一种模糊Petri网的逆向知识推理方法设计实现

杨劲松 凌培亮
(同济大学机械工程学院 上海 200092)

摘 要 模糊Petri网是模糊产生式知识表示和推理的理想工具。针对基于模糊产生式规则的知识库,在已知决策目标的前提下,设计了该知识库的模糊Petri网模型及基于递归的逆向知识推理方法,并以实例对该方法进行了验证。对于任意指定的库所,通过该方法可以确定其模糊托肯值,即对应命题的模糊真值。该方法的逻辑表达力强,利于计算机实现,而且其逆向推理策略能有效减少计算空间,使计算在一个复杂的模糊Petri网系统的子系统中进行,提高了计算效率。

关键词 FPN,DSS,知识表示
中图法分类号 TP301 **文献标识码** A

Design and Implement on a Backward Reasoning Algorithm Based on Fuzzy Petri Net

YANG Jin-song LING Pei-liang
(College of Mechanical Engineering, Tongji University, Shanghai 200092, China)

Abstract Fuzzy Petri net is a ideal tool for fuzzy production knowledge representation and reasoning. This paper designed the FPN model for fuzzy production rule-based knowledge base and implemented a backward fuzzy reasoning algorithm based on FPN through the recursive. The algorithm was verified by examples in this paper. Using the algorithm, one can calculate the tokens of any appointed places which correspond to the true fuzzy values of the relevant propositions. The algorithm is strong in logic expression and easy to implement in computer system. The proposed algorithm can effectively reduce the computing space through transforming a large and complex system based on FPN into a small subsystem relating to the problems and improve the computational efficiency.

Keywords FPN, DSS, Knowledge representation

大多数决策支持系统(Decision Support System,DSS)中的问题决策表现为一种不确定情况下的决策过程,这类的核心是规则库。规则库中包含着大量DSS的专家知识。这种带有模糊性的知识无法用数学公式精确地表示,而具有图形描述能力的Petri网和模糊集合理论结合后,成为这类模糊知识的表示和推理的理想工具。因此,本文在产生式知识表示规则库的基础上,根据决策规则的模糊性,结合具有图形描述能力的模糊Petri网(Fuzzy Petri Net,FPN),探索一种基于模糊产生式规则的知识库的知识表示FPN模型和基于递归的逆向知识推理算法,以改善基于模糊产生式规则的知识表示方法和推理过程。

1 产生式知识表示的FPN模型与基本运算规则

1962年,德国科学家C. A. Petri博士在其博士论文中首次提出Petri网模型以来,Petri网在各种领域得到了广泛的应用和推广,尤其是在那些并行处理和基于图形的界面设计方面更能显示其特点和优越性。FPN是基于模糊产生式规则的知识系统的良好建模工具,它结合了Petri网的图形描述能力,使得知识的表示简单、清晰,又表现出了知识库系统中规则之间的结构化特性;它还具有模糊系统的模糊推理能力,便于知识的分析、推理、测试,以及决策支持等。

定义1 令R是模糊产生式规则集,$R=\{R_1,R_2,\cdots,R_n\}$。对于其中每个规则R的定义为:IF d_j THEN d_k($CF=\mu_i$),其中d_j和d_k为命题,它们的真值是介于0和1之间的实数;μ_i为规则的模糊因子(Certainty Factor,CF),$\mu_i\in[0,1]$。μ_i越接近1,规则R_i越可信。

定义2 按表示模糊产生式规则的FPN模型来表示一个基于规则的系统。根据文献[1]的定义,一个FPN被定义为八元组:

$FPN=(P,T,D,I,O,f,\alpha,\beta)$,这里$P=\{p_1,p_2,\cdots,p_n\}$是一个有限库所集合;

$T=\{t_1,t_2,\cdots,t_m\}$是一个有限变迁集合;

$D=\{d_1,d_2,\cdots,d_n\}$是一个有限命题集合;

$|P|=|D|$;

$I:T\to P^\infty$是一个输入函数,映射一个变迁到它的输入库所集合;

$O:T\to P^\infty$是一个输出函数,映射一个变迁到它的输出库所集合;

到稿日期:2009-05-22 返修日期:2009-07-28

杨劲松(1976—),男,博士生,讲师,主要研究领域为决策支持系统,E-mail:yangjinsong@tongji.edu.cn;凌培亮(1949—),男,博士,教授,主要研究领域为决策支持系统、Agent技术、人工智能。

