

推理维护系统的 Petri 网模型*

林 闯

陆维明

(国家信息中心信息科学与应用研究所,北京 100045)

(中国科学院数学研究所,北京 100080)

摘要 本文为推理维护系统引入一种 Petri 网模型并对有限非单调理论应用 Petri 网分析方法. 本文提议的模型不但能描述推理维护系统,而且可用网论与网技术分析推理维护. 对于一组给定的非单调规则我们给出一个算法可求得其外延,我们还给出一个算法可发现系统的不一致性. 我们提出的寻找外延的算法要比现有的重构过程更有效,且它是可判定的.

关键词 Petri 网,推理维护系统,非单调推理,逻辑外延,一致性维护.

推理维护系统(Reason Maintenance Systems,简记为 RMS)是适于作维护 AI 系统信念的一类数据库管理系统. 从本质上说 RMS 并不涉及信念的可信或不可信以及信念的含义,它们是要保证 AI 系统具有的全部信念满足某种一致性准则与(或)合理性准则. 其中两条重要的准则是:避免已知信念矛盾和只要有正当理由一个信念就要保留下来. 故推理维护系统可看作是这样的系统,它们为 AI 系统维护以相关信息为参数的断言式数据库,这里的 AI 系统指的是如图 1^[1]所示的一类有效推理机.

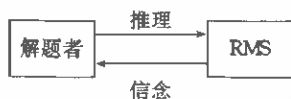


图1 包括一个RMS的一种AI系统

在一个 RMS 中,推理是非单调的推理,因为后加入的信息可能使系统抛弃先前适用的信念.

人们已用不同模型来表示知识推理. 比如,利用具有例外处理的继承网络于 AI 系统的继承推理^[2];语义网络是 AI 中常用的表示机制,它也已用于表示缺省推理^[3];相关性网络已成功地用于为推理维护系统建模^[4]. 最近,我们已用 Petri 网为非单调逻辑建模^[5]. 在文献[4]中,所使用的相关性网络与 Petri 网的结构相象. 本文要使用 Petri 网为推理维护系统建模,因为 Petri 网本身就是描述并行性,不确定性和异步特性的好模型;另外,研究 Petri 网

* 本文 1992-04-11 收到,1992-08-12 定稿

本文工作部分得到国家自然科学基金和中国科学院管理、决策、信息系统开放实验室(MADIS)开放基金的资助. 作者林闯,46岁,副研究员,主要研究领域为 Petri 网应用,人工智能推理,分布系统,人机界面. 陆维明,53岁,研究员,主要研究领域为 Petri 网理论与应用,分布式系统,算法设计与分析.

本文通讯联系人:林闯,北京 100045,西城区三里河路 58 号国家信息中心信息科学与应用研究所

已有一整套理论与技术可用. 利用 Petri 网为 RMS 建模和进行分析使我们有机会用有效而有别于其他的方法来处理 RMS 中的各种问题.

1 基本定义与术语

1.1 Petri 网

由于篇幅的限制, 这里只非形式地简单回顾一下 Petri 网有关术语, 详细定义见[5].

用图形表示时, Petri 网是有向偶图, 横表示变迁 T 元素, 圆圈表示位置 S 元素, 横与圆圈间可能有有向弧 F 元素连接, S 元素可含标志表示为标识 M , 标志个数为 $M(s)$. 当一个变迁 t 的所有输入位置都至少含有一个标志时, t 就是使能的, t 的实施后, 它的所有输入位置的标志个数减“1”, 所有输出位置的标志个数加“1”, 形成一个新的标识 M . 变迁连续的实施可形成变迁实施序列. 当存在标识 M 及从 M 起的实施序列 β 且回至 M , 则 β 的 Parikh 映射 $\bar{\beta}$ 就是这个 Petri 网的 T 不变量 X . 一个实施序列 β 的支持就是它所包含的变迁集, 记为 $|\beta|$.

