单道批处理系统的建模与验证

鱼先锋 雷丽晖 李永明
(陕西师范大学计算机科学学院 西安 710062)

摘 要 单道批处理系统的模型是其性能评价、仿真、作业调度及控制的研究基础。建立了单道批处理系统的一个数学模型——批处理自动机,并给出了相应的转换算法,将所建数学模型转换成Kripke结构;完成了基于Kripke结构的单道批处理系统模型检测,验证了单道批处理系统的合理性即兼顾公平性与效率。

关键词 批处理系统,模型检测,Kripke结构,自动机

Modeling and Verification of Batch System

YU Xian-feng LEI Li-hui LI Yong-ming
(School of Computer Science, Shaanxi Normal University, Xi'an 710062, China)

Abstract The model of simple batch systems is the foundation of its performance evaluation, simulation, scheduling, and control research. This paper introduced a mathematical model named batch automata for simple batch systems, and gave an algorithm for translating a batch automata into a Kripke structure, and completed the model checking of batch system based on the Kripke structure to verify the reasonable property, i. e. , both fair and efficiency.

Keywords Simple batch systems, Model checking, Kripke structure, Automata

1 引言

批处理系统是混合动态系统中最常见的一种。该系统包含了离散事件动态系统和连续变量动态系统,两者相互作用。一个单道批处理系统如图1所示,可描述为:

(i)该系统由作业调度器、作业队列和处理机3部分构成,其中作业队列负责管理所有的待运行作业;作业调度器则在处理机空闲时选择一个响应比高的作业投入运行;

(ii)任何时候只能有一个作业在处理机上运行;

(iii)任何一个时段有有限个作业被系统处理;一批作业处理完系统时间归零,接着处理下批作业,循环往复批处理系统可以不停地运行下去。

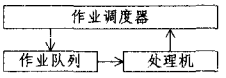

图1 一个简单的批处理系统

已有文献中,单道批处理系统的建模主要基于时间自动机、时间Petri网以及进程代数的时间扩充,其中以时间自动机与时间Petri网的影响和应用最为广泛。Alur等人提出的方法[1]把动态特性引入自动机结点,状态转移依赖结点状态值,从而得到时间自动机和混合自动机。时间自动机模型可用来设计、分析和研究混合系统[2],如Stiver利用时间自动机将逻辑监控框架扩展到混合系统。与时间Petri网相比,时间自动机的规则比较简单、直观,但用于混合系统建模、分析和综合时,有局限性,主要表现在模型的复杂性上(其离散状态空间数会随着过程数目按指数方式增加),对并发过程进行建模时尤为突出。与时间自动机不同,时间Petri网的标识规则较为复杂,但它允许对系统的同步现象进行处理,是对复杂系统的并发及冲突现象建模的较好工具,被广泛应用[3-6]。其表达形式紧凑(顶点少),且结合使用局部顺序语义学,可以用一种比时间自动机更加有效的方式对时间Petri网状态进行搜索。经扩展得到各种变种的Petri网,如文献[4]定义微分库所、微分转移以及适当规则,构成微分Petri网,以解决系统性能分析和监控设计等问题。

形式化验证是在对软件系统进行规约的基础上,分析系统是否具有所期望的性质,主要分为两种优势互补的技术:模型检测[7]和定理证明[8]。模型检测的优点有:自动化程度高,当系统不具备所期望的性质时,能给出相应的反例路径;其缺点是状态爆炸问题。定理证明的优点有:能基于无穷域上的归纳法处理无穷状态空间;其缺点是自动化程度不高,证明过程需要人工干预,不能在证明失败之后给出易于理解的反例。

本文为单道批处理系统建立了一个新数学模型,即批处理自动机。与时间自动机和时间Petri网相比,利用批处理自动机对简单批处理系统建模具有更强的适用性。建模时可引入时序控制阵来计算、存储各种调度算法下的作业响应比;其通用性强,计算简单。作者将所建模型转化成Kripke结构,将节点状态值由多值编码成二值的,给出了相应的转换算法;便于用二值逻辑公式描述系统性质。结合实例进行了基于Kripke结构和线性计算树逻辑的简单批处理系统形式化验证,所用不动点计算算法简单且自动化程度较高。

