基于 Petri 网局部性的极大冲突集枚举算法

潘 理[1],郑 红[2],刘显明[3],杨 勃[1]
(1. 湖南理工学院信息与通信工程学院,湖南岳阳 414006;2. 华东理工大学信息科学与工程学院,上海 200237;
3. 江西省电力公司信息通信分公司,江西南昌 330077)

摘 要: 冲突是 Petri 网研究的重要主题. 目前 Petri 网冲突研究主要集中于冲突建模和冲突消解策略,而对冲突问题本身的计算复杂性却很少关注. 提出 Petri 网的冲突集问题,并证明冲突集问题是 NP(Non-deterministic Polynomial)完全的. 提出极大冲突集动态枚举算法,该算法基于当前标识的所有极大冲突集,利用 Petri 网实施局部性,仅计算下一标识中受局部性影响的极大冲突集,从而避免重新枚举所有极大冲突集. 该算法时间复杂度为 $O(m^2n)$,m 是当前标识的极大冲突集数目,n 是变迁数. 最后证明自由选择网、非对称选择网的极大冲突集枚举算法复杂度可降至 $O(n^2)$. 极大冲突集枚举算法研究将为 Petri 网冲突问题的算法求解提供理论参考.

关键词: Petri 网; 冲突集问题; NP(Non-deterministic Polynomial)完全性; 极大冲突集枚举算法
中图分类号: TP301 **文献标识码:** A **文章编号:** 0372-2112 (2016)08-1858-06
电子学报 URL: http://www.ejournal.org.cn **DOI:** 10.3969/j.issn.0372-2112.2016.08.013

Maximal Conflict Set Enumeration Algorithm Based on Locality of Petri Nets

PAN Li[1],ZHENG Hong[2],LIU Xian-ming[3],YANG Bo[1]
(1. *School of Information and Communication Engineering*,*Hunan Institute of Science and Technology*,*Yueyang*,*Hunan* 414006,*China*;
2. *Information Science and Engineering College*,*East China University of Science and Technology*,*Shanghai*,200237 *China*;
3. *Information and Communication Branch*,*Jiangxi Electric Power Company*,*Nanchang*,*Jiangxi* 330077,*China*)

Abstract: Conflict is an essential concept in Petri net theory. The existing research focuses on the modelling and resolution strategies of conflict problems,but less on the computational complexity of the problems theirselves. In this paper,we propose the conflict set problem for Petri nets,and prove that the conflict set problem is NP-complete. Furthermore,we present a dynamic algorithm for the maximal conflict set enumeration. Our algorithm only computes those conflict sets that are affected by local firing,which avoids enumerating all maximal conflict sets at each marking. The algorithm needs time O(m2n) where m is the number of maximal conflict sets at the current marking and n is the number of transitions. Finally,we show that the maximal conflict set enumeration problem can be solved in O(n2) for free-choice nets and asymmetric choice nets. The results on complexity of thel conflict set problem provide a theoretical reference for solving conflict problems of Petri nets.

Key words: Petri nets;conflict set problem;NP completeness;maximal conflict set enumeration algorithm

1 引言

冲突是 Petri 网理论的重要概念. 冲突的本质是资源竞争[1]. 当多个冲突变迁竞争有限资源,若其中一个获得资源,其它与之冲突的变迁就会丧失使能. 若变迁集合中的任意两个变迁都是冲突的,称这个集合为冲突集. 现有 Petri 网冲突研究主要集中于冲突处理策略和冲突问题建模. 经典 Petri 网采用不确定选择策略处理变迁冲突[2];优先级 Petri 网通过设置优先级来解决冲突[3];时间 Petri 网则利用时间竞争来处理变迁之间的冲突[4];谓词/变迁网通过谓词来控制冲突变迁的实施[5];随机 Petri 网则引入某种实施概率分布来处理变迁间的冲突[6,7]. 在建模方面,近期 Zeng 等利用 Petri 网方法探讨应急响应过程中资源冲突侦测和消解[8];Tian 等利用时间 Petri 网的时间约束特性处理实时系统的冲突问题[9];Popescu 等利用 Petri 网方法解决面向服务制造系统调度过程中的资源冲突[10].

收稿日期:2015-05-12;修回日期:2015-11-09;责任编辑:马兰英
基金项目:国家自然科学基金(No. 61473118);湖南省教育厅科学研究重点项目(No. 15A079);湖南省科技计划项目(No. 2014GK3026);江西省电力公司科技项目(No. 5218351400A1)

上述研究注重冲突问题建模及冲突消解策略,而对冲突问题本身的计算复杂性却很少关注. 现实系统中的冲突情况往往非常复杂,枚举系统的极大冲突集,对于全面了解系统冲突、寻求冲突消解方案是非常重要的. 因此,对冲突集问题复杂度进行界定,并提出基于 Petri 网特征的枚举算法,将为 Petri 网冲突问题的计算求解提供一种探索途径. 目前已有大量文献研究 Petri 网的死锁问题、活性问题、可达性问题、合法实施序列问题、合成问题、步问题等,得出了一系列有价值的计算复杂性结论[11-16]. 但这些结论并没涉及冲突集问题. 本文尝试以冲突集问题为研究对象,通过从已知 NP 完全问题归约到冲突集问题,证明冲突集问题的 NP 完全性;然后利用 Petri 网实施的局部性特征,提出动态枚举算法,降低极大冲突集枚举的计算成本,为冲突集问题的复杂性研究提供有益参考.

