

文章编号: 1004-4353(2017)03-0242-05

优先权 Petri 网在电梯运行控制中的应用

严顺, 方贤文*, 刘祥伟

(安徽理工大学 数学与大数据学院, 安徽 淮南 232001)

摘要: 建立具备及时处理实时数据功能的优先权 Petri 网模型, 弥补数据流模型 Petri 网较少考虑数据信息实时性的不足. 首先, 针对需优先处理的实时数据提出变迁优先权及优先权 Petri 网的概念; 其次, 利用 Petri 网的随机性, 通过比较“原则”对模型干扰前后马尔可夫链的区别来确定优先权 Petri 网模型中的 prime 变迁和 spare 变迁, 进一步利用“原则”确定的 T^p 与 T^s 间的对应关系来构造优先权结构并增添 input 库所, 并由此提出建模的算法; 最后, 通过分析案例说明了优先权 Petri 网的优越性及算法的可行性.

关键词: Petri 网; 优先权; 实时数据; 马尔可夫链

中图分类号: TP391.9 **文献标识码:** A

Application of priority Petri net in the controlling of elevator operation

YAN Shun, FANG Xianwen*, LIU Xiangwei

(School of Mathematics and Big Data, Anhui University of Science and Technology, Huainan 232001, China)

Abstract: The priority Petri net are modeled for dealing timely with current data, then improve drawbacks of the Data-flow Petri net about consider currents of information. Firstly, the notion of priority and priority Petri net is presented in terms of prior addressed data; then, according to randomness of Petri net, the prime transition and spare transition are clarified in priority Petri net through "Rule" affect on Markov chain, and construct priority structure and input place by corresponding relation between T^p and T^s reflected in "Rule", such that an algorithm of model is given in the paper; at last, a case study is used to show the advantage of priority Petri net and the feasibility of algorithm.

Keywords: Petri net; priority; current data; Markov chain

0 引言

由于 Petri 网在模拟异步并发行为系统时能够直接展示并发系统的物理结构及动态机理, 使其深受广大建模工作者的青睐; 因此, 如何优化、改进 Petri 网使其能够模拟新型问题、适用新领域已成为各行业工作者关注和研究的对象. 流程模型在模拟实际问题时需要模型语言作为支撑, 如 BPMN、EPCs、UML 和 BPEL^[1-2] 保证了流程

模型形式化程序的逻辑性和有序性. 文献[3]利用时间 Petri 网中变迁分配的参数, 通过对比标准模型和故障模型中对应变迁的时间序列来缩小故障范围, 以便有针对性地处理故障. 文献[4]以紧急救援指挥平台作为模型, 考虑模型中实际活动的时延性及不同结构对模型片段延迟时间的影响, 结合具体的实际参数, 通过替代、修改时延较大的模型结构达到缩短整个模型延迟时间的优化目的, 为指挥救援节省时间. 文献[5]在控制流模

收稿日期: 2017-06-08 * 通信作者: 方贤文(1975—), 男, 教授, 研究方向为 Petri 网和可信软件.

基金项目: 国家自然科学基金资助项目(61572035, 61402011, 61272153); 安徽理工大学研究生创新基金资助项目(2017CX2048); 安徽省优秀青年基金资助项目(ZY290)

型上增加了特殊的数据信息,产生一个能够按照数据信息正常运行的流程模型(即数据流模型),并在此基础上给出了数据流模型的合理性及证明,为依赖于数据的实际问题提供了建模语言的支持.文献[6]从数据量的角度建立了一种满足特定操作符定义规则的新数据结构,使得模型在数据方面具备更高的表现力.考虑 Petri 网模型变迁的发生满足随机性,不论是模型的数据还是结构,只要对模型有影响就会反映在模型的马尔可夫链上.文献[7]利用马尔可夫链中状态概率转移的现象来判断模型的稳定性与可靠性.文献[8]利用开放的 Petri 网,给模型建立了恰当的接口库所及接口的组合关系,允许标识通过接口库所并游离于模型之间,实现了模型相互间的牵制与依赖关系,且体现了数据在模型间的变迁传递关系.本文利用“模型中活动发生的优先顺序应时刻满足用户的主观意愿,模型应能够随时接受用户指令”的设想,构造优先权 Petri 网,特别是时延较长的流程,提高了其运行效率和用户体验.

