta \delta_3$ 。 $\delta \delta_3$ 是可调度的 $\Leftrightarrow \delta$ 和 δ_3 都可调度 $\Leftrightarrow \delta_1, \delta_2$ 和 δ_3 都可调度。

假设 $\delta = \delta_1 \dots \delta_{k-1}$ 是可调度的 $\Leftrightarrow \delta_i$ $(1 \leq i \leq k-1)$ 都可调度。 $\delta_1 \dots \delta_k = \delta \delta_k$, $\delta_1 \dots \delta_k$ 可调度 $\Leftrightarrow \delta$ 和 δ_k 可调度 $\Leftrightarrow \delta_i$ $(1 \leq i \leq k)$ 皆可调度, 定理成立。

定理 2 表明, 只要可能和有必要, 可调度性可通过把一个序列分解为一些子序列来分析。

定理 3 假设序列 δ_2 同自己可合成且同 δ_1 可合成, δ_3 同 δ_2 可合成。令 $\delta = (\delta_2)^k$, $\delta_1 \delta \delta_3$ 可调度 $\Leftrightarrow \delta_1 \delta_2 \delta_3$ 可调度。

如果 $\delta_1 \delta_2 \delta_3$ 可调度, 则 $\psi(\delta_1 \delta_2^k \delta_3) = \psi(\delta_1 \delta_2 \delta_3) + (k-1) * \psi(\delta_2)$ 。

证明: 若 $\delta_2 = (M_{20} t_{21} M_{21} \dots t_{2m} M_{2m})$ 可自合成, 则 $M_{20} = M_{2m}$, 由定理 2 直接得证。

定理 2 和定理 3 的重要意义在于, 它们不仅简化了包含循环的那些序列的可调度性分析, 而且有助于分析可达性, 因为基于相对和绝对时间模式, 可调度分析可以为 TPN 传统可达性分析提供一种更为有效可行的方法。

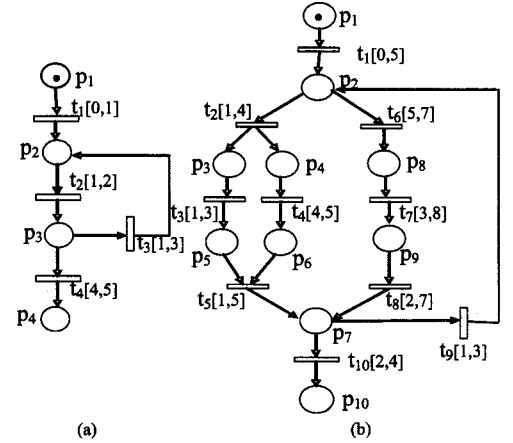


图 3 可调度性的综合分析

在图 3(a) 中, 源网中从 $M_0 = \{p_1\}$ 有无穷多个发生 (出现) 序列到达 $M_3 = \{p_4\}$ 。但任何一个序列可由 $(M_0 t_1 M_1 t_2 M_2) (\delta)^k (M_2 t_4 M_3)$ 构成, 其中 $\delta = (M_2 t_3 M_1 t_2 M_2)$, $M_1 = \{p_1\}$, $M_2 = \{p_2\}$, $k \in N$ 。根据定理 3 可知, $(M_0 t_1 M_1 t_2 M_2) (\delta)^{100} (M_2 t_4 M_3)$ 可调度当且仅当 $(M_0 t_1 M_1 t_2 M_2) \delta (M_2 t_4 M_3)$ 可调度。在这里, 因为 $(M_2 t_4 M_3)$ 不可调度, 所以任何一个都不可调度。为了判断 TPN 中 M_3 的可达性, 只需检查基本序列的可调度性, 即: $k=0$ 时的 $\delta_1 = (M_0 t_1 M_1 t_2 M_2 t_4 M_3)$ 和 $k=1$ 时的 $\delta_2 = (M_0 t_1 M_1 t_2 M_2 \delta M_2 t_4 M_3)$ 。由于时间约束, δ_1 和 δ_2 都不可调度, 因此 M_3 是不可达的。在图 3(b) 中, 我们也可鉴别源网中到达 $\{p_{10}\}$ 的一些