2　冲突集问题

2.1　基本定义

用 N 表示自然数集 $\{0,1,2,\cdots\}$,用 Z^+ 表示正整数集.

定义 1[1]　一个 Petri 网是一个四元组 $\boldsymbol{\Sigma}=(P,T,F,M_0)$,其中:

(1)P 是库所集;(2)T 是变迁集,且 $P\cap T=\varnothing$;(3)$F\subseteq(P\times T)\cup(T\times P)$是流关系;(4)$\boldsymbol{M}_0:P\to N$ 是初始标识.

标识 $\boldsymbol{M}$ 可表示为一个$|P|$元向量,第 i 个元素表示库所 p_i 中的令牌数. 流关系 F 可表示为特征函数的形式,即 $F:(P\times T)\cup(T\times P)\to\{0,1\}$. 用 ${}^{\cdot}p=\{t\,|\,(t,p)\in F\}$ 和 $p^{\cdot}=\{t\,|\,(p,t)\in F\}$ 表示库所 p 的输入变迁集和输出变迁集;类似的,用 ${}^{\cdot}t$ 和 $t^{\cdot}$ 表示变迁 t 的输入库所集和输出库所集.

定义 2[1]　一个变迁 $t\in T$ 在标识 M 是使能的,当且仅当 $\forall p\in P,F(p,t)\leqslant\boldsymbol{M}(p)$. 用 $En(\boldsymbol{M})$ 表示标识 $\boldsymbol{M}$ 下所有使能变迁的集合.

定义 3[1]　如果 t_f 在标识 M 是使能的,那么 t_f 可以实施并产生一个新的后继标识 M',且 $\forall p\in P,M'(p)=M(p)-F(p,t_f)+F(t_f,p)$.

定义 4[1]　给定 $t_1,t_2\in En(\boldsymbol{M})$,若 $\forall p\in P$,使 $F(p,t_1)+F(p,t_2)>M(p)$,则称变迁 t_1 和 t_2 在标识 M 是(有效)冲突的,用 $t_1\Uparrow_M t_2$ 表示.

定义 5　给定一个变迁集 $U\subseteq T$,如果 $\forall t_1,t_2\in U$,有 $t_1\Uparrow_M t_2$,则 U 是标识 M 的一个冲突集. 若不存在其它冲突集 U',使 $U\subset U'$,则称 U 是 $\boldsymbol{M}$ 的极大冲突集. 用 $MCS(\boldsymbol{M})$表示在标识 M 下的所有极大冲突集的集合.

2.2　冲突集问题

定义 6[17]　复杂类 P 表示用确定型图灵机在多项式时间内可解决的判定问题集,NP 表示用非确定型图灵机在多项式时间内可解决的判定问题集.

定义 7[17]　若判定问题 A 属于 NP,且所有其它 NP 问题都能多项式时间多一归约到 A,则称 A 是 NP 完全的.

引理 1[17]　对于判定问题 A,若存在 NP 完全问题 B,使 B 多项式时间多一归约到 A,则 A 是 NP 困难的. 若 A 是 NP 困难的且 $A\in$ NP,则 A 是 NP 完全的.

定义 8　(冲突集问题,CS)给定 Petri 网 $\boldsymbol{\Sigma}=(P,T,F,\boldsymbol{M}_0)$,标识 $\boldsymbol{M}\in\mathrm{RS}(\boldsymbol{M}_0)$和正整数 $k\leqslant|T|$,问在标识 $\boldsymbol{M}$ 是否包含变迁数至少为 k 的冲突集?

接下来,通过从已知 NP 完全问题(团问题)归约到冲突集问题,来证明冲突集问题的 NP 完全性.

设 $G=(V,E)$是简单无向图,这里 V 是顶点集,E 是边集. 给定一个顶点子集 $A\subseteq V$,用 $G(A)$表示由 A 诱导的子图.

定义 9[18]　给定图 $G=(V,E)$,如果 $\forall u,w\in V$,有$(u,w)\in E$,则称 G 是完全的. 若 $G(Q)$是完全图,则称 Q 是 G 的一个团.

定义 10[18]　(团问题,CLIQUE)给定简单无向图 $G=(V,E)$和正整数 $k\leqslant|V|$,问 G 是否包含基数至少为 k 的团?

引理 2[18]　团问题是 NP 完全的.

定理 1　冲突集问题是 NP 完全的.

证明　(1)首先证明冲突集问题是 NP 困难的.

