Petri 网的符号 ZBDD 可达树分析技术

李凤英[1),2)] 古天龙[2)] 徐周波[1),2)]

[1)](西安电子科技大学电子工程学院 西安 710071)

[2)](桂林电子科技大学计算机学院 广西 桂林 541004)

摘 要 Petri 网是一种适合于并发系统建模、分析和控制的图形工具.可达树是 Petri 网分析的典型技术之一,它通过标识向量集合表征系统的状态空间,组合复杂性严重制约了该分析技术可处理系统问题的规模.零压缩决策图(Zero-Suppressed Binary Decision Diagrams,ZBDD)是一种新型的数据结构,是表示和处理稀疏向量集合的一种有效技术.文章基于 Petri 网可达标识向量的稀疏特征,给出了 Petri 网分析的符号 ZBDD 技术,该技术通过对标识向量(状态)的布尔向量表示、可达标识向量(状态)的符号 ZBDD 生成,实现 Petri 网可达状态空间的高效符号操作和紧凑符号表示.实验表明,基于 ZBDD 的符号可达性分析算法能够有效处理较大规模 Petri 网问题.

关键词 Petri 网;零压缩二叉决策图;可达树;状态空间

中图法分类号 TP393 **DOI 号**: 10.3724/SP.J.1016.2009.02420

Symbolic Reachability Analysis of Petri Nets Using ZBDDs

LI Feng-Ying[1),2)] GU Tian-Long[2)] XU Zhou-Bo[1),2)]

[1)](*Electronic Engineering School, Xidian University, Xi'an* 710071)

[2)](*School of Computer Science, Guilin University of Electronic Technology, Guilin, Guangxi* 541004)

Abstract Petri net is a graph-based mathematical formalism suitable to model, analyze and control the behavior of discrete event concurrent systems. Reachability tree is one of typical techniques to analyze Petri nets. Highly concurrent systems often suffer from state combinatorial explosion problem which renders it very hard and even impossible to handle Petri nets using the reachability tree technique. Fortunately, the marking vectors or the state spaces of Petri nets are often very sparse, and Zero-Suppressed Binary Decision Diagram (ZBDD) is a novel data structure and efficient way to represent and manipulate 0—1 sparse vectors. The authors developed the novel ZBDD-based symbolic technique to formulate, generated and manipulated the reachability marking vectors, thus the performance of Petri nets could be evaluated symbolically. The experimental results show that the symbolic technique can handle large-scale Petri nets more efficiently and compactly.

Keywords Petri net; Zero-suppressed binary decision diagram;

1 引 言

Petri 网是一种适于对具有异步、并发特征的离散事件系统进行模拟、分析、设计和控制的图形数学工具,已在制造系统、通信网络、数字电路综合与验证等领域得到了广泛应用[1-3].Petri 网分析是其应用中的一个关键环节,现有的 Petri 网分析方法主要有可达树、状态方程以及结构化简等.可达树方法通过枚举方式穷尽生成 Petri 网的所有可达标识向

收稿日期:2007-08-20;最终修改稿收到日期:2009-11-09.本课题得到国家自然科学基金(60963010,60903079)资助.**李凤英**,女,博士研究生,主要从事符号模型检验、Petri 网、符号调度技术的研究.E-mail: lfy@guet.edu.cn. **古天龙**,男,博士,教授,博士生导师,主要从事形式化方法、符号计算、知识工程等的研究. **徐周波**,女,博士研究生,主要从事符号调度技术、符号模型检验、软件测试的研究.

量,并以树结构形式表示 Petri 网的可达状态空间,进而判定相应的活性、安全性、有界性、可达性等性质.对于有界 Petri 网,可达树方法是一种有效和实用的技术;对于无界 Petri 网,则需要采用可达树方法的扩展技术——可达图方法.该类方法的局限性在于:状态组合复杂性制约了可处理 Petri 网模型的规模;状态方程方法通过 Petri 网的关联矩阵建立标识向量和迁移引发向量之间的矩阵代数方程,基于矩阵代数方程求解 Petri 网的位置(迁移)不变量,根据不变量的特征来判定 Petri 网的一些性质.该方法适合于特殊结构 Petri 网,且随着 Petri 网模型规模的增大关联矩阵的维数也随之增大,对于大规模 Petri 网的处理也会带来不便;结构化简方法通过合并特殊的位置、迁移或结构,在保持一定性质不变的前提下来降低 Petri 网的规模.该方法要求所处理 Petri 网具有特定的结构,一般作为前两种方法的辅助技术使用.在 Petri 网模型中,由系统自身的并发特性和状态迁移的语义交织引起的状态组合复杂性,是 Petri 网分析技术中的一个瓶颈问题,严重制约了 Petri 网的应用[4].

状态组合复杂性一直是计算机科学和应用人员关注的一个重要问题.减缓或者部分程度上避免状态组合复杂性问题的一种可行策略是:状态的符号或者隐式描述.有序二叉决策图(Ordered Binary Decision Diagram,OBDD)则是迄今为止最为有效的符号技术之一[5-6].OBDD 为布尔函数提供了一种有效和规范的描述方法,同时,布尔函数的所有复杂运算都可以基于 OBDD 数据结构得到极大地简化实现.尽管,OBDD 技术并不能克服所有应用中的状态组合爆炸,但确实解决了许多无法解决的大规模状态应用问题(如,OBDD 可处理具有 10^{20} 状态的应用系统,而显式枚举所能处理的系统状态为 $10^3\sim10^6$).Pastor 等建立了基于有序 OBDD 的 Petri 网符号分析方法,其实质是用 OBDD 表示 Petri 网标识集的特征函数,对 Petri 网的各种性能进行分析.该方法适用于有界 Petri 网,为大规模 Petri 网模型的分析探索了一条新的有效途径[7-8].零压缩二叉决策图(Zero-Suppressed Binary Decision Diagrams,ZBDD)是 OBDD 的一种扩展形式,是描述和操作稀疏二进制向量的高效技术[9-11].在 ZBDD 技术中,ZBDD 的结点数取决于二进制向量集合中 1 的个数.在 Petri 网中,位置中的令牌(token)对应于标识向量中非零元素,并且 Petri 网的标识向量具有稀疏特征.鉴于此,我们建立了基于 ZBDD 的 Petri 网符号可达树分析技术,给出了 Petri 网的符号 ZBDD 表示、可达标识向量生成算法.算例仿真结果表明了符号 ZBDD 分析技术的有效性.