到稿日期:2010-05-26 返修日期:2010-08-04 本文受国家自然科学基金(60873119),高等学校博士学科点专项科研基金(20090202120006),中央高校基本科研业务费专项资金(GK200902017)资助。

鱼先锋(1985—),男,硕士生,主要研究方向为模型检测;雷丽晖(1976—),女,博士,副教授,主要研究方向为模型检测;李永明(1966—),男,博士,教授,博士生导师,主要研究方向为非经典计算理论、量子逻辑、量子计算等。

2 批处理系统建模

定义 1 一个批处理自动机 A 为一个八元组 $A=\langle AP, L, N, Q, q_0, \delta, F, T^t_{4\times n}\rangle$，其中：

$AP=\{p_1, p_2, \cdots, p_n\}$ 为作业之集，表示一时段要处理的一批作业；

赋值集 $L=\{0, w, r, 1\}$，0 表示作业未就绪，w 表示作业等待，r 表示作业被处理，1 表示处理结束；

Q 是有穷状态集合，$Q\subset L^{AP}$，即 AP 到 L 的部分映射；$\forall q\in Q, p_i\in AP, l_i\in L, \frac{l_i}{p_i}$ 表示第 i 个作业 p_i 的值为 l_i，记 $q=\frac{l_1}{p_1}+\frac{l_2}{p_2}+\cdots+\frac{l_n}{p_n}$，为方便表示再记 l_i 为 q_i；

N 为自然数集合，表示系统时间；

$q_0\in Q$ 为初始状态，$q_0=\frac{0}{p_1}+\frac{0}{p_2}+\cdots+\frac{0}{p_n}$ 表示这批作业都未就绪；

$F\in Q$ 为终结状态之集，$\forall q\in F, q=\frac{1}{p_1}+\frac{1}{p_2}+\cdots+\frac{1}{p_n}$ 表示该批作业全处理结束；

时序控制阵 $T^t_{4\times n}=\begin{pmatrix} T_{11} & T_{12} & \cdots & T_{1i} & \cdots & T_{1n} \\ T_{21} & T_{22} & \cdots & T_{2i} & \cdots & T_{2n} \\ T_{31} & T_{32} & \cdots & T_{3i} & \cdots & T_{3n} \\ T_{41} & T_{42} & \cdots & T_{4i} & \cdots & T_{4n} \end{pmatrix}$，其中行向 $i=1,\cdots,n$ 对应不同的作业；$j=1,2,3,4$，T_{ij} 为有限自然数：

(i) T_{1i} 表示第 i 个作业提交时刻，初始化时给出；

(ii) T_{2i} 表示第 i 个作业预计运行时长，初始化时给出；

(iii) $T^t_{3i}=f(T_{1i}, T_{2i}, t)$ 表示第 i 个作业在系统时间 t 时响应比，应最大程度地体现公平性和效率，T^t_{3i} 是随系统时间动态变化的，且不同作业调度算法定义不同。

(iv) T_{4i} 表示第 i 个作业开始被处理时刻，初始化为 0，若在系统时间 t 时 $\exists s, q\in Q$, s. t. $\delta(s,t)=q$ 且 $s_i=w, q_i=r$，则 $T_{4i}=t$；

作业处理函数为映射 $\delta: Q\times N\to Q$，系统时间为 t 时，有 $\delta(s,t)=q$，则：

(ⅰ) $q_i=w$，当 $\forall s_i=0, T_{1i}\leqslant t$；

(ⅱ) $q_i=1$，当 $\exists s_i=r$, s. t. $T_{2i}\leqslant t-T_{4i}$；

(ⅲ) $q_i=r$，当 $\forall s_i, s_i\neq r$，且 $\exists s_i=w\forall s_j=w, T^t_{3i}\geqslant T^t_{3j}$；