考虑从团问题多项式时间多一归约到冲突集问题. 设简单无向图 $G=(V,E)$和正整数 s 是团问题的一个实例,要构造冲突集问题的一个实例:Petri 网 $\boldsymbol{\Sigma}$、标识 $\boldsymbol{M}$ 和正整数 k,使 $\boldsymbol{\Sigma}$ 在标识 $\boldsymbol{M}$ 有一个大小至少为 k 的冲突集当且仅当 G 有一个大小至少为 s 的团. 按照下面的方式进行构造:

(ⅰ)$v_i\in V$ 当且仅当 $t_i\in T$,即 Petri 网 $\boldsymbol{\Sigma}$ 中的每个变迁 $t_1,t_2,\cdots,t_n$ 对应图 G 中的每个顶点 $v_1,v_2,\cdots,v_n$.

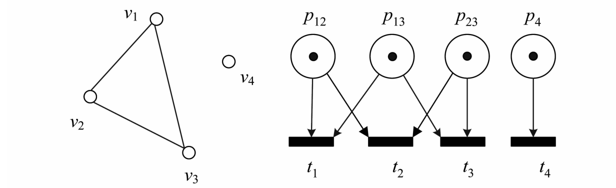
(ⅱ)$(v_i,v_j)\in E$ 当且仅当 $p_{ij}\in P\wedge\boldsymbol{M}(p_{ij})=1\wedge F(p_{ij},t_i)=1\wedge F(p_{ij},t_j)=1$. 换句话说,两顶点 v_i 和 v_j 相邻当且仅当 t_i 与 t_j 共享输入库所 p_{ij},库所 p_{ij} 中放置 1 个令牌,且弧(p_{ij},t_i)和(p_{ij},t_j)的权值均为 1.

(ⅲ)v_i 是孤立点当且仅当 $p_i\in P\wedge M(p_i)=1\wedge F(p_i,t_i)=1$. 即 v_i 是孤立点当且仅当 t_i 有一个输入库所 p_i,库所 p_i 中放置 1 个令牌,且弧(p_i,t_i)的权值为 1.

(ⅳ)$k=s$.

从上面构造可以看出,Petri 网 $\boldsymbol{\Sigma}$ 的每个变迁对应图 G 的每个顶点,且每个变迁在 $\boldsymbol{M}$ 都使能;两个变迁共享一个库所当且仅当图 G 中对应的两个顶点是相邻的. 如图 1(a)所示,图 G 有 4 个顶点 v_1,v_2,v_3,v_4,其中包含一个 $s=3$ 的团$\{v_1,v_2,v_3\}$. 通过上述构造得到 Petri 网 $\boldsymbol{\Sigma}$ 和标识 M,如图 1

(b)所示,Σ 有 4 个变迁 t_1,t_2,t_3,t_4,4 个库所 p_{12},p_{13},p_{23},p_4,每个库所中都有 1 个令牌,4 个变迁均使能. 容易看出,Petri 网 Σ 在 M 包含一个 $k=3$ 的冲突集$\{t_1,t_2,t_3\}$.



(a) 图G (b) Petri 网Σ

图1 从图G到Petri网Σ的构造

现在证明 Petri 网 Σ 在标识 $\boldsymbol{M}$ 有一个大小至少为 k 的冲突集当且仅当 G 有一个大小至少为 s 的团. Q 是 G 的一个团且 $|Q|\leqslant s$,当且仅当 $\forall v_i,v_j\in Q,(v_i,v_j)\in E$;由上面构造步骤,当且仅当得到对应的变迁集 $U,|U|=|Q|,U\subseteq En(M)$ 且 $\forall t_i,t_j\in U,F(p_{ij},t_i)+F(p_{ij},t_j)>\boldsymbol{M}(p_{ij})$;即 U 在标识 M 是一个冲突集且$|U|\leqslant k$.

进一步分析这是一个多项式时间的构造. 设$|V|=n$,则 V,E 和 k 的输入长度分别为 $O(n)$,$O(n^2)$ 和 $O(n)$,故实例的输入总长度是 $O(n^2)$. 构造步(ⅰ)根据顶点集 V 确定变迁集 T,故$|T|=n$;构造步(ⅱ)和(ⅲ)根据边集 E 确定库所集 P,故$|P|=O(n^2)$;标识 M 的规模是$|P|=O(n^2)$,流关系 F 的规模是$|P\parallel T|=O(n^3)$. 因此整个构造需时间 $O(n^3)$,即从 CLIQUE 能多项式时间归约到冲突集问题. 由于 CLIQUE 是 NP 完全的(引理 2),根据引理 1,冲突集问题是 NP 困难的.

(2)证明冲突集问题属于 NP.

对于 Petri 网 Σ,对任何标识 $\boldsymbol{M}\in RS(\boldsymbol{M}_0)$,给出一个验证冲突集问题的非确定型算法(见算法 1).