1.2 非单调推理

常见的 AI 系统会跳跃式推理取得结论, 因为通常不会有足够多的信息使推理取得结论一定正确, 故它的推理总是基于一些简化了的假设前题的. 这类 AI 系统推理称之为非单调的, 因为后增加的信息可能会使该系统先前得到的结果不再成立.

本文我们将大致地采用 Goodwin 的非单调规则的定义方法^[6]. 一个非单调理论是有序对 $\langle J, P \rangle$, 其中 P 是前提集, J 是非单调规则集. 而一条非单调规则是由有限命题集组成的三元式 $\langle M, N, C \rangle$, 式中 M 是单调支持, N 是非单调支持, 而 C 为结论. 其含义是若 M 中命题均为真, N 中命题为假或不知道, 则可推出 C 为真. 三元式中 M, N 与 C 均可能为空集. 若 N 为空, 规则是单调的. 这里我们讨论的是命题逻辑而不是一阶逻辑, 要作进一步了解的读者可以参考关于非单调推理第二届国际会议的会议论文集^[7].

1.3 用作非单调推理的一种 Petri 网模型

为了切合非单调推理的性质, 我们在 Petri 网模型中引入禁止弧 (inhibitor arcs) 用以表示非单调支持. 一条禁止弧从位置引向变迁, 它不以箭头表示方向而代之以小圆圈表示禁止. 含禁止弧 Petri 网的变迁的使能充要条件增加, 有禁止弧引至此变迁引出禁止弧的位置中须设有标志; 使能变迁的实施并不减少有禁止弧引向它的输入位置中标志个数.

一个变迁 t 称为源变迁当且仅当它没有输入位置, 而一个变迁 t 称为漏变迁当且仅当它没有输出位置. 源变迁意为它的使能、实施使标志增加是无条件的, 相反漏变迁意为它实施总是使标志减少.

将非单调理论转换为 Petri 网使用下列规则:

(1) 对每条非单调规则引入一个变迁. 支持与结论分别由此变迁的输入位置集与输出位置集表示. 把用作非单调支持的位置称为非单调位置, 也可称为禁止位置.

(2) 为每一前提引进一个源变迁.

(3) 以有向弧联结单调支持与结论. 用禁止弧连接非单调支持.

具体的转换算法详见文献[5].

例 1.1:

给定一个非单调理论 $\langle J, P \rangle$ 如下:

$$P = \{a, b\},$$

$$J = \{\langle \{a\}, \{b\}, \{c\} \rangle\}$$

显然, J 中只含一条非单调规则 $\langle M, N, C \rangle$, 其中 $M = \{a\}$, $N = \{b\}, C = \{c\}$.

利用转换算法可得 Petri 网结构示于图 2.

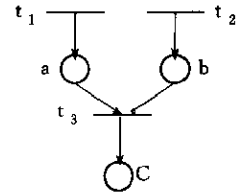


图2 例1.1转换得的Petri网

2 变迁、实施序列间的关系

为了讨论非单调推理的推理机制,引入变迁直连概念.若一位置有输入变迁 t_i 、输出变迁 t_j ; 则称 t_i 与 t_j 直连,表示为 $t_i < t_j$. 如果定义非单调推理的 Petri 网无环,直连关系的传递闭包定义了变迁集合 T 上的偏序关系.

在 t_i 与 t_j 由正常弧直连时, t_i 的实施使 t_j 有可能实施; 在 t_i 与 t_j 有禁止弧直连时, t_i 与 t_j 就称为有互斥关系.

例 2.1:

考察图 2 中的网模型. 在此 Petri 网中, 变迁 t_2 与 t_3 有直连, $t_2 < t_3$. 很明显它们是互斥的, 即 t_2 实施使 t_3 不可能实施. 这里称 t_2 为禁止变迁, t_3 为有禁变迁. t_1 与 t_3 有直连, 很明显 t_1 实施使 t_3 有可能实施.