2　预备知识

2.1　Petri 网

定义 1.　一个 Petri 网定义为六元组:$PN=\langle P,T,F,W,K,M_0\rangle$,其中,

$P=\{p_1,p_2,\cdots,p_n\}$为位置集合;

$T=\{t_1,t_2,\cdots,t_m\}$为迁移集合($P\cap T=\varnothing$,$P\cup T\neq\varnothing$);

$F\subseteq(P\times T)\cup(T\times P)$为流关系(关系弧);

W:$F\to Z^+$ 为流关系的权函数($Z^+=\{1,2,3,\cdots\}$),即关系弧的权重;

K:$P\to Z^+\cup\{\omega\}$,是位置上的容量函数(ω 为无穷大数);

$\boldsymbol{M}_0$:为初始标识向量.一个标识向量是一个函数 M:$P\to Z^+\cup\{0\}$,表示位置中令牌的分布情况.

定义 2.　对于 $t\in T$,$p\in P$,令 ${}^\cdot t=\{p\mid(p,t)\in F\}$,$t^\cdot=\{p\mid(t,p)\in F\}$,${}^\cdot p=\{t\mid(t,p)\in F\}$,$p^\cdot=\{t\mid(p,t)\in F\}$,则称 $t^\cdot({}^\cdot p)$为迁移 t(位置 p)的前集或输入集,$t^\cdot(p^\cdot)$为迁移 t(位置 p)的后集或输出集.

Petri 网中位置是静态的,迁移是动态的.迁移的引发将使位置中令牌的分布发生变化,即产生新的标识.迁移 $t_1,t_2,\cdots,t_r$ 的依次引发,将使标识依次发生变化 $M_0,M_1,\cdots,M_r$,相应地产生一引发序列 $\sigma=M_0t_1M_1t_2\cdots t_rM_r$,或简记为 $\sigma=t_1t_2\cdots t_r$,并称标识 M_r 是从 M_0 引发 σ 可达的,记为 $M_0[\sigma\rangle M_r$ 或 $M_r\in[M_0\rangle$.

定义 3.　在一个具有初始标识 M_0 的 Petri 网中,位置 $p\in P$ 为 k-有界的($k\in Z^+$),当且仅当 $\forall M\in[M_0\rangle$:$M(p)\leq k$.一个 Petri 网为 k-有界的,当且仅当 Petri 网中每一位置都是 k-有界.Petri 网是安全的,当且仅当它是 1-有界的.显然,对于安全 Petri 网,对于 $\forall M\in[M_0\rangle$有:$M(p)\leq1$,且每条弧上的权重为 1,所以安全 Petri 网可以简写为四元组:$PN=\langle P,T,F,M_0\rangle$.

定义 4(迁移的使能条件).　如果 $\forall p\in{}^\cdot t$ 有 $M(p)\geq W(p,t)$,且 $\forall p\in t^\cdot$ 有 $M(p)\leq K(p)-W(t,p)$,则称迁移 t 是使能的或者具有引发权.

定义 5(迁移的引发规则).　在标识 M 下,使能迁移 t 的引发将产生新的标识 M',记为 $M[t\rangle M'$:

$$M'(p)=M(p)-W(p,t)+W(t,p).$$

2.2 零压缩二叉决策图

对于从$\{0,1\}^n$到$\{0,1\}$的布尔函数$f(x_1,x_2,\cdots,x_n)$,若第i个分量x_i取值为0或1,则得到$\{0,1\}^{n-1}$到$\{0,1\}$的布尔函数$f(x_1,\cdots,x_{i-1},1,x_{i+1},\cdots,x_n)$或$f(x_1,\cdots,x_{i-1},0,x_{i+1},\cdots,x_n)$,简记为$f_{x_i}$或$f_{x_i'}$.布尔函数$f_{x_i}$和$f_{x_i'}$分别称为布尔函数$f(x_1,x_2,\cdots,x_n)$关于变量$x_i$的1-分量($x_i$分量)和0-分量($x_i'$分量).

定义6. 对于从$\{0,1\}^n$到$\{0,1\}$的布尔函数$f(x_1,x_2,\cdots,x_n)$和给定变量序π(即$x_1,x_2,\cdots,x_n$是有顺序的),有序二叉决策图(OBDD)是一个无环有向图,它满足:①OBDD中结点分为根结点、终结点和内部结点三类;②终结点仅有两个,分别标记为0和1,并表示布尔常量0和1;③根结点和内部结点具有四元组属性($pointer(u)$,$var(u)$,$low(u)$,$high(u)$),其中,$pointer(u)$表示结点u所对应的布尔函数(对于根结点u,$pointer(u)=f(x_1,x_2,\cdots,x_n)$);$var(u)$表示结点$u$的标记变元(根结点$u$的标记变元为变量序$\pi$中的第一个变量);$low(u)$表示结点$u$的0分枝子结点,对应于该结点布尔函数$pointer(u)$中变元$var(u)$取0值后的布尔函数;$high(u)$表示结点$u$的1分枝子结点,对应于该结点布尔函数$pointer(u)$中变元$var(u)$取1值后的布尔函数;④根结点和内部结点均具有两个输出分枝弧,将它们和各自的两个分枝子结点联系在一起.结点u和$low(u)$的连接弧称为0-边,结点u和$high(u)$的连接弧称为1-边,且满足:对于结点u,$low(u)\neq high(u)$;对于$var(u)=var(v)$的不同结点u和v,则$low(u)\neq low(v)$或者$high(u)\neq high(v)$;⑤任一有向路径上,布尔函数$f(x_1,x_2,\cdots,x_n)$中的每个变元均以变量序π所规定的次序依次至多出现一次.