算法 1 冲突集问题的非确定算法

input: $(P,T,F,\boldsymbol{M}),k$
begin
  nondeterministically guess a transition set U, and verify that:
    (i) $|U|\geqslant k$
    (ii) $\forall t_i,t_j\in U$: $t_i \Uparrow_{\boldsymbol{M}} t_j$
  if all these conditions hold
    accept
  end if
end

假设$|P|=m,|T|=n$. 则 $P,T,F,\boldsymbol{M}$ 和 k 的长度分别是 $O(m)$,$O(n)$,$O(mn)$,$O(m)$和 $O(n)$,故输入的总长度是 $O(mn)$. U 的长度是 $O(n)$,验证(ⅰ)需时间 $O(n)$,验证(ⅱ)需时间 $O(mn^2)$,故验证(ⅰ)和(ⅱ)需时间 $O(mn^2)$. 这是一个能在多项式时间内验证冲突集问题的非确定型算法. 因此冲突集问题属于 NP.

综合(1)和(2)的证明,根据引理 1,冲突集问题是 NP 完全的. 证毕.

3 极大冲突集枚举算法

3.1 极大冲突集枚举问题

定义 11[17] 复杂类 FP 表示用确定型图灵机在多项式时间可解决的函数问题集.

定义 12 (极大冲突集枚举问题)给定 Petri 网 $\Sigma=(P,T,F,\boldsymbol{M}_0)$和标识 $\boldsymbol{M}\in RS(\boldsymbol{M}_0)$,求 Petri 网 Σ 在标识 M 的所有极大冲突集.

Petri 网的极大冲突集枚举问题,可以使用已有的极大团枚举算法来求解. 不幸的是,极大团枚举问题是一个已知的 NP 困难问题. 求解这个问题最有效的算法是由 Bron 和 Kerbosch 提出的分支限界算法[19],文献[20]证明这种算法在最坏情况下的时间复杂度为 $O(n^{n/3})$. 幸运的是,Petri 网的实施具有局部性特征. 每一次变迁实施仅仅改变与该变迁相关的前集和后集. 因此,每一次变迁实施之后,并不需要重新生成所有的极大冲突集,而只需考虑受实施变迁局部环境影响的那些极大冲突集,这无疑将降低极大冲突集枚举问题的计算复杂度.

3.2 极大冲突集枚举算法

接下来讨论极大冲突集枚举问题的算法求解. 为了简化算法描述,我们假设 Petri 网是安全的. 用 $MCS(\boldsymbol{M},t_i)$表示在标识 $\boldsymbol{M}$ 下包含变迁 t_i 的极大冲突集的集合. 进一步,用 $MCS(\boldsymbol{M},S)$表示包含 S 中所有变迁的极大冲突集的集合,即这些极大冲突集均包含变迁子集 S.

Petri 网变迁实施可分为两步,第一步根据流关系,从变迁的输入库所中取出相应数量的令牌;第二步根据流关系,将相应数量的令牌放置到变迁的输出库所. 第一步可能使某些原来冲突的变迁变为不冲突,第二步可能新增一些冲突关系. 下面根据影响冲突关系变化的两个步骤,定义两个相应的基本操作:

(1)操作 del$(\boldsymbol{M},t)$:执行 $\boldsymbol{M}-F(\cdot,t)$,删除不再冲突的关系;(2)操作 add$(\boldsymbol{M},t)$:执行 $\boldsymbol{M}-F(\cdot,t)+F(t,\cdot)$,添加新的冲突关系.

用 $Dis(\boldsymbol{M},t)=En(\boldsymbol{M})\setminus En(M-F(\cdot,t))$在标识 M 实施 t 后丧失使能的变迁集. 操作 del$(\boldsymbol{M},t)$如算法 2 所示. 对标识 M 下的每个极大冲突集 c,在标识 $\boldsymbol{M}-F(\cdot,t)$,若 c 包含非使能变迁,则删除非使能变迁. 再判断 c 是否包含在其它极大冲突集中,若包含,则说明 c 不是标识 $\boldsymbol{M}-F(\cdot,t)$的极大冲突集,删除 c.

算法 2 操作 del$(\boldsymbol{M},t)$的算法

input: $MCS(\boldsymbol{M}),t$
begin
  for each $c\in MCS(M)$

```
    if c∩Dis(M,t) ≠ ∅
      c←c\Dis(M,t)
      if|MCS(M,c)| > 1
          MCS(M)←MCS(M)\{c}
      endif
    endif
  endfor
  return MCS(M)
end
```

设$|T| = n$,$|MCS(\boldsymbol{M})| = m$. 变迁 t 的实施通常只会影响少量极大冲突集,但最坏情况下,可能涉及绝大多数极大冲突集. 因此,求解 $MCS(\boldsymbol{M},c)$ 通常需 $O(n)$,但最坏情况下需 $O(mn)$. 若循环 m 次,则该操作的时间复杂度通常为 $O(mn)$,最坏情况下为 $O(m^2n)$.