1 基础知识

定义 1^[9] 四元组 (S,E,T,s_{in}) 为变迁系统 (transition system),其中 S 为有限的非空状态集, s_{in} 为初始状态, E 为有限的事件集, $T \subset S \times E \times S$ 是一个变迁关系,用符号 $s \xrightarrow{e} s'$ 表示.

对于变迁系统,若存在事件 e 或多个事件构成的事件序列状态 σ 使一个状态 s 到达另一个状态 s' ,称状态 s' 是可达的.变迁系统 $TS = (S,E,T,s_{in})$ 满足以下假定:

- 1) 没有自环,即 $\forall (s \xrightarrow{e} s') \in T: s \neq s'$;
- 2) 同一事件不会出现在两组状态之间,即 $\forall (s \xrightarrow{e_1} s'), (s \xrightarrow{e_2} s') \in T: e_1 = e_2$;
- 3) 每个事件都存在一个对应的变迁关系,即 $\forall e \in E: \exists (s \xrightarrow{e} s') \in T$;
- 4) 每个状态都可由初始状态达到,即 $\forall s \in S, \exists \sigma: s_{in} \xrightarrow{\sigma} s$.

根据变迁系统对现实生活案例建模较为简单.以图 1(a) 所示的电梯工作流程模型为例,图中 6 个圆形分别对应电梯在相应楼层停靠的状

态,即 TS 的状态集 S 确定了.由实际调研可知,任意两个不同的状态都是相互可达的,那么 TS 中的 $6 \times 5 \times 2$ 个变迁关系 T 也就确定了.初始状态 s_{in} 假定为电梯某个空闲时间的状态.只要将变迁关系 T 视为 Petri 网的变迁集,状态集 S 视为 Petri 网的库所集,便可得到基于 Petri 网的电梯模型.

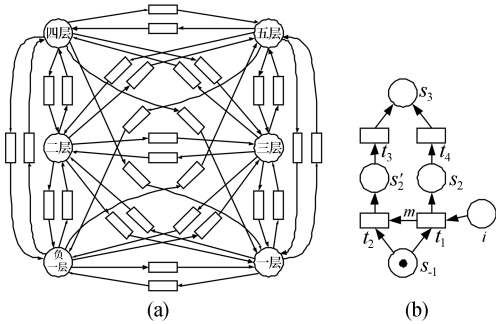


图 1 电梯运行的流程模型

定义 2^[3,10] 对网 $N = (P,T;F,M_0)$ 按如下规则相应地增加时间区间分配函数 I ,得到的网 $TPN = (P,T;F,M_0,I)$ 为时间 Petri 网 (time Petri net):

- 1) 函数 $I: T \rightarrow [R,R + \epsilon]$ 是定义在变迁集 T 上的函数, R 为非负实数;
- 2) 变迁瞬时发生在时间点上,对变迁 $t \in T$,参数 $I(t) = [R_0,R + \epsilon]$ 表示网 TPN 只允许变迁 t 在 $[R_0,R + \epsilon]$ 时间区间内发生,其中 R_0 表示某个时间点;

3) 假定模型中变迁分配的是时间长度而不是区间,则得到相应的时延 Petri 网.

图 2 为灌溉机的工作流程模型,分别用字母 D,I,M,R,T 指代各变迁(变迁 D 指土壤干燥时发生的活动,变迁 M 指土壤湿润时发生的活动,变迁 I 指灌溉活动,变迁 R 用于计时,变迁 T 指检测土壤环境).由于变迁 I 分配了时间区间 $[6:00,6:05]$,因此图 2 为时间 Petri 网.又因为变迁 R 分配了时间长度,所以图 2 也属于时延 Petri 网.