现定义死变迁. 记从初始标识 M_0 出发的所有可能的实施序列为 $L(M_0)$. 某变迁 t 称为是死变迁当且仅当 $L(M_0)$ 的任何元素中均不含 t . 当 t_i 与 t_j (设 $t_i < t_j$) 直连、互斥, 对于局部推理而言, 知道禁止变迁 t_i 是否死了颇重要. 若 t_i 是死的, 则与 t_i 有关的非单调支持不会影响到 t_j , 即可在网模型中将 t_i 与 t_j 间的禁止弧去掉. 若 t_i 不是死的, 则 t_j 不应在有效推理实施序列中出现, 即可将 t_j 及其连接的弧均删除. 用这样的办法可以简化非单调推理 Petri 网模型中外延的计算.

逻辑演绎可视作一个实施序列, 从一个变迁至后继变迁经由激发式直连. 演绎可起始于源变迁.

在实施序列与 Sandewall 方法^[8]的信息状态表示的赋值序列之间有些区别. Petri 网有优点是因为它很适于描述状态序列. 当一个对象在新状态中的事实导出时, 它在旧状态中的事实立即被否定. 相反, 在赋值函数方法中, 赋值目的是增加系统的知识, 即将新的事实加到知识库中去. 在赋值方法中要设法找出一个赋值为不动点; 以 Petri 网为推理模型时要找出一个实施序列为不动点.

记 q 为实施序列, \bar{q} 为 q 的 Parikh 映射及 $|q|$ 为 q 的支持集. 一个实施序列 $q \in L(M_0)$ 的支持称为是最大的, 当且仅当对所有的 $t \in (T - |q|)$, t 在 M_0 不可实施, 这里 $M_0[|q| > M_i$.

一个实施序列 $q \in L(M_0)$ 的支持称为是一致的当且仅当 $|q|$ 中不存在处于互斥关系的变迁.

最大与一致这两种性质在寻找作为不动点的实施序列时是必须的.

3 寻找非单调理论外延的算法

在由 Petri 网为非单调理论建模后, 非单调理论的不动点表示为最大与一致的实施序

列的支持, 而外延由含于该支持内每个变迁的全部输出命题组成.

至此, 我们可以介绍在非单调理论的网模型中如何计算外延问题.

算法 1.

输入: 一个非单调理论的 Petri 网模型;

输出: 此非单调理论的所有外延 $E_i (i \geq 1)$;

- (1) 构造 Petri 网的覆盖树;
- (2) 在覆盖树上找实施序列的所有最大支持, 放入 $E_i (i \geq 1)$;
- (3) 查验各个 E_i 是否一致, 将不一致的删去;
- (4) 停机.

含有禁止弧 Petri 网的覆盖标识概念有别于没有包含禁止弧的 Petri 网. 非单调位置中标志的增加不支持标识的覆盖性; 相反, 减少标志支持标识的覆盖性. 引入记号 ω 意为无穷; 它有性质 $\omega > n, \omega \pm n = \omega, \omega \geq \omega$, 这里 n 是任意整数.

算法 1.1 给出构造含有禁止弧 Petri 网的覆盖树.

算法 1.1.

输入: 一个非单调理论的 Petri 网模型.

输出: 此 Petri 网的覆盖树 T_r .

步(1)把初始标识 M_0 记为树根, 加上标签‘新’;

步(2)While 存在‘新’标识 do

步(2.1)选一个‘新’标识 M ;

步(2.2)if M 与从根至 M 路上一个标识相同

then M 标签改为‘旧’ and go to next;

步(2.3)if 所有变迁于 M 不使能

then M 标签改为‘死’ and go to next;

步(2.4)for 每个于 M 使能的变迁 t do

步(2.4.1)从 M 实施 t 得标识 M' (注: 若有 p 使计算 $M'(p) = \omega - n$, 不要写为 ω , 在这一步仍写为 $\omega - n$);

步(2.4.2) if 在从根至 M 的路上有 M'' 满足, 对每个 $p: M'(p) \geq M''(p)$, 且下列两个条件之一满足:

(1)若 $M' = M''$, 则一定有 $p, M'(p) = \omega - n$.