用 $Newly(\boldsymbol{M},t) = En(\boldsymbol{M}') \backslash En(\boldsymbol{M} - F(\cdot,t))$ 表示在新标识 $\boldsymbol{M}' = \boldsymbol{M} - F(\cdot,t) + F(t,\cdot)$ 新使能的变迁集合. 操作 add($\boldsymbol{M},t$)如算法 3 所示. 对每个新使能的变迁 t_i,若 t_i 不与任何极大冲突集相冲突,则 t_i 独自构成极大冲突集;若 t_i 与某个极大冲突集 c 的所有变迁冲突,则 t_i 并入 c;若 t_i 与极大冲突集 c 的部分变迁冲突,且这部分变迁不是其它极大冲突集的子集,则 t_i 与部分变迁构成新的极大冲突集.

算法 3　操作 add($\boldsymbol{M},t$)的算法

```
input:MCS(M),t
begin
for each tᵢ ∈ Newly(M,t)
  let C = ∪_{c∈MCS(M)} c, S = ∪_{t∈C} ˙t
  if ˙tᵢ∩S = ∅
      MCS(M)←MCS(M)∪{tᵢ}
  else
      for each c ∈ MCS(M)
        let S₁ = ∩_{t∈c} ˙t, S₂ = ∪_{t∈c} ˙t
        if ˙tᵢ∩S₁ ≠ ∅
            c←c∪{tᵢ}
        elseif tᵢ∩S₂ ≠ ∅
            let w = ∪_{t∈c}(t ⇑_M tᵢ)
            if |MCS(M,w)| = 1
              c' = w∪{tᵢ}
              MCS(M)←MCS(M)∪{c'}
            endif
        endif
      endfor
  endif
endfor
return MCS(M)
end
```

设$|T| = n$,$|MCS(\boldsymbol{M})| = m$,$|Newly(\boldsymbol{M},t)| = k$. 外层 for 循环 k 次,内层 for 循环 m 次,内层循环体通常需 $O(n)$;最坏情况下变迁实施影响多数极大冲突集,则需 $O(mn)$. 故整个算法通常情况需 $O(kmn)$,最坏情况下需 $O(km^2n)$. 极大冲突集动态枚举算法 enum($\boldsymbol{M},t$)如算法 4 所示,算法思路如下:若在标识 $\boldsymbol{M}$ 实施变迁 t,首先使用 del($\boldsymbol{M},t$)从所有极大冲突集中删除丧失使能的变迁,计算得到标识 $\boldsymbol{M} - F(\cdot,t)$ 的极大冲突集;然后在标识 $\boldsymbol{M} - F(\cdot,t) + F(t,\cdot)$,对所有新增使能变迁,使用 add($\boldsymbol{M},t$)添加新的冲突关系,得到新标识下所有极大冲突集.

算法 4　极大冲突集动态枚举算法 enum($\boldsymbol{M},t$)

```
input:MCS(M),t
begin
    Calculate disabled transition set Dis(M,t) at marking M − F(·,t)
    del(M,t)
    Calculate newly enabled transition set Newly(M,t) at marking M' =
    M − F(·,t) + F(t,·)
    add(M,t)
    return MCS(M')
end
```

由于新使能变迁数 k 通常远小于变迁数 n,根据 del 和 add 操作的复杂度分析,极大冲突集动态枚举算法的时间复杂度通常情况为 $O(mn)$,最坏情况下需 $O(m^2n)$. 下面举例说明极大冲突集动态枚举算法. 如图 2 所示.

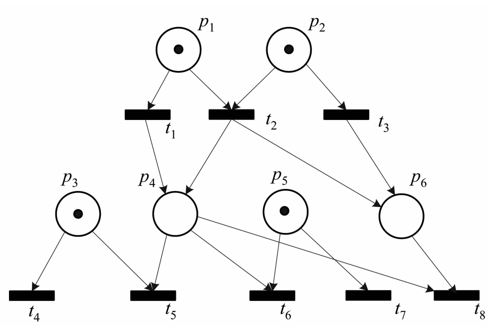


图2　极大冲突集的动态构造

在初始标识 $\boldsymbol{M}_0 = (1\ 1\ 1\ 0\ 1\ 0)$,$En(\boldsymbol{M}_0) = \{t_1,t_2,t_3,t_4,t_7\}$,使用 add($\boldsymbol{M}_0,\lambda$)得到 $MCS(\boldsymbol{M}_0) = \{\{t_1,t_2\},\{t_2,t_3\},\{t_4\},\{t_7\}\}$,这里 λ 是空变迁.