(ⅳ) $q=\frac{0}{p_1}+\frac{0}{p_2}+\cdots+\frac{0}{p_n}$，当 $s=\frac{1}{p_1}+\frac{1}{p_2}+\cdots+\frac{1}{p_n}$，且 $\forall s_i, T_{2i}\leqslant t-T_{4i}$；

(Ⅴ) $q_i=s_i$，Otherwise。

定义 2 若 A 在系统时间 k 时算出批作业 $p_1,\cdots,p_n$ 对应的一个终结状态，则称 k 为这批作业的处理时间；记作 $run(A,T)=k$。

定义 3 批处理函数为映射 $\delta^*: Q\times N^*\to Q$，其中：$N^*=\{S\mid S=0,1,2,\cdots,k, k\geqslant 0\}$，$\forall S=S'k, k\in N, q\in Q$，有：$\delta^*(q, S)=\delta(\delta^*(q,S'),k)$。

定义 2′ 如果存在递增自然数串 $S=0,1,2,\cdots,k$, s. t. $\delta^*(q_0,S)\in F$，则 $run(A,T)=k$。

定义 1 是对批处理系统的形式化即建模；定义 2、定义 3 是为形式化描述系统性质做准备的。

下面通过简单实例说明批处理自动机 A 是怎样运行的。

例 1 下面用批处理自动机 A 处理一个批作业，这里 $n=3$ 有 3 个作业，定义响应比为 $T^t_{3i}=f(T_{1i}, T_{2i}, t)=1+\frac{t-T_{1i}}{T_{2i}}$。

初始时序控制阵 $T^0_{4\times 3}=\begin{pmatrix} 1 & 2 & 5 \\ 5 & 4 & 2 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$

处理作业 p_1, p_2, p_3 的状态转移图如图 2 所示。

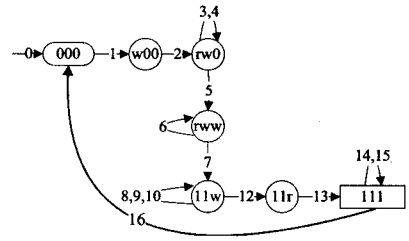

图 2 状态转移图

3 系统合理性的描述

定理 1 若 A 在时序控制阵 $T_{4\times n}$ 控制下处理批作业 $p_1,\cdots,p_n$，则有：

(ⅰ) p_i 值一定按照 0，w，r，1 的顺序进行状态变化；

(ⅱ) 在任意系统时间 $t<run(A,T)$ 时，只能有一个作业被处理；

(ⅲ) 恒存在有限自然数 $k\in N$, s. t. $run(A,T)=k$。

根据定义 1 中作业处理函数的描述，这里的(ⅰ)(ⅱ)是显然的，下面简单构造证明(ⅲ)。

证明(ⅲ)：记 $t_s=\bigvee_{i=1}^{n} T_{1i}$，$t_r=\bigvee_{i=1}^{n} T_{2i}$，我们证明 $run(A,T)\leqslant t_s+nt_r$。

t_s 时刻所有作业进入等待队列，必有作业开始被处理，不妨设第 1 个作业开始被处理，即 $p_1=r$；经过 t_r 时间，即 t_s+t_r 时 p_1 必然处理结束，即 $p_1=1$。而后存在作业 p_2 开始运行，$p_2=r$；再经过 t_r 时间，即 t_s+2t_r 时 p_2 必然处理结束，有 $p_2=1$。

这样重复地让系统时间增加 t_r，n 次后有 $p_1=p_2=\cdots=p_n=1$，即系统时间 t_s+nt_r 时已经算出一个终状态。故 $run(A,T)\leqslant t_s+nt_r$。

另外 $run(A,T)\geqslant\sum_{i=1}^{n} T_{2i}$ 是显然的，因为系统不停地处理完这 n 个作业需要的时间为 $\sum_{i=1}^{n} T_{2i}$；于是就有：$\sum_{i=1}^{n} T_{2i}\leqslant run(A,T)\leqslant t_s+nt_r$。

因为 $T_{4\times n}$ 中的元素与 n 都是有限自然数，所以存在有限自然数 $k\in N$ s. t. $run(A,T)=k$，$\sum_{i=1}^{n} T_{2i}\leqslant k\leqslant t_s+nt_r$。