$f: T \rightarrow [0,1]$是一个函数，映射变迁到一个从 0～1 的数值，用来表示变迁对应的推理规则的置信度(CF)；

$\alpha: P \rightarrow [0,1]$ 是一个函数，映射库所到一个从 0～1 的数值，用来表示该库所对应的命题成立的真实度；

$\beta: P \rightarrow D$ 是一个函数，映射库所对应的命题。

当用 FPN 进行模糊推理时，一个库所表示一个命题，一个变迁表示一条模糊推理规则，即两个命题之间的因果关系。托肯值代表命题的真实度。每个变迁有一个置信度，表示推理规则的可信度。

FPN 的基本推理规则的形式分以下 4 类：

类型 1 IF d_j THEN d_k ($CF=\mu_i$)，其中 d_j 和 d_k 为命题，推理过程可以建模为图 1 所示，其中命题 d_j，d_k 用库所 p_j 和 p_k 表示，命题 d_j 的真实度为 y_j。命题之间的因果关系用变迁 t_j 表示，它的置信度为 μ_j。当变迁触发后命题 d_k 的真实度为 $y_j \times \mu_j$。

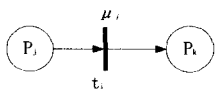

图 1　类型 1 的 FPN 模型

类型 2 IF d_{j1} AND d_{j2} AND…AND d_{jm} THEN d_k ($CF=\mu_i$)，其中 $d_{j1}, d_{j2}, \cdots, d_{jm}$ 为命题，推理过程可以建模为图 2 所示。当变迁触发后，命题 d_k 的真实度为 $\min(y_{j1}, y_{j2}, \cdots, y_{jm}) \times \mu_i$。

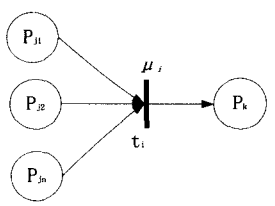

图 2　类型 2 的 FPN 模型

类型 3 IF d_{j1} OR d_{j2} OR …OR d_{jm} THEN d_k ($CF=\mu_i$)，其中 $d_{j1}, d_{j2}, \cdots, d_{jm}$ 为命题，推理过程可以建模为图 3 所示。当变迁触发后，命题 d_k 的真实度为 $\max(y_{j1} \times \mu_i, y_{j2} \times \mu_i, \cdots, y_{jm} \times \mu_i)$。

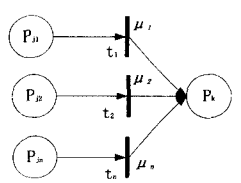

图 3　类型 3 的 FPN 模型

类型 4 IF d_j THEN d_{k1} AND d_{k2} AND … AND d_{kn} ($CF=\mu_i$)，推理过程可以建模为图 4 所示。当变迁触发后，命题 d_{ki} 的真实度为 $y_j \times \mu_i$。

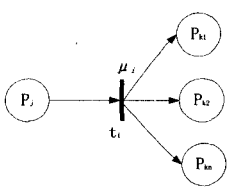

图 4　类型 4 的 FPN 模型

2　知识的推理

所谓推理是指按照某种策略从已知事实出发去推出结论的过程。推理的实质就是搜索，不同的推理方法是采用了不同的搜索方式来找到解答的。搜索空间(或称解答空间)的大小是影响推理效率的因素之一。如果缩小了搜索空间，将有助于提高推理速度。对于具体问题，与解有关的状态空间一般仅是全部状态空间的一部分。因此，在推理过程中，没有必要生成和保存该问题的全部状态空间，只要能够生成和保存与解有关的那部分状态空间即可。

本文的主要思想是在已知决策目标的前提下，去除与决策目标无关的规则或条件，对模糊 Petri 网进行化简，缩小问题空间，在后面处理与推理结论有关的信息时，可以直接在这个子空间中进行，这在提高大型知识库处理效率时尤为明显，不仅容易满足实时性的要求，也能满足决策支持的要求。

在本文给出的算法中，假设条件是充分的。首先本文需要给出几个算法描述需要的定义：结点、目标库所集、种子库所集、变迁的紧后库所集(Set of nearest backward places，SNBP)。

定义 3(节点)　节点 n_i 可以形式化为一个五元组(p_i，$\alpha(p_i)$，$\{t_{pi}\}$，$bool$，$depth$)，$\alpha(p_i)$为节点的真值；$\{t_{pi}\}$为 p_i 所有输入变迁集合；$bool$ 为布尔变量，$bool \in \{0,1\}$，当 $\alpha(p_i)$已知时，$bool=1$，当 $\alpha(p_i)$未知时，$bool=0$。目标库所表示的节点为目标节点，种子库所表示的节点为种子节点。$depth$ 为非负整数，记录该库所所在的最深层级，从目标库所出发逆向遍历，每通过一次变迁，层级增加 1。由于 FPN 的复杂性，同一个库所可能在不同的层级出现，在逆向搜索时，每遍历到某库所一次，即更新其所在的层级。为方便计算，目标库所层级为 1，种子库所不记录层级，$depth$ 记为 0。