若在标识 $\boldsymbol{M}_0$ 实施 t_1,则先计算中间标识 $\boldsymbol{M}'_0 = \boldsymbol{M}_0 - F(\cdot,t_1) = (0\ 1\ 1\ 0\ 1\ 0)$,$Dis(\boldsymbol{M}_0,t_1) = \{t_1,t_2\}$,使用 del($\boldsymbol{M}_0,t_1$)得到 $MCS(\boldsymbol{M}'_0) = \{\{t_3\},\{t_4\},\{t_7\}\}$;计算新标识 $\boldsymbol{M}_1 = (0\ 1\ 1\ 0\ 1\ 0)$,$Newly(\boldsymbol{M}_0,t_1) = \{t_5,t_6\}$,使用 add($\boldsymbol{M}_0,t_1$)得到 $MCS(\boldsymbol{M}_1) = \{\{t_3\},\{t_4,t_5\},\{t_5,t_6\},\{t_6,t_7\}\}$. 若在标识 $\boldsymbol{M}_1$ 实施 t_3,则先计算中间标识 $\boldsymbol{M}'_1 = \boldsymbol{M}_1 - F(\cdot,t_3) = (0\ 0\ 1\ 1\ 1\ 0)$,$Dis(\boldsymbol{M}_1,t_3) =$

$\{t_3\}$,使用 del($\boldsymbol{M}_1,t_3$)得到 $MCS(\boldsymbol{M}'_1)=\{\{t_4,t_5\},\{t_5,t_6\},\{t_6,t_7\}\}$;计算新标识 $\boldsymbol{M}_2=(0\ 0\ 1\ 1\ 1\ 1)$,$Newly(\boldsymbol{M}_1,t_3)=\{t_8\}$,使用 add($\boldsymbol{M}_1,t_3$)得到 $MCS(\boldsymbol{M}_2)=\{\{t_4,t_5\},\{t_5,t_6,t_8\},\{t_6,t_7\}\}$.

4 特殊子问题

下面介绍三种典型的特殊子网:自由选择网、扩展自由选择网和非对称选择网.

定义 13[2] 给定 Petri 网 $\boldsymbol{\Sigma}=(P,T,F,\boldsymbol{M}_0)$,且所有弧的权值为 1.

(1)若 $\forall p_1,p_2\in P:p_1^{\bullet}\cap p_2^{\bullet}\neq\varnothing\rightarrow|p_1^{\bullet}|=|p_2^{\bullet}|=1$,则称 $\boldsymbol{\Sigma}$ 是自由选择网(FC).

(2)若 $\forall p_1,p_2\in P:p_1^{\bullet}\cap p_2^{\bullet}\neq\varnothing\rightarrow p_1^{\bullet}=p_2^{\bullet}$,则称 $\boldsymbol{\Sigma}$ 是扩展自由选择网(EFC).

(3)若 $\forall p_1,p_2\in P:p_1^{\bullet}\cap p_2^{\bullet}\neq\varnothing\rightarrow p_1^{\bullet}\subseteq p_2^{\bullet}\vee p_2^{\bullet}\subseteq p_1^{\bullet}$,则称 $\boldsymbol{\Sigma}$ 是非对称选择网(AC).

已经证明,这三种子网具有包含关系:$FC\subseteq EFC\subseteq AC$[7]. 因此,从 AC 获得的结论,对 FC 和 EFC 仍是有效的. 定义变迁集 T 上的结构冲突关系 $R=\{(t_1,t_2)\in T^2\mid{}^{\bullet}t_1\cap{}^{\bullet}t_2\neq\varnothing\}$.

定理 2 对于非对称选择网,结构冲突关系 R 是等价关系.

证明 $\forall t\in T$,有 ${}^{\bullet}t\cap{}^{\bullet}t\neq\varnothing$,即 $(t,t)\in R$,故 R 是自反的.

$\forall t_1,t_2\in T$,若 $(t_1,t_2)\in R$,即 ${}^{\bullet}t_1\cap{}^{\bullet}t_2\neq\varnothing$,则 ${}^{\bullet}t_2\cap{}^{\bullet}t_1\neq\varnothing$,即 $(t_2,t_1)\in R$,故 R 是对称的.

$\forall t_1,t_2,t_3\in T$,若 $(t_1,t_2)\in R$ 且 $(t_2,t_3)\in R$,即 ${}^{\bullet}t_1\cap{}^{\bullet}t_2\neq\varnothing$ 且 ${}^{\bullet}t_2\cap{}^{\bullet}t_3\neq\varnothing$. 不妨设 $p_{12}\in{}^{\bullet}t_1\cap{}^{\bullet}t_2$ 和 $p_{23}\in{}^{\bullet}t_2\cap{}^{\bullet}t_3$,则有 $\{t_1,t_2\}\subseteq p_{12}^{\bullet}$ 和 $\{t_2,t_3\}\subseteq p_{23}^{\bullet}$,即 $t_2\in p_{12}^{\bullet}\cap p_{23}^{\bullet}\neq\varnothing$,因为 $\boldsymbol{\Sigma}$ 是 AC,故有 $p_{12}^{\bullet}\subseteq p_{23}^{\bullet}$ 或 $p_{23}^{\bullet}\subseteq p_{12}^{\bullet}$,即得 $\{t_1,t_2,t_3\}\subseteq p_{23}^{\bullet}$ 或 $\{t_1,t_2,t_3\}\subseteq p_{12}^{\bullet}$,于是 ${}^{\bullet}t_1\cap{}^{\bullet}t_3\neq\varnothing$,即 $(t_1,t_3)\in R$,所以 R 是传递的.