定义 4 系统 A 在时序控制阵 $T_{4\times n}$ 控制下处理批作业 $p_1,\cdots,p_n$，满足定理 1 则称 $T_{4\times n}$ 是公平有效的，系统 A 是合理的。

4 构造 Kripke 结构进行模型检测

4.1 批处理系统对应的 Kripke 结构

定义 5 一个批处理系统 A 对应一个 Kripke 结构 $K=\langle AP, S, S_0, R, L_s\rangle$，其中：

AP 是作业之集，$AP=\{p_1,\cdots,p_n\}$；

S 是有穷状态之集，$S=Q$；

$s_0 \in S$ 为初始状态，$s_0=q_0$；

$R \subseteq S \times S$ 为状态转移关系，$\forall (s,s') \in S \times S, (s,s') \in R$ 当且仅当 $(s,s') \in Q \times Q, \exists t \in N$, s. t. $\delta(s,t)=s'$；

$L_s: S \to 2^{L \times AP}$ 为状态标识函数，$\forall s \in S$, $L_s(s)=\begin{pmatrix} s_{11} & \cdots & s_{1n} \\ s_{21} & \cdots & s_{2n} \\ s_{31} & \cdots & s_{3n} \\ s_{41} & \cdots & s_{4n} \end{pmatrix}$

(i)s_{1j} 表示第 j 个作业在该状态取值为“0”时的真值(0|1)；

(ii)s_{2j} 表示第 j 个作业在该状态取值为“w”时的真值(01)；

(iii)s_{3j} 表示第 j 个作业在该状态取值为“r”时的真值(0|1)；

(iv)s_{4j} 表示第 j 个作业在该状态取值为“1”时的真值(0|1)。

定义 5 是将定义 1 中描述的批处理自动机抽象成 Kripke 结构。

4.2 Kripke 结构生成算法

根据定义 5 对 L_s 的定义，$\forall s_i \in S$ 计算 $L_s(s_i)$(实际上是把多值向量编码成二值矩阵)。

4.3 用不动点算法进行模型检测

定理 2(系统合理性描述) 定理 1 等价为 LTL 公式：$A(f_1 \cup f_2)$，其中：$f_1=\neg(\bigvee_{i,j=1,\cdots,n}^{i \neq j}(s_{3i} \wedge s_{3j}))$；$f_2=s_{41} \wedge s_{42} \wedge \cdots \wedge s_{4n}$。

4.3.1 不动点模型检测算法[7]

(i) 基于 4.2 节生成的 Kripke 结构计算 $A(f_1 \cup f_2)=uZ.\ f_2 \vee (f_1 \wedge AXZ)$；

(ii) 如果 $s_0 \in A(f_1 \cup f_2)$，则系统是合理的，反之不合理。

定理 3(算法复杂度)[9] 不动点模型检测算法的复杂度为 $O(|f| \cdot (|S|+|R|))$，其中，$|f|$ 是 f 子表达式个数，$|S|$ 是状态数，$|R|$ 是状态迁移数。

例 2 以例 1 中的例子为例，用不动点进行模型检测。图 2 生成的 Kripke 结构如图 3 所示。

编码如下：

$$L_s(s_0)=L_s(\langle w,0,0\rangle)=\begin{pmatrix} 0 & 1 & 1 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

$$L_s(s_1)=L_s(\langle 0,0,0\rangle)=\begin{pmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

$$\vdots$$

$$L_s(s_6)=L_s(\langle 1,1,1\rangle)=\begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{pmatrix}$$

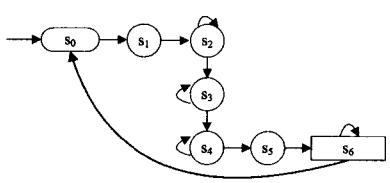

图 3 Kripke 结构

系统合理性公式为 $A(f_1 \cup f_2)$，其中：$f_1=\neg((s_{31} \wedge s_{32}) \vee (s_{32} \wedge s_{33}) \vee (s_{33} \wedge s_{31}))$，$f_2=s_{41} \wedge s_{42} \wedge s_{43}$。不动点计算如下。注意：(ⅰ)计算过程中公式看作满足它的状态之集；(ⅱ)逻辑运算 $\vee$，$\wedge$，$\neg$ 对应集合 $\cup$，$\cap$，$-$ 运算。