关键变迁序列,如, $(t_2 t_3 t_4 t_5)$, $(t_2 t_4 t_3 t_5)$, $(t_6 t_7 t_8)$, $(t_9 t_2 t_3 t_4 t_5)$, $(t_9 t_6 t_7 t_8)$, (t_1) 和 (t_{10}) 。由于 $(t_2 t_4 t_3 t_5)$ 和 $(t_6 t_7 t_8)$ 从 $M_1 = \{p_2\}$ 是不可调度的,任何包含 $(t_2 t_4 t_3 t_5)$ 和 $(t_6 t_7 t_8)$ 的序列都不可调度。然而, $t_1 (t_2 t_3 t_4 t_5) (t_9 t_2 t_3 t_4 t_5)^k (t_9 t_2 t_3 t_4 t_5)^k (t_{10})$ 是可调度的,其中 $k \in N$ 。因此, $\{p_{10}\}$ 在 TPN 中是可达的。注意,只有序列可被分解(即在分解标识下无并发性)时,定理 1—定理 3 才有用。

6 车间装配系统的可调度性分析

本节描述如何通过把其时间性能同功能性能分开来分析柔性制造系统(FMS)中的车间装配子系统的可调度性。

FMS 系统是由一些计算机控制工具、自动化材料处理、装配和存贮系统组成的一个实时系统,在主计算机的控制下有机结合,在加工自动化的基础上实现物料流和信息流的自动化^[18]。这些系统中,高性能和灵活性的不断增长的需要以及并发、最后期限驱动的活动的连锁因素和实时决策等都对 FMS 的设计,特别是在控制和调度方面的设计,提出了重大挑战。

下面考虑 FMS 中的一个车间装配系统。如图 4 所示,装配系统由 3 个工序、一个质检员、一个装配工和两个拆卸工组成。系统接收两类零件(A 和 B)作为输入,经过对输入零件进行处理,一个 A 零件和一个 B 零件被装配成一个最后产品,装配过程如下:原始零件(毛坯)成对到达, A 零件经由工序 1 和工序 2 顺序处理, B 零件经由工序 3 处理。处理后的 A 零件和 B 零件最后由装配工装配,质检员负责已装配产品的质量检查。产品如果满足质量要求,则作为最终产品下线;否则,它由拆卸工 1 或拆卸工 2(依赖于其状态)拆卸。拆卸工 1 产生一个 A 零件分别送回工序 1、2,产生一个 B 零件送回装配工。拆卸工 2 产生 A 零件分别送回工序 1、2,产生一个 B 零件送回工序 3。对所有的工序,质检员、装配工和拆卸工在其上都有一定的时间限制(见表 1),考虑到时间约束,我们需要检查一个装配调度(计划)、一个 A 零件和一个 B 零件能否在给定时间内被装配,这主要依赖于下列情况的分析:

- (1) A 零件和 B 零件都没有质量问题;
- (2) B 零件没问题,但 A 零件 m 次未通过质检($m > 0$);
- (3) A 零件和 B 零件分别有 m 和 n 次未通过质检($m, n > 0$)。

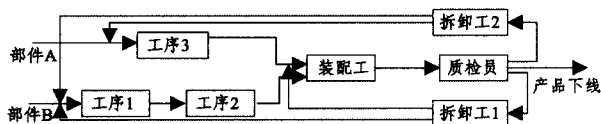


图 4 车间装配系统的结构

装配系统的 TPN 模型如图 5 所示,库所和变迁在表 1 中给出其描述。

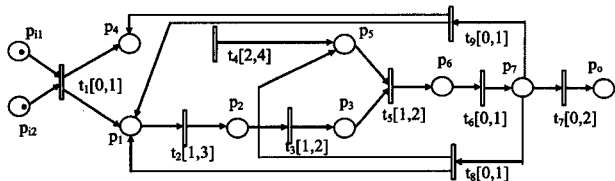


图 5 车间装配系统的 TPN 模型

表 1 图 5 的说明

库所	描述
P_{11}, P_{12}	来自其它提供 A 零件和 B 零件的子系统的输入
P_0	到其它存储装配产品的子系统的输出
P_1	A 零件
P_2	经过工序 1 处理后的 A 零件
P_3	经过工序 2 处理后准备装配的 A 零件
P_4	B 零件
P_5	经过工序 3 处理后准备装配的 B 零件
P_6	装配好的产品
P_7	质检员的检查结果