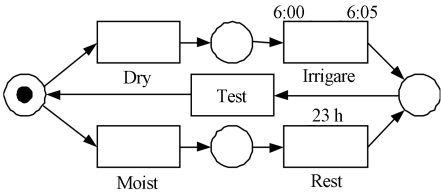


图 2 自动灌溉机的工作流程图

定义 3^[5] 三元组 (P^D, T^D, F^D) 为数据流模型 Petri 网 (data Petri net), 若 $DN = (P^D, T^D, F^D)$ 满足:

1) P^D 和 T^D 分别是有限的数据库所和数据变迁;

2) P^D 和 T^D 是不相交的两个非空集合, 即 $P^D \neq \emptyset, T^D \neq \emptyset$ 且 $P^D \cap T^D = \emptyset$;

3) $F^D \subseteq (P^D \times T^D) \cup (T^D \times P^D)$ 为 DN 的流关系, $(P^D \cup T^D, F^D)$ 是强连通图。

对图 2 所示的模型, D 与 M 为数据变迁, 实现了变迁 I 只有在土壤干燥的时候才有发生权的机能。信息时代数据的实时更新速度越来越快, 然而数据流 Petri 网在及时处理信息方面存在一定的局限性, 因此需要提高数据流 Petri 网的信息处理能力。

2 优先原则的实现

为保证优选方案中的活动能够尽快落实, 首先给出优先权的概念, 并在实际案例中分析其可行性; 其次将优先权转化成合法的 Petri 网语言, 建立可处理实时数据的优先权 Petri 网; 最后将马尔科夫链一步转移概率的概念^[7] 演变成能表达优先原则中优先权问题的表达式, 利用规范的表达式实现优先权 Petri 网的建模过程。

2.1 基于实时数据的优先权

定义 4 给定的网 $N = (P, T; F, M_0)$, 对 $t_i, t_j \in T$ 给出如下的约束, 则称变迁 t_i 对变迁 t_j 在标识 M_0 下有优先权, 其中 $P(M_0)[t_i]$ 表示变迁 t_i 在标识 M_0 下发生的概率:

1) 对 $N = (P, T; F, M_0)$ 中变迁 t_i 和 t_j , 若 $P(M_0)[t_i] + P(M_0)[t_j] = 1$, 会出现变迁 t_j 的发生概率被 t_i 吞噬, 即 $P(M_0)[t_i] = 1$ 且 $P(M_0)[t_j] = 0$, 模型中表示为 $M_0[t_i] \rightarrow M'$, 有 $\neg M'[t_j]$;

2) 若变迁 t_i 由于本身的故障或外界干扰使得 $\neg M_0[t_i]'$, 则 $M[t_j]$, 即变迁 t_i 发生的概率在未被 t_i 吞噬时大于零。

下面构造一个图 3(a) 所示的控制流结构, 不考虑时间分配函数对所有变迁的影响时, 变迁 t_i 和变迁 t_j 在标识 M_0 下的发生概率满足均匀分布。当考虑时延 Petri 网的时间分配函数时, 假定着成黑色的变迁分配到的时间 $time_1 > 0$, 未着色

变迁分配的时间均为无穷小量 ϵ , 其中 $time_2 = \epsilon$, 在标识 $M_1 = M_0[t_0]$ 下, 变迁 t_i 和 t_j 均有发生权。介于 $time_1 > time_2$, 变迁 t_j 的发生先结束, 变迁发生的完成对应着标识的变化, 可知标识状态 $M_0[t_0, t_1]$ 的出现先于 $M_0[t_0, t_2]$, 必有变迁 t_i 先于 t_j 发生。但由于故障或人为干扰使得变迁 t_i 不发生, 变迁 t_j 是有发生权的。优先权可以由多种流程结构实现。图 3(b) 为这类结构的效果图, 它满足定义对相关变迁的规定, 并且具备可受控制的机制。

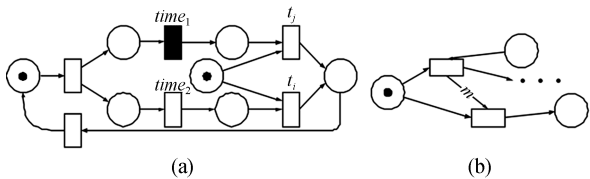


图 3 优先权结构

定义 5 七元组 $(P, I, T, T^p, T^s; F, m)$ 为优先权 Petri 网, 若 $PPN = (P, I, T, T^p, T^s; F)$ 满足:

1) $(P \cup I, T \cup T^p \cup T^s; F)$ 为 Petri 网, 其中 $P \cup I$ 构成库所集, $T \cup T^p \cup T^s$ 构成变迁集, F 为流关系集;

2) T, T^s, T^p 分别代表网中的普通 (common) 变迁集、备用 (spare) 变迁集、优先 (prime) 变迁集;

3) 对应关系 $m: T^p \rightarrow T^s$ 为优先变迁集到备用变迁集的映射, 约束着优先变迁只能吞噬由 m 对应备用变迁的概率, 且 $\forall i \neq j$, 有 $m(T_i^p) \cap m(T_j^p) = \emptyset$;

4) 若 $t \in T^p, m(t) \in T^s$, 则变迁 t 对变迁 $m(t)$ 有优先权;

5) P 为普通库所集 (common-place);

6) I 为输入库所集 (input-place), 用于运行过程中接受实时数据, 满足 $\cup I \cdot = T^p$ 且 $I \cdot = \emptyset$;

7) 任意库所 $p \in P, i \in I$ 有 $M(p) \leq 1, M(i) \leq 1$, 即所有库所的容量均不大于 1。

2.2 算法实现

与传统的数据流模型 Petri 网相比, 优先权 Petri 网能够按照事先设定的原则 (后称“原则”) 对实时数据信息进行处理, 这不仅能够提高效率, 也大大增加了模型的可控性。本文在此结合“原则”将已有的 Petri 网模型演化为可控性强、具备及时处理信息的优先权 Petri 网。首先, 利用马尔

可夫链中一步转移概率的概念将“原则”表述成概率的形式;然后对比原模型马尔可夫链中数值与“原则”约束下新马尔可夫链中数值,将有变动的数值对应的变迁提出来,概率变小的作为 spare 变迁,概率变大的作为 prime 变迁,并进一步为它们匹配合适的对应关系;最后,结合图 3(b) 的优先原则结构为 prime 变迁设计相应的 input 库所. 算法的具体实现步骤如下:

算法: 优先权 Petri 网的建模方法

输入: Petri 网模型,“原则”

输出: 具备处理实时数据能力的优先权 Petri 网模型

1) 对输入的 Petri 网模型,利用随机性计算出模型相关序列的马尔可夫链.

2) 对输入的“原则”,按照一步转移概率的模型进行格式规范,并得出在“原则”下发生序列对应的新马尔可夫链.

3) 对比马尔可夫链,针对概率变大和变小的变迁,建立两个集合 T^P, T^S .

4) 通过对“原则”进行控制变量法处理来查询集合 T^P, T^S 之间的对应关系 m .

5) 对分配到时间长度参数的变迁 $t^s \in T^S$ 进行实际分解,由“当前状态- t^s -目标状态”(一个变迁)细化为“当前状态-变迁-中间状态-变迁-...-变迁-目标状态”(多个变迁)的形式,再通过步骤 4) 中的对应关系确定变迁 t^s 的优先变迁,并按图 3(a) 所示方法建立两者间的优先权关系结构. 进一步,设定相应的输入库所容量值为 0.

6) 检测“原则”对新模型相关序列马尔可夫链的影响情况,若有影响则将新模型作为输入跳到步骤 1),若无影响则将“当前状态-变迁-中间状态-变迁-...-变迁-目标状态”的形式按有无备用变迁重新合并.

7) 将所有 input 库所容量值重新设定为 1,并对所有 input 库所按照“原则”分配相应的面向对象的名称.

3 案例分析

以图 1(a) 的粗糙模型及电梯的顺向优先原则作为输入,利用前面提出的算法输出满足顺向优先原则的优先权 Petri 网模型. 通过对电梯的顺

向优先原则进行初步整理,得到如图 4 所示的“原则”,其具体符号表示的含义见表 1.

```
Rule #up: IF Floor [Max(s1-s2-s3...)] <= Floor [New call]
           ELSE IF Timestamp [New call] < Timestamp [M(s)=1] and
               Floor [s] < Floor [New call]
               THEN DO {"Priority of arriving at Floor [New call]"}
               ELSE DO {"List data in queue"}
           THEN DO {"Priority of sequence:s1-s2-s3..."}
Rule #down: IF Floor [Min(s1-s2-s3...)] >= Floor [New call]
            ELSE IF Timestamp [New call] > Timestamp [M(s)=1] and
                Floor [s] > Floor [New call]
                THEN DO {"Priority of arriving at Floor [New call]"}
                ELSE DO {"List data in queue"}
            THEN DO {"Priority of sequence:s1-s2-s3..."}