在图形表示中,用方框表示终结点,用圆圈表示其它结点,结点之间通过虚线或者实线连接.通常,假设连接弧的方向向下,0-边用虚线表示,1-边用实线表示.

零压缩二叉决策图(ZBDD)是一种特殊的OBDD,是日本学者Minato为了克服OBDD表示组合集合(有限集合A的组合集合是该集合的幂集2^A的一个子集合)的不足提出的,进一步降低了组合集合表示的空间需求,从而更为有效地处理组合集合相关的问题[9-10].

对于具有n个元素的集合$A=\{a_1,a_2,\cdots,a_n\}$,从A中不考虑次序地取出任意r个元素,其中$0\leq r\leq n$,所得到的集合称为集合A中元素的一个组合.若集合A中的每个元素a_i均对应一个布尔变量x_i,则组合集合中的元素可以用一个n维布尔向量$(x_1,x_2,\cdots,x_n)$表示,当元素a_i不包含在组合中时,对应布尔变量$x_i=0$,否则$x_i=1$.由此可见组合集合可以用一个n元布尔函数$f(x_1,x_2,\cdots,x_n)$表示.OBDD为布尔函数提供了紧凑规范的表示,在很多情况下OBDD表示可大大降低组合集合表示的空间需求,使表示组合集合所用的结点数远小于集合包含的元素个数,并且使集合上的操作时间与表示集合所用的结点数目成正比.但是,用OBDD表示组合集合仍然存在一些问题,即组合集合的OBDD表示依赖于布尔变量的个数,当布尔变量个数不同时,相同的组合集合会得到全然不同的OBDD.比如对于两个不同的布尔变量域$\{a,b,c\}$和$\{a,b,c,d\}$,要表示相同的组合集合$\{a,b\}$,就会有两种截然不同的OBDD表示,如图1(a)和(b)所示.这表明OBDD表示与布尔变量域的选取有关.

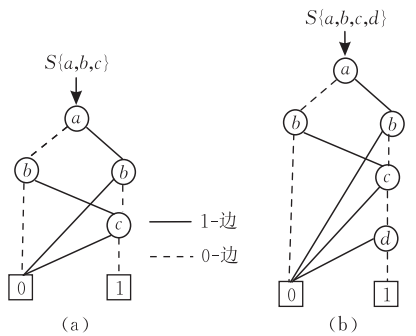


图1 组合集合的OBDD表示

在ZBDD中,通过采用不同的冗余结点删除规则,有效改善了表示组合集合的效率.这也正是ZBDD不同于OBDD之处.在OBDD中,采用的是S-删除规则:删除那些0分枝结点和1分枝结点相同的冗余结点,如图2(a)所示.在ZBDD中,采用的是pD-删除规则:删除1分枝结点为0终结点的结点,如图2(b)所示.使用了新的删除规则的ZBDD中,未出现变量的缺省就表示该变量取值为0.

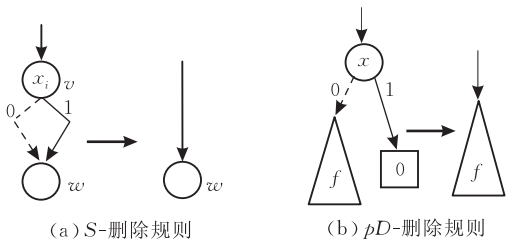


图2 不同的删除规则

图3给出了不同的布尔变量域$\{a,b,c\}$和$\{a,b,c,d\}$下组合集合$\{a,b\}$的ZBDD表示.对于同一个组

合集合,只有一种 ZBDD 表示,这是因为 ZBDD 表示与布尔变量域无关. 由图 3 可见,ZBDD 表示组合集合比 OBDD 更为紧凑,这得益于 ZBDD 所采用的 pD-删除规则. 从 ZBDD 的根结点到 1-终结点的路径数正好跟组合集合的元素个数相等,每条路径对应一个组合,路径中结点的个数与组合中变量的个数相等.

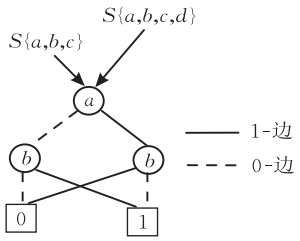


图 3　组合集合的 ZBDD 表示

ZBDD 的基本操作有 9 个[①],这里主要介绍其中 3 个,分别由一个基本函数实现,其中 P 和 Q 表示以 ZBDD 表示的集合,v 表示输入变量(布尔变量域):

*Subset*0(P,v):返回 P 中不包含元素 v 的所有组合的集合;

*Subset*1(P,v):返回 P 中包含元素 v 的所有组合的集合;

Change(P,v):对 P 中的每个组合,若包含 v,在该组合中删除 v;若不包含 v,在该组合中增加 v.

对于变量集合 $V=\{v_1,v_2,\cdots,v_n\}$,*Subset* 操作和 *Change* 操作可进行如下扩展:

*Subset*1(P,V)=*Subset*1(*Subset*1(…(*Subset*1$(P,v_1),\cdots),v_{n-1}),v_n$)

*Subset*0(P,V)=*Subset*0(*Subset*0(…(*Subset*0$(P,v_1),\cdots),v_{n-1}),v_n$)

Change(P,V)=*Change*(*Change*(…(*Change*$(P,v_1),\cdots),v_{n-1}),v_n$)

3　安全 Petri 网的符号分析

对于安全 Petri 网 $PN=(P,T,F,M_0)$,设 $P=\{p_0,p_1,\cdots,p_m\}$,任一可达标识的特征函数都可以用位置集合的一个子集来表示,在子集中出现的位置,说明该位置包含了令牌. 标识集(标识向量构成的集合)可以用一个关于 P 的组合集合表示. 例如,图 4 给出了一个包含 5 个位置 $P=\{p_0,p_1,p_2,p_3,p_4\}$的安全 Petri 网,用$\{p_0\,p_2\}$表示初始标识位置 p_0 和 p_2 含有令牌,位置 p_1,p_3 和 p_4 不含有令牌的标识. 空标识集用逻辑"0"表示,对应 ZBDD 的 0-终结点,本文也用"logicZero"表示,标识全集(不一定与可达标识集相等)的特征函数用逻辑"1"表示,对应 ZBDD 的 1-终结点,本文也用"logicOne"表示. 标识集之间的操作可以通过 ZBDD 提供的运算来完成.