变迁	描述	时间区间
t_1	接受 A 零件和 B 零件	$[0, 1]$
t_2	工序 1 处理 A 零件	$[1, 3]$
t_3	工序 2 处理 A 零件	$[1, 2]$
t_4	工序 3 处理 B 零件	$[2, 4]$
t_5	装配工作	$[1, 2]$
t_6	质检员检查装配的产品	$[0, 1]$
t_7	最终产品的下线	$[0, 2]$
t_8	拆卸工 1 工作	$[0, 1]$
t_9	拆卸工 2 工作	$[0, 1]$

假设装配系统在某个时刻收到 A 零件和 B 零件,即 $M_0 = \{p_{11}, p_{12}\}$ 。不考虑时间约束,上述情况的功能需求很容易分析。

例如: $\delta_1 = (t_1 t_2 t_3 t_4 t_5 t_6 t_7)$, $\delta_2 = (t_1 t_2 t_3 t_4 t_5 t_6 t_8 t_2 t_3 t_5 t_6 t_7)$ 和 $\delta_3 = (t_1 t_2 t_3 t_4 t_5 t_6 t_9 t_2 t_3 t_4 t_5 t_6 t_7)$ 是源网中模拟对应装配过程功能行为的 3 个基本出现序列。

根据第 3 节中的算法, δ_1 的可调度性可如下分析:

- 0) $TS_0 = [0, 0]$; $//M_0 = \{p_{11}, p_{12}\}$
 $D_0 = \{t_1[0, 1]\}$; $AD_0 = \{t_1[0, 1]\}$
- 1) t_1 在 $[0, 1]$ 内可调度; $//M_1 = \{p_1, p_4\}$
 $TS_1 = [0, 1]$; $D_1 = \{t_2[1, 3], t_4[2, 4]\}$;
 $AD_1 = \{t_2[1, 4], t_4[2, 5]\}$;
- 2) t_2 在 $[1, 3]$ 内可调度; $//M_2 = \{p_2, p_4\}$
 $TS_2 = [1, 4]$; $D_2 = \{t_3[1, 2], t_4[0, 3]\}$;
 $AD_2 = \{t_3[2, 6], t_4[2, 5]\}$
- 3) t_3 在 $[1, 2]$ 内可调度; $//M_3 = \{p_3, p_4\}$
 $TS_3 = [2, 5]$; $D_3 = \{t_4[0, 2]\}$;
 $AD_3 = \{t_4[2, 5]\}$
- 4) t_4 在 $[0, 2]$ 内可调度; $//M_4 = \{p_3, p_5\}$
 $TS_4 = [2, 5]$; $D_4 = \{t_5[1, 2]\}$;
 $AD_4 = \{t_5[3, 7]\}$
- 5) t_5 在 $[1, 2]$ 内可调度; $//M_5 = \{p_6\}$
 $TS_5 = [3, 7]$; $D_5 = \{t_6[0, 1]\}$;
 $AD_5 = \{t_6[3, 8]\}$
- 6) t_6 在 $[0, 1]$ 内可调度; $//M_6 = \{p_7\}$
 $TS_6 = [3, 8]$; $D_6 = \{t_7[0, 2], t_8[0, 1], t_9[0, 1]\}$;
 $D_6 = \{t_7[3, 10], t_8[3, 9], t_9[3, 9]\}$
- 7) t_7 在 $[0, 1]$ 内可调度; $//M_7 = \{p_0\}$
 $TS_7 = [3, 9]$; $D_7 = \emptyset$; $AD_7 = \emptyset$