$\tau(Z)=uZ.\ f_2 \vee (f_1 \wedge AXZ)$ 的最小不动点计算如下：

$\tau^0(\emptyset)=\emptyset$

$\tau^1(\emptyset)=f_2 \vee (f_1 \wedge AX(\tau^0(\emptyset)))=\{s_6\} \cup (S \cap AX\{\emptyset\})=\{s_6\}$

$\tau^2(\emptyset)=f_2 \vee (f_1 \wedge AX(\tau^1(\emptyset)))=\{s_6\} \cup (S \cap AX\{s_7\})=\{s_5,s_6\}$

$\tau^3(\emptyset)=f_2 \vee (f_1 \wedge AX(\tau^2(\emptyset)))=\{s_6\} \cup (S \cap AX\{s_6,s_7\})=\{s_4,s_5,s_6\}$

$\vdots$

$\tau^6(\emptyset)=f_2 \vee (f_1 \wedge AX(\tau^5(\emptyset)))=\{s_6\} \cup (S \cap AX\{s_2,\cdots,s_6\})=\{s_1,\cdots,s_6\}$

$\tau^7(\emptyset)=f_2 \vee (f_1 \wedge AX(\tau^6(\emptyset)))=\{s_6\} \cup (S \cap AX\{s_1,\cdots,s_6\})=\{s_0,\cdots,s_6\}=S$

$\tau^8(\emptyset)=f_2 \vee (f_1 \wedge AX(\tau^7(\emptyset)))=\{s_6\} \cup (S \cap AXS)=S=\tau^7(\emptyset)$

显然，$\tau^7(\emptyset)=S$ 为最小不动点；因为 $s_0 \in S$，所以该批处理系统是合理的。

结束语 本文提出了单道批处理系统的批处理自动机模型，并且通过模型检测讨论了这个系统的合理性即兼顾公平性与效率性。首先将批处理自动机转化为 Kripke 结构，然后在此基础之上完成了系统的形式化验证，结果表明所建模型是合理的。在今后的研究工作中，可以进一步探索批处理自动机这个模型是否能应用于其他系统的建模，并给出在该模型上的验证算法。

参 考 文 献

[1] Alur R, David L D. The theory of timed automata [J]. Theoretical Computer Science, 1994, 126(2): 183-235

[2] Nerode A, Kohn W. Models for hybrid systems: automata, toplogies, controllability, observability [M]// Grossman R L, et al., eds. Hybrid Systems. New York, Springer-Verlag, 1993: 317-356

[3] David R, Alla H. Petri-Nets and Grafcet: Tools for Modelling Discrete Events Systems [M]. New Jersey: Prentice-Hall, 1992

[4] Demongodin I, Koussoulas N T. Differential Petri nets: Reprecontinuous systems in a discrete- event world [J]. IEEE Trans. on Automatic Control, 1998, 34(4): 573-579

[5] Koutsoukos X D, He K X, Lemmon M D, et al. Timed Petrinets in hybrid systems: Stability and supervisory control [J]. Discrete Event Dynamic Systems: Theory and Applications, 1998, 8(2): 137-173

[6] Tromp L, Benveniste A, Basseville M. Fault detection and isolation in hybrid systems: A Petri- net approach [A]// 14th IFAC World Congress [C]. Beijing, 1999: 79-84

[7] Clarke E M, Grumberg O, Long D E. Model checking [M]. M IT Press, 1999: 61-65

[8] Zhang H. SATO: An Efficient Propositional Prover[C]// Proceedings of the International Conference on Automated Deduction. 1997: 272-275

[9] 边计年. 数字系统设计自动化(第 2 版)[M]. 北京：清华大学出版社，2005：351