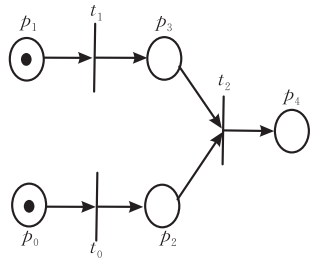


图 4　安全 Petri 网的例

Petri 网的结构蕴含着一个局部状态改变集,可称之为迁移函数,迁移函数决定了 Petri 网的动态特性. 对于安全 Petri 网 $PN=(P,T,F,M_0)$,若用 E_t 表示迁移 t 的使能标识集的特征函数,则

$$E_t=\prod_{p_i\in{}^\cdot t}p_i\prod_{p_j\in t^\cdot}\overline{p_j},$$

迁移 t_i 和 t_j 并发的使能标识集为

$$E_{t_i\wedge t_j}=\prod_{p_i\in{}^\cdot t_i}p_i\prod_{p_j\in{}^\cdot t_j}p_j\prod_{p_l\in t_i^\cdot}\overline{p_l}\prod_{p_k\in t_j^\cdot}\overline{p_k}\quad(i\neq j,l\neq k),$$

在标识 M 下使能迁移 t 的迁移函数 δ^t 将每个属于 E_t 的标识 M 变为新的标识 M',定义为

$$M'=\delta^t(M)=\begin{cases}1, & \text{若 } p_i\in t^\cdot\\ 0, & \text{若 } p_i\in{}^\cdot t\\ p_i, & \text{其它}\end{cases}.$$

考虑在一个标识集而不是一个标识的情形下引发迁移 t 的情况. 这种情况下,标识集 $\mathcal{M}_1$ 由于迁移 t 引发而变换成标识集 $\mathcal{M}_2$. 通常把迁移函数 δ^t 导出的这种求一个标识集下由于迁移 t 引发而得到的可达标识集的运算称作 *image* 运算,*image* 运算可表达为

$$\begin{aligned}\mathcal{M}_2&=img(\mathcal{M}_1,t)\\&=\{M_2\mid\exists M_1\in\mathcal{M}_1\wedge E_t,\delta^t(M_1)=M_2\},\end{aligned}$$

t_i 和 t_j 并发时,可表达为

$$img(\mathcal{M}_1,t_i\wedge t_j)=\mathcal{M}_2=\{M_2\mid\exists M_1\in\mathcal{M}_1\wedge E_{t_i\wedge t_j},\ \delta^{t_i}(M_1)=M'\wedge\delta^{t_j}(M')=M_2\},$$

更多个迁移并发的情况可进行类似推广计算.

标识集 $\mathcal{M}$ 下一步能到达的标识集的计算可表达为

$$\Delta(\mathcal{M})=\sum_{\forall t\in T}(img(\mathcal{M},t)).$$

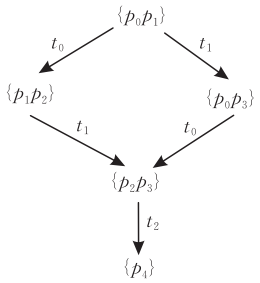
① An Intruduction to Zero-Suppressed Binary Decision Diagrams. http://www.ee.pdx.edu/~alan-mi/research.htm

标识集 $\mathcal{M}$ 的可达标识集可通过一个运用 *image* 运算的符号遍历算法求出. 该算法在宽度优先搜索算法的基础上,引用贪婪链技术实现了求可达标识的集合计算,即每次计算都是求一个标识集的可达标识集,而不是只求某个标识的可达标识,因而极大地减少了生成 Petri 网的所有可达标识集的运算次数. 图 5 给出了 Petri 网的符号遍历算法.

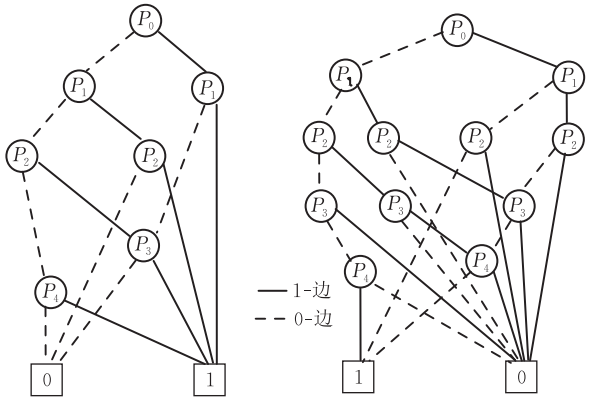
```
Traverse_Petri-net(PN=(P,T,F,W,M0))
{  Reached=From={M0};
   do {
      for each t∈T do From=From+img(From,t);
      New=From−Reached;
      From=New;
      Reached=Reached+New;
     } while (New!=∅)
   return Reached; /* M0 的可达标识集 */
}
```

图 5 Petri 网的符号遍历算法

以图 4 所示的安全 Petri 网为例,利用图 5 所示的 Petri 网符号遍历算法,用一次外部迭代和三次内部迭代就可得到所有可达标识(如图 6(a)所示). 图 6(b)为可达标识集的 ZBDD 表示,只用了 9 个结点. 图 6(c)为该可达标识集的 OBDD 表示,需要 14 个结点.