```

图 4 顺向优先原则的内容

表 1 “原则”中的符号与定义

符号	含义
$s_1-s_2-s_3\cdots$	原计划状态序列
New call	(t, f)
$t = \text{Timestamp}[\text{New call}]$	New call 诞生的时间戳
$f = \text{Floor}[\text{New call}]$	New call 诞生的楼层数
$f = \text{Floor}[s]$	满足 $M(s) = 1$ 的 s 对应的楼层
$t = \text{Timestamp}[M(s) = 1]$	达到标识 $M(s) = 1$ 的时间戳

如表 2,当电梯处于负一楼状态,用 16 条序列代表电梯可能的运行轨迹,其马尔可夫链上数值表示轨迹上每一步活动发生的概率值. 图 4 中“原则”使得马尔可夫链上数值发生变化,得到 16 条新的马尔可夫链. 根据数值变化的情况,只有相邻楼层间变化的对应变迁发生概率不受“原则”影响. 故这些变迁不属于 spare 变迁 T^S ,其余变迁都属于 T^S . 由算法步骤 4) 进一步判断出 T^P, T^S 间的对应关系 m . 实际上,两个状态之间的变迁关系的实现是需要时间的,即变迁分配到了不为零的时间参数. 以图 1(b) 为例分析 m 的具体构造,〈“负一层”到“三层”〉对应活动的概率为 0.20,而“原则”使它的概率变成 ϵ , 概率被转移到变迁〈“二层”到“三层”〉上,当系统在执行状态由 s_{-1} 变化到 s_3 时,将执行过程分解为状态 s_{-1} 变化到状态 s'_2 再变化到状态 s_3 (确保与实际效果没有冲突). 状态 s_{-1} 到 s_3 表示由“负一层停靠”状态直接到“三层停靠”状态,分解后的执行过程表示“负一层停靠”状态到“途径二层(不停靠)”状态,再到“三层停靠”状态. t_1, t_2, t_3, t_4 分别为状态间的变迁关系, TS 系统容易演化为图 1(b) 的优先权 Petri 网,其中 i 为 input 库所, t_1 为 prime 变迁, t_2 为 spare 变迁, t_3 和 t_4 为 common 变迁, $s_{-1}, s'_2, s_2,$

s_3 为 common 库所. 那么对于优先权 Petri 网, 满足顺向优先原则的实时数据信息 (如二层到三层的呼叫) 便能通过改变库所 i 中标识的方式得到及时合理的受理, 实现 $m(t_1) = t_2$. 用同样的方法将对应关系 m 补充完整, 并设定 input 库所, 使其容量为零, 得到满足“原则”规定的优先权的模型. 按照算法步骤 7) 将 input 库所容量恢复为 1, 并按实际需求将 input 库所命名“—1”“1”“2”等等. 对图 1(a) 所示的原模型, 在顺向优先原则下得到图 5 所示的优先权 Petri 网模型 (该模型只描述了电梯上行过程, 下行过程类似可得). 为了避免流程模型中流关系因交叉而不易识别的困扰, 本文利用开放 Petri 网的可组合性知识^[8] 将优先权 Petri 网分成图 5 中(a) 和(c) 两部分, 为清晰实时数据或用户指令对模型作用的位置, 对 input 库所进行了直观的命名, 如图 5 中(b) 和(c) 中向上的三角形.