因此 R 是等价关系. 证毕.

用 $[t]=\{t(\mid t\in T\wedge(t,t')\in R\}$ 表示 t 关于 R 的等价类.

定理 3 若 t 是 AC 的一个变迁,则 $\exists p\in{}^{\bullet}t$,使 $p^{\bullet}=[t]$.

证明 根据 AC 的定义,$\forall p',P\in{}^{\bullet}t$,有 $p'^{\bullet}\subseteq p^{\bullet}$ 或 $p^{\bullet}\subseteq p'^{\bullet}$,故 $\subseteq$ 是一个全序关系. 由于 ${}^{\bullet}t$ 是有限集,故必有最大元,不妨设为 $p^{\bullet}$. 又 $\forall t_i\in[t]$,有 ${}^{\bullet}t\cap{}^{\bullet}t_i\neq\varnothing$,不妨设 $p_i\in{}^{\bullet}t\cap{}^{\bullet}t_i$,于是 $t\in p^{\bullet}\cap p_i^{\bullet}$,故 $p_i^{\bullet}\subseteq p^{\bullet}$ 或 $p^{\bullet}\subseteq p_i^{\bullet}$. 由于 $p^{\bullet}$ 是 ${}^{\bullet}t$ 上关于 $\subseteq$ 的最大元,即有 $p_i^{\bullet}\subseteq p^{\bullet}$,故得 $t_i\in p^{\bullet}$. 于是 $p^{\bullet}\supseteq[t]$.

假设 $\exists t_i\in p^{\bullet}$ 但 $t_i\notin[t]$,则有 $p\in{}^{\bullet}t\cap{}^{\bullet}t_i$,即 $(t,t_i)\in R$,于是有 $t_i\in[t]$,矛盾. 因此,$p^{\bullet}=[t]$. 证毕.

依据等价关系 R,可以将变迁集划分成互不相交的子集,每个子集就是一个等价类. 因此,每个变迁仅属于一个等价类.

定理 4 若 $t\in En(M)$,则 $({}^{\bullet}t)^{\bullet}\cap En(M)$ 是极大冲突集.

证明 由定理 3 可知,$({}^{\bullet}t)^{\bullet}$ 是 t 关于 R 的等价类 $[t]$,即等价类中的变迁相互冲突,因此 $({}^{\bullet}t)^{\bullet}\cap En(M)$ 是冲突集;又 $[t]$ 中变迁不属于任何其他等价类,因此 $({}^{\bullet}t)^{\bullet}\cap En(M)$ 是极大冲突集. 证毕.

定理 5 对于非对称选择网,求解极大冲突集枚举问题的时间复杂度为 $O(n^2)$,这里 $|T|=n$.

证明 非对称选择网的极大冲突集枚举算法如算法 5 所示. 首先算法初始化 $MCS(\boldsymbol{M})$ 为空集. 然后求出每个变迁的关于有效冲突关系 R 的等价类,每个等价类即为一个极大冲突集.

算法 5 非对称选择网的极大冲突集枚举算法

```
input:P,T,F,M
begin
   MCS(M) = ∅
   for each t_i ∈ En(M)
      U = (˙t_i)˙ ∩ En(M)
      MCS(M) = MCS(M) ∪ {U}
      En(M) = En(M) \ U
   endfor
   return MCS(M)
end
```

先证明算法的正确性.

(ⅰ)由于 $En(\boldsymbol{M})$ 是有限集,故 for 循环的次数是有限的. 它根据等价类产生极大冲突集,因此 for 循环执行完成后,极大冲突集的数目不超过 $|En(\boldsymbol{M})|$. 因此算法一定会终止.

(ⅱ)根据定理 4,对任何 $t_i\in En(\boldsymbol{M})$,$({}^{\bullet}t_i)^{\bullet}\cap En(\boldsymbol{M})$ 是一个极大冲突集. 由定理 2 和定理 3 可知,根据冲突关系 R,可以将 $En(\boldsymbol{M})$ 中所有变迁划分到不同的等价类(极大冲突集),且每个使能变迁对应一个等价类(极大冲突集). 在循环过程中,每得到一个等价类,就会从 $En(\boldsymbol{M})$ 中删除该等价类中所有变迁,因此,不会出现重复的极大冲突集. 因此算法最终返回的 $MCS(\boldsymbol{M})$ 一定包含所有极大冲突集.

再分析算法的复杂性. 设 $|T|=n$. 循环次数不超过 n,每次循环需时间 $O(n)$,因此整个算法需时间 $O(n^2)$. 证毕.

5 结论

冲突的本质是系统资源的竞争,对冲突问题的研究是 Petri 网的重要主题之一. 本文提出 Petri 网的冲突

集问题,并利用 Petri 网的实施局部性,提出极大冲突集枚举动态算法. 文章的主要结论有:(1)Petri 网的冲突集问题是 NP 完全的;(2)所提极大冲突集枚举算法的时间复杂度为 $O(m^2n)$,其中 m 是当前标识的极大冲突集数目,n 是变迁数;(3)极大冲突集枚举问题对自由选择网、非对称选择网的复杂度为 $O(n^2)$. 下阶段工作将运用上述研究结论,探索混合语义时间 Petri 网模型的调度策略[21],处理调度过程中面临的资源冲突问题,并应用于柔性制造系统、工作流系统的调度问题研究.