(a) 可达树



(b) ZBDD表示 (c) OBDD表示

图 6 安全 Petri 网的符号 ZBDD 可达树分析

4 加权和有界 Petri 网的符号分析

这一节主要将 Petri 网的符号 ZBDD 表示和分析技术扩展到加权和有界 Petri 网.

4.1 位置编码

一个最多包含 k 个令牌的位置 $p\in P$ 可以由一个布尔变量集 $p^0,\cdots,p^{K_p}$ 来表示,用它来为 p 中的 k 个令牌编码. 使用二进制编码时,一个 k-有界的位置需要 $\lceil\log_2(k+1)\rceil$ 个布尔变量,即 $K_p=\lceil\log_2(k+1)\rceil-1$. 例如,在 3-有界的 Petri 网中,需要两个布尔变量. 对形如 $N=\sum_{0}^{K_p}n_i2^i$ 的自然数 N,则

$$\forall i\in 0,\cdots,K_p,p^i=\begin{cases}1, & n_i=1\\ 0, & n_i=0\end{cases}.$$

以图 7 所示 Petri 网为例,位置 p_0、p_1、p_2 和 p_3 的界分别为 3,2,2 和 6. 因此位置 p_0、p_1 和 p_2 需要两个变量编码,分别为 $p_0^0p_0^1$,$p_1^0p_1^1$ 和 $p_2^0p_2^1$,p_3 需要 3 个变量编码,即 $p_3^0p_3^1p_3^2$. 该 Petri 网的初始标识的布尔特征函数为 $\{p_0^1p_2^0\}$.

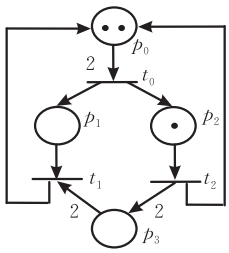


图 7 加权和有界 Petri 网

4.2 迁移引发

这里主要讨论加权和有界 Petri 网的迁移函数和迁移关系的二进制编码问题. 迁移 t 使能的标识集的特征函数(E_t)写成如下形式:

$$E_t=\prod_{p\in{}^\cdot t}(M(p)\geq W(p)\wedge M(p)\leq k).$$

在标识 M 下使能迁移 t 的迁移函数可写成如下形式:

$$\delta^t(M)=\begin{cases}M(p)-W(p,t), & p\in{}^\cdot t\backslash t^\cdot\\ M(p)+W(t,p), & p\in t^\cdot\backslash{}^\cdot t\\ M(p)-W(p,t)+W(t,p), & p\in{}^\cdot t\cap t^\cdot\\ M(p), & \text{其它}\end{cases}$$

使用二进制编码表示时,用一组布尔变量 $p^0,\cdots,p^{K_p}$ 表示位置 p 中令牌的数量,用同样数量的布尔常量 $w^0,\cdots,w^{K_p}$ 表示权重 $W(p,t)$,关系 $M(p)\geq W(p,t)$ 可以用下面的式子来描述:

$$M(p)\geq W(p)=(p^{K_p}>w^{K_p})\cdot(p^{K_p-1}>\overline{p^{K_p-1}})\cdot\cdots\cdot(p^1+\overline{p^1})\cdot(p^0+\overline{p^0})+(p^{K_p}=w^{K_p})\cdot(p^{K_p-1}>w^{K_p-1})\cdot(p^{K_p-2}+\overline{p^{K_p-2}})\cdot\cdots\cdot(p^1+\overline{p^1})\cdot(p^0+\overline{p^0})+(p^{K_p}=w^{K_p})\cdot(p^{K_p-1}=w^{K_p-1})\cdot(p^{K_p-2}>w^{K_p-2})\cdot\cdots\cdot(p^1+\overline{p^1})\cdot(p^0+\overline{p^0})+\cdots+(p^{K_p}=w^{K_p})\cdot(p^{K_p-1}=w^{K_p-1})\cdot\cdots\cdot(p^1=w^1)\cdot(p^0>w^0)+(p^{K_p}=w^{K_p})\cdot(p^{K_p-1}=w^{K_p-1})\cdot\cdots\cdot(p^1=w^1)\cdot(p^0=w^0).$$

其中,· 是逻辑与操作;+是逻辑或操作;>是大于操作,$a>b$ 即表示 $a\bar{b}$.

由特征函数 E_t 定义的使能迁移 t 的标识为

$$\prod_{p\in{}^\cdot t}\Big[\sum_{i=0}^{K_p}\Big[(p^i>w^i)\cdot\prod_{j=i+1}^{K_p}(p^i=w^i)\cdot\prod_{h=0}^{i-1}(p^h+\overline{p^h})\cdot M(p)\leq k\Big]+\prod_{i=0}^{K_p}(p^i=w^i)\Big].$$

在图 7 所示 Petri 网中,每个迁移的使能标识集为

$$E_{t_0}=[(p_0^1>0)\cdot(p_0^1+\overline{p_0^0})\cdot M(p)\leq 3+(p_0^0=0)\cdot(p_0^1=1)]=p_0^0p_0^1+p_0^1,$$

$$E_{t_1}=[(p_1^1>0)\cdot(p_1^0+\overline{p_1^0})\cdot(M(p)\leq k)+(p_1^0=1)\cdot(p_1^1=0)]\cdot[(p_3^0>0)\cdot(p_3^1=1)\cdot(p_3^2=0)\cdot(M(p)\leq k)+(p_3^2>0)\cdot(p_3^0+\overline{p_3^0})\cdot(p_3^1+\overline{p_3^1})\cdot(M(p)\leq k)+(p_3^0=0)\cdot(p_3^1=1)\cdot(p_3^2=0)]=(p_1^1+p_1^0)\cdot(p_3^1+p_3^2+p_3^0p_3^1+p_3^0p_3^2+p_3^1p_3^2),$$

$$E_{t_2}=[(p_2^1>0)\cdot(p_2^0+\overline{p_2^0})\cdot(M(p)\leq k)+(p_2^0=1)\cdot(p_2^1=0)]=p_2^0+p_2^1.$$

对于任何使能迁移 t 的标识,迁移的引发必须从其输入位置中移走相应的令牌,并在输出位置中增加相应的令牌.