(2) $M' > M''$ 且从 M'' 至 M 路上有有禁变迁 t' 或 t 是有禁变迁则对 t' (或 t) 的非单调位 $p, M'(p) = M''(p)$.

then 称 M' 覆盖 M'' 且对 $M'(p) > M''(p)$ 者置 $M'(p)$ 为 ω , 对 $M'(p) = \omega - n$ 者置 $M'(p)$ 为 0;

步(2.4.3)若存在 $M'(p) = \omega - n$ (注: M' 是正常标识), 则对 $M'(p) = \omega - n$ 者置 $M'(p)$ 为 ω ,

步(2.4.4)引入 M' 为节点, 从 M 至 M' 画一条标记为 t 的弧并对 M' 加标签‘新’.

od

end of for;

next; od

end of while.

为更清楚地理解算法 1.1,考察图 3 中例子.

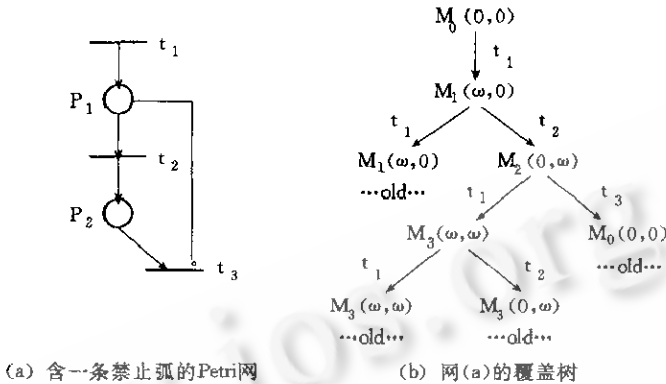


图3

于初始标识 $M_0 = (0, 0)$, t_1 可实施. 实施 t_1 得覆盖 M_0 的标识 $M'_1 = (1, 0)$. 由于 $M'_1(P_1) > M_0(P_1)$, 在覆盖树中引入结点 $M_1 = (\omega, 0)$, 接着实施 t_1 得标识 $(\omega, 0)$, 它与 M_1 相同; 实施 t_2 得 $M'_2 = (\omega - 1, 1)$ 覆盖 $M_1 = (\omega, 0)$, 按算法 1.1 分别以 0 与 ω 代替 $\omega - 1$ 与 1 得新标识为 $M_2 = (0, \omega)$. 于 M_2 , t_1 和 t_3 是使能的. 实施 t_1 得覆盖 M_2 的 $M'_3 = (1, \omega)$, 按规则(算法 1.1)引入 $M_3 = (\omega, \omega)$; 实施 t_3 得 $M_0 = (0, 0)$ 与根相同. 于 M_3 , t_1 和 t_2 是使能的. 在 M_3 实施 t_1 得标识 (ω, ω) 与 M_3 相同. 在 M_3 实施 t_2 得覆盖 M_3 的 $(\omega - 1, \omega)$; 按算法 1.1 规则 $\omega - 1$ 被 0 代替得节点 $(0, \omega)$, 和 M_2 相同. 至此, 覆盖树求得, 如示于图 3(b).

从 Petri 网的覆盖树可得实施序列的所有最大支持, 就是在其覆盖图上查找最长路, 其详细算法见文献[12]. 查验一个实施序列的最大支持是否为一组是很简单的事, 只检查有否互斥的变迁即可.

4 一致性维护

有时, 一个 AI 系统由于下列三个原因之一或许会推论至一个明显的矛盾: (1) 前提不一致; (2) 缺省推理或许会导出与观察的矛盾; (3) 推理过程引入的几种假设可能会导致互不相容.

这一节简要引入一种 Petri 网模型以示不一致性的存在, 并探讨含有一致性维护过程的推理维护系统怎样用以支持不一致性的搜寻.