因此 δ_1 是可调度的, $\psi(\delta_1) = [3, 9]$ 。

类似地, δ_2 和 δ_3 也都可调度, 且 $\psi(\delta_2) = [6, 18]$, $\psi(\delta_3) = [6, 17]$ 。

除了分析基本序列,也需要处理并发和回路,在这里, t_4 和 $t_2 t_3$ 是并发的。 $(t_2 t_3 t_4)$, $(t_2 t_4 t_3)$ 和 $(t_4 t_2 t_3)$ 都是从标识 $M_1 = \{p_1, p_4\}$ 可调度的,在上述 δ_1 , δ_2 和 δ_3 中,将 $(t_2 t_3 t_4)$ 用

$(t_2 t_4 t_3)$ 替换可得到 3 个序列。

例如,如果把 δ_1 中的 $(t_2 t_3 t_4)$ 用 $(t_4 t_2 t_3)$ 替换得 δ_4 , 则 δ_4 的分析可把上述的 2), 3), 4) 步改为:

2) t_4 在 $[2, 3]$ 内可调度; $//M_2 = \{p_1, p_5\}$

$TS_2 = [2, 4]; D_2 = \{t_2[1, 0]\};$

$AD_2 = \{t_2[2, 4]\}$

3) t_2 在 $[0, 1]$ 内可调度; $//M_3 = \{p_2, p_5\}$

$TS_3 = [2, 4]; D_3 = \{t_3[1, 2]\};$

$AD_3 = \{t_3[3, 6]\}$

4) t_3 在 $[1, 2]$ 内可调度; $//M_4 = \{p_3, p_5\}$

$TS_4 = [3, 6]; D_4 = \{t_5[1, 2]\};$

$AD_4 = \{t_5[4, 8]\}$

实际上第 5 步以后也不一样,但其 D_i 相同。

如果 δ_1 中 $(t_2 t_3 t_4)$ 替换为 $(t_2 t_4 t_3)$, 2), 3), 4) 步可改为:

2) t_2 在 $[1, 3]$ 内可调度; $//M_2 = \{p_2, p_4\}$

$TS_2 = [1, 4]; D_2 = \{t_3[1, 2], t_4[0, 3]\};$

$AD_2 = \{t_3[2, 6], t_4[2, 5]\}$

3) t_4 在 $[0, 2]$ 内可调度; $//M_3 = \{p_2, p_5\}$

$TS_3 = [2, 5]; D_3 = \{t_3[0, 2]\};$

$AD_3 = \{t_3[2, 6]\}$

4) t_3 在 $[0, 2]$ 内可调度; $//M_4 = \{p_3, p_5\}$

$TS_4 = [2, 6]; D_4 = \{t_5[1, 2]\};$

$AD_4 = \{t_5[3, 8]\}$

$\psi(t_2 t_3 t_4) = [2, 5] - [0, 1] = [2, 4],$

$\psi(t_2 t_4 t_3) = [2, 6] - [0, 1] = [2, 5],$

$\psi(t_4 t_2 t_3) = [3, 6] - [0, 1] = [3, 5].$

应该考虑的回路由是 $\sigma_1 = (t_8 t_2 t_3 t_4 t_5 t_6)$ 和 $\sigma_2 = (t_9 t_5 t_6)$, 其中, $\sigma = (t_2 t_3 t_4)$ 或 $(t_2 t_4 t_3)$ 或 $(t_4 t_2 t_3)$, 易证, $\psi(\sigma_1) = [3, 10]$, $\psi(\sigma_2) = \varphi(\sigma) + [1, 3]$ 。

回顾一下装配调度, 它们实际上对应着下面的任务序列:

① $(t_1 \sigma t_5 t_6 t_7)$

② $(t_1 \sigma t_5 t_6 \sigma_1^m t_7)$

③ $(t_1 \sigma t_5 t_6 \sigma_1^m \sigma_2^n t_7)$

其中, $m > 0, n > 0$ 。这些序列很明显是源网中的出现序列, 它们的时间特性根据上述讨论很容易计算。注意到 σ_1 和 σ_2 都是可自合成的, 且 σ_2 和 σ_1 可合成, 开始和结束标识都是 $\{p_7\}$ 。 $\sigma_1 \sigma_2 = \sigma_2 \sigma_1$ (即其顺序并不重要)。因此, 以任何顺序 A 零件失败 m 次, B 零件失败 n 次都可表示为 $\sigma_1^m \sigma_2^n$ 。