位置 p 中令牌的数量用布尔变量 $p^0,p^1,\cdots,p^{K_p}$ 的向量来表示,必须要减去的令牌的数量用布尔常量的向量 $\boldsymbol{w}^0,\boldsymbol{w}^1,\cdots,\boldsymbol{w}^{K_p}$ 来表示时,迁移函数 $\delta^t(M)$ 与二进制向量表示的两个自然数的减法是等价的,可以由下面的表示:

$$\delta^t(M)=M(p)-W(p,t)=(p^0\oplus w^0,p^1\oplus w^1\oplus B_0,\cdots,p^{K_p}\oplus w^{K_p}\oplus B_{K_p-1}),$$

其中,⊕为异或操作;$B_j(j=0,1,\cdots,K_P)$ 为借位函数,定义为

$$B_0=\overline{p^0}\cdot w^0$$

$$B_j=\begin{cases}\overline{p_i^j}\cdot w^j+B_{j-1}(p_i^j\equiv w^j), & \text{若 } j\geq 1\\ 0, & \text{其它}\end{cases}.$$

其中,≡表示逻辑相等,$a\equiv b$ 即表示 $ab+\bar{a}\bar{b}$.

另一方面,必须添加的令牌的数量可以由另一个布尔常量的向量 $\boldsymbol{w}^0,\boldsymbol{w}^1,\cdots,\boldsymbol{w}^{K_p}$ 表示,这时,迁移函数 $\delta^t(M)$ 与二进制向量表示的两个自然数的加法是等价的,可以由下面的等式表示:

$$\delta^t(M)=M(p)+W(t,p)=(p^0\oplus w^0,p^1\oplus w^1\oplus C_0,\cdots,p^{K_p}\oplus w^{K_p}\oplus C_{K_p-1}),$$

其中,C_j 为进位函数,定义为

$$C_0=p^0\cdot w^0,$$

$$C_j=\begin{cases}p_i^j\cdot w^j+C_{j-1}(p_i^j\oplus w^j), & \text{若 } j\geq 1\\ 0, & \text{其它}\end{cases}.$$

以上两种情况都必须选择合适的变量 w^j,其中 w^j 表示自然数常量 W 的第 j 位:

$$w^j=\begin{cases}W^j(p_i,t), & \text{若 } p_i\in{}^\cdot t\backslash t^\cdot\\ W^j(t,p_i), & \text{若 } p_i\in t^\cdot\backslash{}^\cdot t\\ [W(p_i,t)-W(t,p_i)]^j, & \text{若 } p_i\in t^\cdot\cap{}^\cdot t\wedge W(p_i,t)>W(t,p_i)\\ [W(t,p_i)-W(p_i,t)]^j, & \text{若 } p_i\in t^\cdot\cap{}^\cdot t\wedge W(t,p_i)>W(p_i,t)\end{cases}.$$

综上所述,位置 p_i 的第 j 个变量的迁移函数 $\delta_i^{t^j}$ 为

$$\delta_i^{t^j}(M)=\begin{cases}p_i^j\oplus w^j\oplus B_{j-1}, & \text{若 } p_i\in{}^\cdot t\backslash t^\cdot\\ p_i^j\oplus w^j\oplus C_{j-1}, & \text{若 } p_i\in t^\cdot\backslash{}^\cdot t\\ p_i^j\oplus w^j\oplus B_{j-1}, & \text{若 } p_i\in t^\cdot\cap{}^\cdot t\wedge W(p_i,t)>W(t,p_i)\\ p_i^j\oplus w^j\oplus C_{j-1}, & \text{若 } p_i\in t^\cdot\cap{}^\cdot t\wedge W(t,p_i)>W(p_i,t)\\ p_i^j, & \text{其它}\end{cases}.$$

所以,迁移函数的计算式为

$$\delta^t(M)=\prod_{i=0}^{|P|}\prod_{j=0}^{K_{p_i}}\delta_i^{t^j}(M).$$

映像函数 img 扩展到有界和加权 Petri 网为

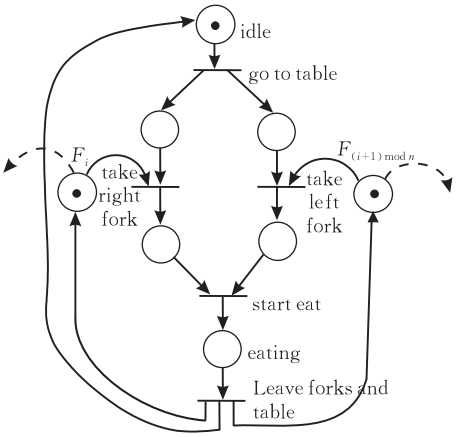
$$img(\mathcal{M}_1,t)=\mathcal{M}_2=\{M_2\mid\exists M_1\in\mathcal{M}_1\wedge E_t,\delta^t(M_1)=M_2\}.$$

在图 5 的 Petri 网符号遍历算法基础上,我们采用上式给出的映像函数 img,就可以实施加权和有界 Petri 网的可达状态空间的生成.