4.1 四值逻辑的 Petri 网模型

为了构造 Petri 网模型以示存在不一致性须要引入四值逻辑, 详见文献[9]. 这种 Petri 网模型中, 其标志和弧上都注有四值 $\{u, t, f, k\}$.

与经典两值逻辑的直观相关, t 与 f 分别为“真”与“假”. u 表示无定义状态, 直观上称为“不知道”, 而 k 为“矛盾”. 四真值集组成格, u 是底元素, k 是顶元素. 定义为: $\neg(t) = f$, $\neg(f) = t$, $\neg(k) = k$, $\neg(u) = u$.

在这种 Petri 网模型中,一般具有 u (即不知道)的标志并不显示表示. 在一个位置无标志时,我们认为该位置含有值为 u 的标志. 具有值为 k 的标志可以由值为 t 和值为 f 的两个标志合成. 反之,一个值为 k 的标志也可分解成一个值为 t 和一个值为 f 的两个标志. 当具有禁止弧输入的变迁实施,禁止弧输入位置中的标志值变为 f . 用此模型,在一个标志值为 k 时,就表示与此标志相关的命题存在不一致性,即一个命题的真假值同时存在.

4.2 不一致性的消除

为了解决推理系统中的不一致性,首先要在我们的 Petri 网模型知道哪个位置可能含有值为 k 的标志. 我们可以在必要检验的位置上加一个漏变迁并将连接弧标上值 k . 而后检查是否有实施变迁序列可重复生成空标识与实施这个附加上去的漏变迁. 若有这样的实施序列,则对应命题是不一致的. 最后,我们要找出这样一种实施序列的支持,它包含漏变迁且若有任意变迁从此支持中挪走则它就不包含漏变迁. 它被称为可产生不一致性的实施序列的最小支持. 一致性维护程序的问题是如何在会产生不一致的序列的最小支持中使一个变迁不能实施. 下面的算法用以查验矛盾与查找出会产生矛盾的实施序列的最小支持.

算法 2.

输入: 从一个具有四值注释的非单调理论(简称 FANMT)转换得的 Petri 网模型 PN .

输出: 可产生不一致性的实施序列的一切最小支持与所有不一致位置.

步(1) PN 外延中每个位置加上漏变迁,对应弧上标上 k ,得到一个新 Petri 网 PN' .

步(2)按算法 1.1 为 PN' 构造一个覆盖树 T_r .

步(3)从 T_r 找出产生不一致性的实施序列的一切最小支持与所有不一致位置.

因为算法 2 的步(1)与步(2)都是清楚的,这里只要细化步(3)即可.

算法 3.

输入: 一个 Petri 网的覆盖树 T_r .

输出: 产生从初始标识与实施漏变迁的实施序列的所有最小支持与所有不一致位置.

步(1)对所有的“旧”结点(恒同的叶结点)加上标签“未查验”.

步(2)for 每个“未查验”结点 M do

步(2.1) if $M \neq M_0$ (M_0 是初始标识,为空) then goto 步(2.4);

步(2.2) if $M = M_0$ and 弧标 t 到 M 不是漏变迁 then goto 步(2.4);

步(2.3)引入一个空集 E_t ,并搜集根至 M 路上的弧标,置入 E_t ,列出弧标 t 的输入位置 P ;

步(2.4)改 M 标签为“已查”.

od end of for.

当我们知道模型中有矛盾时,要修改模型以消除矛盾. 下面以例子说明一致性维护程序工作起来是怎样的.

例 4.1:

考察示于图 4 中的 Petri 网模型. 若不考虑模型中弧标,按算法 1,它有一个外延 $\{a, d, g\}$.