表 2 “原则”对马尔可夫链的影响

序列	原模型马尔可夫链	“原则”对应的马尔可夫链
-1,5	(0.2)	(ε)
-1,1,5	(0.2, 0.25)	(0.20, ε)
-1,2,5	(0.2, 0.33)	(ε , ε)
-1,3,5	(0.2, 0.50)	(ε , ε)
-1,4,5	(0.2, 1.00)	(ε , 1.00)
-1,1,2,5	(0.2, 0.25, 0.33)	(0.20, ε , ε)
-1,1,3,5	(0.2, 0.25, 0.50)	(0.20, ε , ε)
-1,1,4,5	(0.2, 0.25, 1.00)	(0.20, ε , 1.00)
-1,2,3,5	(0.2, 0.33, 0.50)	(ε , 0.33, ε)
-1,2,4,5	(0.2, 0.33, 1.00)	(ε , ε , 1.00)
-1,3,4,5	(0.2, 0.50, 1.00)	(ε , 0.50, 1.00)
-1,1,2,3,5	(0.2, 0.25, 0.33, 0.50)	(0.2, 0.25, 0.33, ε)
-1,1,2,4,5	(0.2, 0.25, 0.33, 1.00)	(0.2, 0.25, ε , 1.00)
-1,1,3,4,5	(0.2, 0.25, 0.33, 1.00)	(0.2, ε , 0.33, 1.00)
-1,2,3,4,5	(0.2, 0.33, 0.50, 1.00)	(ε , 0.33, 0.50, 1.00)
-1,1,2,3,4,5	(0.2, 0.25, 0.33, 0.50, 1.00)	(0.2, 0.25, 0.33, 0.50, 1.00)

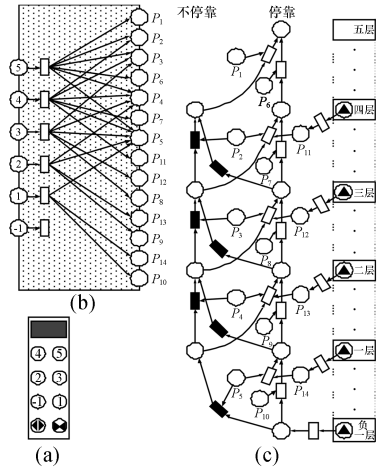


图 5 电梯优先权 Petri 网模型

4 结束语

电梯优先权 Petri 网模型的案例表明, 本文提出的优先权 Petri 网能够及时处理新数据, 接受用户的新指令, 而一般的 Petri 网没有此项功能. 本文所涉及的优先权是从两变迁的直接优先关系入手的, 而现实中存在多个活动相互交叉影响的优先结构. 今后的工作将会针对数据中隐藏的关系进行挖掘, 设计适用性更广泛的优先权 Petri 网结构, 并对“原则”中存在分解性的 spare 变迁做更细处理.

参考文献:

[1] van der A W M P, Pesic M. Decserflow: towards a truly declarative service flow language[J]. Web Services and Formal Methods, 2006,4184:1-23.

[2] van der A W M P, van Hee K M, ter Hofstede A H M, et al. Soundness of workflow nets: classification, decidability, and analysis[J]. Formal Aspects of Computing, 2011,23(3):333-363.

[3] Basile F, Chiacchio P, Coppola J. Real time identification of time Petri net faulty models[C]// IEEE International Conference on Automation Science and Engineering. IEEE, 2015:280-285.

[4] 刘韦光,赵培,赵云胜. 基于 Petri 网的消防应急救援指挥过程建模与性能优化[J]. 安全与环境工程, 2012,19(3):88-92.

[5] Sidorova N, Stahl C, Ka N. Soundness Verification for Conceptual Workflow Nets with Data: Early Detection of Errors with the Most Precision Possible [M]. Amsterdam: Elsevier Science Ltd, 2011.

[6] Couvreur J M, Encrenaz E, Paviot-Adet E, et al. Data decision diagrams for Petri net analysis[C]// International Conference on Applications and Theory of Petri Nets. Springer-Verlag, 2002:101-120.

[7] Singh L, Rajput H, Vinod G, et al. Computing transition probability in Markov chain for early prediction of software reliability[J]. Quality & Reliability Engineering, 2016,32(3):1253-1263.

[8] Bera D, van Hee K M, van der Werf J M. Designing weakly terminating ROS systems[C]// International Conference on Application and Theory of Petri Nets and Concurrency. Springer Berlin Heidelberg, 2012:328-347.

[9] Kalenkova A A, Lomazova I A. Discovery of Cancellation Regions within Process Mining Techniques [M]. Beijing: IOS Press, 2014.

[10] 吴哲辉. Petri 网导论[M]. 北京:机械工业出版社, 2006.