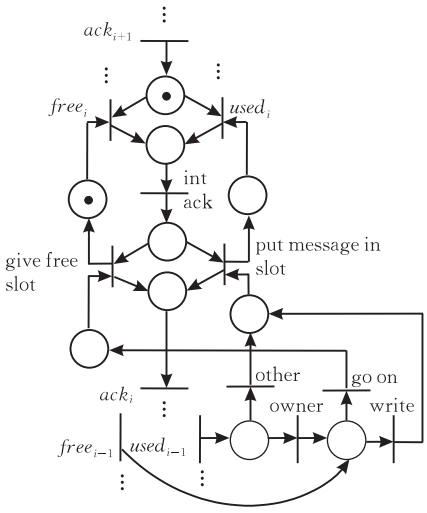
5　实验结果及分析

为了检验 Petri 网符号 ZBDD 分析算法的性

能,选两组例子进行实验.第一组为可以升级的例子,主要检验安全 Petri 网的符号 ZBDD 分析算法的性能.一个是典型的哲学家就餐问题,其中 n 是哲学家的人数,图 8(a)给出了一个哲学家就餐问题的 Petri 网表示[8],图中虚线箭头表示可以扩展为 n 个哲学家就餐问题.另一个是局域网的槽环协议,图 8(b)所示的是一个结点的槽环协议的 Petri 网模型[8],虚线表示可以扩展为 n 个结点的槽环协议.



(a) 一个哲学家就餐问题



(b) 一个结点的槽环协议

图 8 Petri 网模型的例

第二组实验选用柔性制造系统(FMS)中作业调度的 Petri 网模型,主要检验有界和加权 Petri 网的 ZBDD 分析算法的性能.假设一个 FMS 有 3 台机器,4 种作业 J_1、J_2、J_3、J_4,每个作业有 4 个过程,它们对机器的需求如表 1 所示.第 1 行第 J_1 列的 M_1/M_2 表示作业 J_1 的第一个过程可以由 M_1 或 M_2 进行加工.其中 Petri 网的建模借鉴了文献[13]所介绍的方法.

表 1 作业对机器的需求

过程	作业			
	J_1	J_2	J_3	J_4
1	M_1/M_2	M_1/M_2	$M_1/M_2/M_3$	M_2/M_3
2	M_2	M_3	M_2/M_3	M_1/M_3
3	M_1/M_3	M_1/M_2	M_1/M_3	M_2/M_3
4	M_1/M_2	M_1/M_3	M_1/M_2	$M_1/M_2/M_3$

利用 Colorado 大学的 CUDD① 软件包,用 C 语言进行编程,在运行平台 Windows 2000、P4 1500MHz CPU、128MB RAM 上,对上面例子进行了实验,实验结果如表 2 和表 3.表 2 和表 3 中位置列表示 Petri 网模型中位置的个数,迁移列表示迁移的个数,变量是 OBDD 或 ZBDD 所用变量的个数,可达标识是可达标识的个数,可达结点是可达标识集的 OBDD 或 ZBDD 表示的结点数,最大结点是算法运行中所需 OBDD 或 ZBDD 结点数的最大值,迭代次数是遍历算法中需要迭代的次数,时间是算法运行的 CPU 时间,单位为秒.

从表 2 所示实验结果的可达结点列可以看出,对于同样的可达标识集,ZBDD 的大小是 OBDD 的一半或三分之一.比如 phil_15 中,1.0×10^{10} 个可达标识用 OBDD 表示需要 584 个结点,而用 ZBDD 表示只用 312 个结点.从时间列可以看出,用 ZBDD 比用 OBDD 要快,尤其对于大规模 Petri 网,效果更加明显.这就说明,基于 ZBDD 的符号分析算法可用于较大规模 Petri 网的可达树生成和分析.

表 3 中名称列的 Jobi_j_k 表示该 FMS 中有 i 个作业、j 台机器、生产批量为 k.从该表的可达结点列可以看出,有界 Petri 网的标识向量不像安全 Petri 网那样稀疏,所以可达标识的结点数相差不是很悬殊,但 ZBDD 的结点数还是要比 OBDD 的结点数少.从最大结点列可以看出,由于对迁移的使能函数做了改进,ZBDD 算法在执行过程中所需的最大结点数大大减少.这样,对于较大规模的有界 Petri 网,ZBDD 算法能在合理时间内求得其相应的可达标识集.

6 结束语

零压缩二叉决策图(ZBDD)能有效地表示组合集合,是求解组合集合相关问题的有效工具.根据

① CUDD: CU decision diagram package release 2.3.1. http://vlsi.Colorado.edu/fabio/CUDD/cuddIntro.htm

表 2　安全 Petri 网的实验结果

名称	位置	迁移	可达标识	OBDD					ZBDD				
				变量	最大结点	可达结点	迭代次数	时间	变量	最大结点	可达结点	迭代次数	时间
Phil_5	35	25	2.2×10^{3}	35	15712	189	2	0.06	35	5255	92	2	0.03
Phil_8	56	40	2.2×10^{5}	56	22288	279	2	0.44	56	17054	158	2	0.06
Phil_10	70	50	4.7×10^{6}	70	39748	403	2	0.89	70	28760	202	2	0.13
Phil_15	105	75	1.0×10^{10}	105	68613	584	2	4.61	105	71465	312	2	0.28
Phil_20	140	100	2.2×10^{13}	140	71663	744	2	17.02	140	133370	422	2	0.53
Slot_5	50	50	1.7×10^{6}	50	94325	540	8	1.81	50	120717	411	8	0.33
Slot_7	70	70	8.0×10^{8}	70	484231	1014	10	13.88	70	630168	791	10	1.48
Slot_9	90	90	3.8×10^{11}	90	2242495	1632	12	247.21	90	2207140	1291	12	5.20