按弧上赋值,此模型是不一致的. 按算法 2,找出可产生矛盾位置 g 实施序列的支持 $|\sigma| = \{t_1, t_2, t_3, t_4\}$,但 g 在外延中,即此系统被迫相信一条矛盾公式. 使 $|\sigma|$ 中的一个变迁不使

能可排除这个矛盾. 为了从外延中把 g 挪走有二个办法可选用: 一是取消前题 $\{a, d\}$ 中一个前题(即, 使 t_1 或 t_3 不能实施); 一是使非单调支持 $\{b, c, e\}$ 中一个为真(即, 使 t_2 或 t_4 不能实施). 一般说, 由非单调规则得到的结论在一种假设下成立. 若某种附加信息(前题, 规则等)可使非单调支持之一为真, 则假设要放弃. 故图 4 中的非单调支持比前题更可疑. 不过, 面对矛盾如何修改还没有大家一致同意的理论可循, 当然一致性维护程序总要先选好维护方案. 我们已示例说明如何找到矛盾以及如何将矛盾从外延中排除.

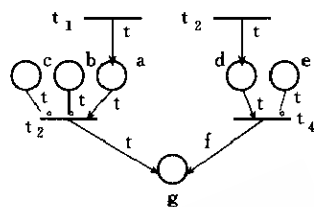


图4 有一处矛盾的Petri网模型

5 结 论

本文中我们已展示 Petri 网是一种有用的形式工具, 并且为非单调推理提供了有力的分析.

本文发展了文献[5]的工作, 我们的主要贡献是: (1) 为含有禁止弧的 Petri 网提供计算覆盖树的新算法. (2) 为 FANMT 的 Petri 网模型提供找出其中矛盾处的算法.

这些算法可以用于说明与分析推理维护系统; 它们也比现有算法^[4]更有效. 本文中实施序列支持的计算基于含禁止弧 Petri 网的覆盖树. 一般说, 构造覆盖树耗费指数时间与空间, 当然可达问题还是可判定的. 关于找实施序列支持的算法有望于将来作进一步改善.

参考文献

- 1 Smith B, Kelleher G. Workshop on reason maintenance systems and their applications. England: Ellis Horwood Series in Artificial Intelligence, 1988.
- 2 Touretzky D S. The mathematics of inheritance systems. Pitman, London, 1986.
- 3 Shastri L. Default reasoning in demantic network: a formalization of recognition and inheritance. *Artificial Intelligence*, 1989(39): 283-355.
- 4 Reifrank M. Lecture notes on reason maintenance systems. *Advanced Reasoning Methods, Report INF2 ArM-5-88*, 1988.
- 5 Lin C, Murata T. A Petri net model for inference in nonmonotonic logic. The University of Illinois at Chicago, Report No. UIC-EECS-90-4, 1990.
- 6 Goodwin J. WATSON: a dependency directed inference system. In: *Proceedings of the American Association for Artificial Intelligence Nonmonotonic Reasoning Workshop*, 1984.
- 7 2nd International Workshop on Nonmonotonic Reasoning, Grassan, FRG, June 1988. In: Reifrank, M. *et al.*, (eds), *Lecture Notes in Artificial Intelligence* 346.
- 8 Sandewall E. A functional approach to nonmonotonic logic. *Computer Intelligence*, 1985, 1: 80-87.
- 9 Murata T *et al.* A Petri net model for reasoning in the presence of inconsistency. The University of Illinois at Chicago, Report No. UIC-EECS-89-3 (To appear in the *IEEE Trans. on Knowledge and Data Engineering*).

A REASON MAINTENANCE SYSTEM MODEL USING PETRI NETS

Lin Chuang

(Institute of Information Science and Application, State Information Center, Beijing 100045)

Lu Weiming

(Institute of Mathematics, The Chinese Academy of Sciences, Beijing 100080)

Abstract This paper introduces a Petri net model for a reason maintenance system and applies Petri net analysis methods to a finite nonmonotonic theory. The model proposed in this paper can not only specify the reason maintenance system, but also analyse the system based upon the net theory and techniques. It is important that the authors derive an algorithm to find the extensions for a given set of nonmonotonic rules and an algorithm to find out inconsistency in a system. Their algorithm to find extensions is considered to be more efficient than the existing reconstruction procedure, and it is decidable.

Key words Petri nets, reason maintenance systems, nonmonotonic inference, logical extensions, consistency maintenance.