结束语 本文给出了用时间 Petri 网建模的实时系统的可调度分析的一种方法。本方法通过把时间特性同其他行为分离, 使用相对和绝对时间模式来判定各个变迁的可调度性和计算任务执行的时间跨度。并给出了把一个复杂的任务分解为若干个子序列的合成技巧, 以降低可调度分析的复杂度。下一步的工作是设计一个软件工具实现, 以便进行验证。已知 Petri 网的可达性和有界性问题是可判定的^[16], 但对 TPN 是不可判定的^[18]。即使一个标识在传统 Petri 网中可达, 其在对应 TPN 中也可能不可达。下一步我们试图在 TPN 的可达性和对应传统 Petri 网的可达性之间建立一些关系, 以便较容易地对某些 TPN 进行可达性分析。

参考文献

- [1] Felder M, Mandrioli D, Morzenti A. Proving properties of real-time systems through logical specifications and Petri net models [J]. IEEE Transactions on Software Engineering, 1994, 20(2): 127-141
- [2] Haban D, Shin K G. Application of real-time monitoring to scheduling tasks with random execution times [J]. IEEE Transactions on Software Engineering, 1990, 16(12): 1374-1389
- [3] Tsai J J P, Jennhwa Yang S, Chang Y H. Timing constraint Petri nets and their application to schedulability analysis of real-time system specifications [J]. IEEE Transactions on Software Engineering, 1995, 21(1): 32-49
- [4] Merlin P M, Farber D J. Recoverability of communication protocols — implications of a theoretical study [J]. IEEE Transaction on Communications, 1976, 24(9): 1036-1049
- [5] Wang Jia-cun, Deng Yi, Xu Gang. Reachability analysis of real-time systems using time Petri nets [J]. IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics, 2000, 30(5): 725-736
- [6] Berthomieu B, Lime D, Roux O H, et al. Reachability problems and abstract state spaces for time Petri nets with stopwatches [J]. Discrete Event Dynamic Systems, 2007, 17(2): 133-158
- [7] Stoyenko A D, Hamacher C, Holt R C. Analyzing hard-real-time programs for guaranteed schedulability [J]. IEEE Transaction on Software Engineering, 1991, 17(8): 737-750
- [8] Jahanian F, Mok A K-L. Safety analysis of timing properties in real-time systems [J]. IEEE Transaction on Software Engineering, 1986, 12(12): 890-904
- [9] Berthomieu B, Diaz M. Modeling and verification of time dependent systems using time Petri nets [J]. IEEE Transaction on Software Engineering, 1991, 17(3): 259-273
- [10] Bucci G, Vicario E. Compositional validation of time-critical systems using communicating time Petri nets [J]. IEEE Transaction on Software Engineering, 1995, 21(12): 969-992
- [11] Furfaro A, Nigro L. Modelling and schedulability analysis of real-time sequence patterns using time Petri nets and Uppaal [C] // Proceedings of 2007 International Workshop on Real Time Software. 2007, 16: 821-835
- [12] <http://www.uppaal.org/>
- [13] Gu Z, Shin K G. Analysis of event-driven real-time systems with time Petri nets: a translation-based approach [C] // Design and Analysis of Distributed Embedded Systems DAPES '02. 2002: 31-40
- [14] 蒋昌俊. Petri 网的行为理论及其应用 [M]. 北京: 高等教育出版社, 2003
- [15] 袁崇义. Petri 网原理及应用 [M]. 北京: 电子工业出版社, 2005
- [16] 吴哲辉. Petri 网导论 [M]. 北京: 机械工业出版社, 2006
- [17] Popova-Zeugmann L. On time Petri nets [J]. Elektronische Informationsverarbeitung und Kybernetik, 1991, 27(4): 227-244
- [18] Wang Jia-cun, Deng Yi. Incremental modeling and verification of flexible manufacturing systems [J]. Journal of Intelligent Manufacturing, 1999, 10(6): 485-502