表 3　有界 Petri 网的实验结果

名称	位置	迁移	可达标识	OBDD					ZBDD				
				变量	最大结点	可达结点	迭代次数	时间	变量	最大结点	可达结点	迭代次数	时间
Job2_3_1	27	28	11	27	946	71	2	0.01	27	596	18	2	0.02
Job2_3_2	27	28	3.0×10^{3}	37	31527	462	4	1.11	37	16700	206	4	0.05
Job2_3_4	27	28	1.6×10^{5}	47	66546	813	6	58.23	47	120012	552	6	0.23
Job2_3_8	27	28	1.6×10^{7}	57	932859	1727	10	307.75	57	773129	1932	10	1.53
Job3_3_1	41	46	1.4×10^{3}	41	18409	322	3	0.88	41	8713	125	3	0.06
Job3_3_2	41	46	1.2×10^{5}	56	183012	936	4	40.42	56	77982	375	4	0.25
Job3_3_4	41	46	3.5×10^{7}	71	溢出				71	561454	949	6	1.16
Job3_3_8	41	46	4.5×10^{10}	86	溢出				86	9946986	3050	11	20.69
Job4_3_1	55	64	1.3×10^{4}	55	59616	408	3	2.42	55	27325	199	3	0.11
Job4_3_2	55	64	3.9×10^{6}	75	溢出				75	196362	552	4	0.47
Job4_3_4	55	64	5.6×10^{9}	95	溢出				95	1477324	1361	6	3.16

Petri 网状态向量稀疏的特点,本文给出了基于ZBDD的 Petri 网的标识向量表示、使能迁移计算、可达标识向量生成的符号方法,使得 Petri 网的动态行为的计算能够在基于 ZBDD 数据结构的紧凑表示和高效操作下得到实施. 在 Petri 网分析的时间和空间效率上,符号 ZBDD 技术都体现出了较之于符号 OBDD 技术具有明显的优势.

Petri 网的可达状态空间是 Petri 网应用于系统模拟、分析、控制及其调度等的核心和基础. 我们可以在生成 Petri 网可达状态空间的过程中,通过符号 ZBDD 的操作函数方便地实施 Petri 网的安全性、活性、有界性、可达性等性质判定. 同时,我们也可以基于 Petri 网可达状态空间的紧凑符号 ZBDD 描述,开展 Petri 网模型下系统的控制策略、调度方案等的设计,这是我们正在开展的研究课题. 此外,基于 ZBDD 描述的 Petri 网性质证明、ZBDD 的变量序问题等也将是进一步开展的研究工作.

参　考　文　献

[1] Murata T. Petri nets: Properties, analysis and applications. Proceedings of IEEE, 1989, 77(4): 541-574

[2] Gu Tianlong, Bahri P A. A survey of Petri-net applications in batch processes. Computers in Industry, 2002, 47(1): 99-111

[3] Luo Jun-Zhou, Shen Jun, Gu Guan-Qun. From Petri nets to formal description techniques and protocol engineering. Journal of Software, 2000, 11(5): 606-615(in Chinese)
(罗军舟, 沈俊, 顾冠群. 从 Petri 网到形式描述技术和协议工程. 软件学报, 2000, 11(5): 606-615)

[4] Jiang Yi-Xin, Lin Chuang, Qu Yang, Yin Hao. Research on model-checking based on Petri nets. Journal of Software, 2004, 15(9): 1265-1276(in Chinese)
(蒋屹新, 林闯, 曲扬, 尹浩. 基于 Petri 网的模型检测研究. 软件学报, 2004, 15(9): 1265-1276)

[5] Bryant R E. Graph-based algorithms for Boolean function manipulation. IEEE Transactions on Computers, 1986, 35(8):677-691

[6] Bryant R E. Symbolic Boolean manipulation with ordered binary decision diagrams. ACM Computing Surveys, 1992, 24(3): 293-318

[7] Pastor E, Cortadella J. Efficient encoding schemes for symbolic analysis of Petri nets//Proceedings of the Conference on Design, Automation and Test in Europe. Paris, 1998: 790-795

[8] Pastor E, Cortadella J. Symbolic analysis of bounded Petri nets. IEEE Transactions on Computers, 2001, 50(5): 432-448

[9] Minato S. Zero-suppressed BDDs for set manipulation in combinatorial problems//Proceedings of the 30th DAC. Texas, USA, 1993: 272-277

[10] Minato S. Zero-suppressed BDDs and their applications. International Journal on Software Tools for Technology Transfer, 2001, 3(2): 156-170

[11] Yoneda T, Hatori H, Takahara A, Minato S. BDDs vs. Zero-suppressed BDDs: For CTL symbolic model checking of Petri nets//Proceedings of FMCAD'96, California, USA, 1996: 435-449

[12] Tomisaka M, Yoneda T. Partial order reduction in symbolic state space traversal using ZBDDs. IEICE Transactions on Information and Systems, 1999, E00-D(1): 704-711

[13] Jiao Li, Lu Wei-Ming. Synthesis and property-preservation of Petri net system based on shared places. Chinese Journal of Computers, 2007, 30(3): 352-360(in Chinese)
(焦莉, 陆维明. 基于共享位置的 Petri 网综合与保性. 计算机学报, 2007, 30(3): 352-360)

LI Feng-Ying, born in 1974, Ph.D.. Her research interests include formal model checking, Petri net and formal scheduling technology.

GU Tian-Long, born in 1964, Ph. D., professor, Ph. D. supervisor. His research interests include formal method, formal computing and knowledge engineering.

XU Zhou-Bo, born in 1976, Ph. D.. Her research interests include formal scheduling technology, formal model checking and software testing.

Background

This work is supported by the National Natural Science Foundations of China (60963010,60903079).

Petri nets are a graphical and mathematical formalism to model and analyze the behavior of discrete event concurrent systems, the state space of which is a set consisting of sparse vectors. But analyzing Petri nets often suffer from state combinational problem that renders it very difficult and even impossible to handle large-scale Petri nets. The existing methods for the analysis of Petri nets, such as reachability tree methods, enumerative methods, matrix-equation methods and reduction methods, have certain limitations respectively and can not overcome state explosion problem. OBDD is reported one of the most efficient symbolic technique and capable of representing large sets of markings with small data structures. ZBDD is a special type of OBDD for manipulating sets of vector combinations. The symbolic ZBDD analysis technology of Petri nets is discussed, and a symbolic ZBDD analysis algorithm for Petri nets is developed in this paper. The experimental results show that the symbolic ZBDD analysis technology can handle large-scale Petri nets, and ZBDDs are competitive for the symbolic analysis of Petri nets.