参考文献

[1] Girault C, Valk R. Petri Nets for Systems Engineering: A Guide to Modeling, Verification, and Applications [M]. Berlin: Springer-Verlag, 2013.

[2] Murata T. Petri nets: Properties, Analysis and Applications [J]. Proceedings of the IEEE, 1989, 77(4): 541 – 580.

[3] Yen H C. Priority conflict-free Petri nets[J]. Acta informatica, 1998, 35(8): 673 – 688.

[4] Merlin P M, et al. Recoverability of communication protocols-Implications of a theoretical study[J]. IEEE Transactions on Communications, 1976, 24(9): 1036 – 1043.

[5] Genrich H J. Predicate/Transition Nets[M]. Berlin: Springer-Verlag Petri Nets: Central Models and Their Properties, 1987.

[6] 林闯. 随机 Petri 网和系统性能评价[M], 北京: 清华大学出版社, 2009.

[7] Balbo G. Introduction to Stochastic Petri Nets[M]. Berlin: Springer-Verlag Lectures on Formal Methods and PerformanceAnalysis, 2001.

[8] Zeng Q, Liu C, Duan H. Resource conflict detection and removal strategy for nondeterministic emergency response processes using Petri nets[J]. Enterprise Information Systems, 2015: 1 – 22.

[9] Tian Z, Zhang Z D, Ye Y D, et al. Analysis of real-time system conflict based on fuzzy time Petri nets[J]. Journal of Intelligent & Fuzzy Systems: Applications in Engineering and Technology, 2014, 26(2): 983 – 991.

[10] Popescu C, Soto M C, Lastra J L M. A Petri net-based approach to incremental modelling of flow and resources in service-oriented manufacturing systems [J]. International Journal of Production Research, 2012, 50(2): 325 – 343.

[11] Cheng A, et al. Complexity results for 1-safe nets[J]. Theoretical Computer Science, 1995, 147(1 – 2): 117 – 136.

[12] Esparza J. Reachability in live and safe free-choice Petri nets is NP-complete[J]. Theoretical Computer Science, 1998, 198(1): 211 – 224.

[13] Esparza J. Decidability and Complexity of Petri Net Problems—an Introduction[M]. Berlin Heidelberg Springer: Lectures on Petri Nets I: Basic Models, 1998: 374 – 428.

[14] Jiang C. Polynomial-time algorithm for the legal firing sequences problem of a type of synchronous composition Petri nets[J]. Science in China Series: Information Sciences, 2001, 44(3): 226 – 233.

[15] Badouel E, Bernardinello L, Darondeau P. The synthesis problem for elementary net systems is NP-complete[J]. Theoretical Computer Science, 1997, 186(1): 107 – 134.

[16] 潘理, 赵卫东, 王志成, 等. Petri 网的步问题研究[J]. 软件学报, 2009, 20(3): 505 – 514.
Li Pan, Weidong Zhao, Zhicheng Wang. On the step problem for Petri nets[J]. Journal of Software, 2009, 20(3): 505 – 514 (in Chinese)

[17] Papadimitriou C H. Computational Complexity[M]. Chichester: John Wiley and Sons Ltd., 2003.

[18] Garey M R, Johnson D S. Computers and Intractability [M]. New York: W. H. Freeman & Co Ltd, 2002.

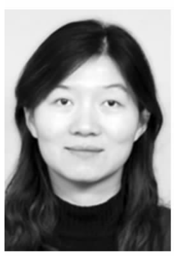
[19] Bron C, Kerbosch J. Algorithm 457: finding all cliques of an undirected graph[J]. Communications of the ACM, 1973, 16(9): 575 – 577.

[20] Tomita E, Tanaka A, Takahashi H. The worst-case time complexity for generating all maximal cliques and computational experiments[J]. Theoretical Computer Science, 2006, 363(1): 28 – 42.

[21] 潘理, 丁志军, 郭观七. 混合语义时间 Petri 网模型[J]. 软件学报, 2011, 22(6): 1199 – 1209.
Li Pan, Zhijun Ding, Guanqi Guo. Time Petri net model with mixed semantics[J]. Journal of Software, 2011, 22(6): 1199 – 1209. (in Chinese)

作者简介

潘　理　男, 1975 年出生, 湖南平江人. 博士, 湖南理工学院副教授. 研究方向为 Petri 网、形式化建模与优化.
E-mail: panli.hnist@gmail.com



郑　红(通信作者)　女, 1973 年出生, 博士, 华东理工大学副教授, 美国加州大学河滨分校访问学者. 研究方向为普适计算、Petri 网应用.
E-mail: zhenghong@ecust.edu.cn