定义 4(目标库所集)　指本次推理目标库所集，记为 $OP=\{p_j, \cdots, p_k\}$，记$|OP|$为目标库所个数。p_j 的模糊托肯值 $\alpha(p_j)$对应命题 d_j 的真值，$\{\alpha(p_j), \cdots, \alpha(p_k)\}$即要推理 p_j 的结果。

定义 5(种子库所集)　是本次推理的起点，记为 $SP=\{p_a, \cdots, p_e\}$，记$|SP|$为目标库所个数。种子库所的模糊托肯值为已知条件。

定义 6(SNBP)　变迁 t_i 的紧后库所集为 SNBP(t_i)，若在 FPN 中存在 p_i 到 t_i 的弧关系，即 $p_i \rightarrow t_i$，且路径中不包括其他任何库所和变迁，则所有这样的库所 p_i 的集合为 SNBP(t_i)，| SNBP(t_i)|为变迁 t_i 的紧后库所个数。

算法步骤：

步骤 1　按目标库所集建立目标的节点集。$p_i \in OP$，以 p_i 为例，由于 $\alpha(p_i)$未知，节点表示为(p_i，$\alpha(p_i)$，$\{t_{pi}\}$，0，1)。

步骤 2　以节点(p_i，$\alpha(p_i)$，$\{t_{pi}\}$，0，1)为例，遍历 p_i 的每一个输入变迁 $t_n \in \{t_{pi}\}$，寻找 t_n 的 SNBP(t_n)，并将每一个库所以节点表示，若新增节点是种子节点，则该路径搜索结束；若新增节点不是种子节点，则以对此节点实施步骤 2，直至发现种子节点。由于假设种子库所是充分的，因此以目标库所的遍历总可以找到种子节点结束。在实际系统中，由于网的复杂性，可以限定遍历深度，当遍历过深时，可终止搜索，定义该路径非法，且删除该路径上有关节点和前置节点的输入变迁，具体方法本文不深入研究。

步骤 3　从原 Petri 网系统中保留算法遍历到有效路径的节点库所及其变迁，保留种子节点的库所(不必保留种子库所的输入变迁)。这个 petri 子网是 $\alpha(p_i)$的求解空间。

步骤 4　求解 $\alpha(p_i)$，记 MaxD 为所有节点层级的最大深度值，先计算第 i=MaxD 层的未知节点真值：令 $n_{ki}=(p_{ki},\alpha(p_{ki}),\{t_{ki}\},0,i)$ 为第 i=MaxD 层的一未知节点，由于 $\{t_{ki}\}$ 的输入库所均为种子节点，按定义 2 的 FPN 的基本推理规则的类型 1、类型 2、类型 3 可完成节点真值计算；计算完所有第 i=MaxD 层的未知节点真值，按上述方法计算第 i=MaxD−1 层的未知节点真值，直至 i=1 时计算出目标节点真值 $\alpha(p_i)$。

步骤 5　重复步骤 2 到步骤 4，完成所有目标节点真值计算。

算法的核心在于设计一个递归算法实现步骤 2，递归函数 ransackpetri (N:Note;)描写如下：

```
procedure ransackpetri (N:Note;);
struct Note //定义节点变量
{
char P;
float α;
char T[];
boolean b;
integer depth;
integer Tag;
};
Integer Dep
Begin
  Dep=1
  For i=1 to |N. T| step 1 //遍历 N 节点的每一个输入变迁的循环
    For j=1 to |SNBP(N. T[i])| step 1 //准备遍历 N 节点 ti 的紧后库所的循环
      Dep=Dep+1 //路径深度+1
      If SNBP(N. T[i])[j]. b<>True then //如果 N 节点 ti 的该紧后库所不是种子库所
        SNBP(N. T[i])[j]. depth = Dep //N 节点 ti 的紧后库所路径深度赋值
        ransackpetri (SNBP)(N. T[i])[j] //对 N 节点 ti 的该紧后库所递归
      End if
    End For
  End For
End;
```

3　实例

本节通过一个例子来说明前面的递归算法，设有如下模糊产生式规则的知识库：

R1：IF s1 THEN s7 ($\mu2=0.9$)

R2：IF s2 AND s3 THEN s5 ($\mu1=0.95$)

R3：IF s3 AND s4 THEN s6 ($\mu3=0.95$)

R4：IF s4 THEN s10 ($\mu5=0.9$)

R5：IF s5 THEN s8 ($\mu4=0.9$)

R6：IF s6 AND s8 THEN s10 ($\mu7=0.85$)

R7：IF s7 AND s8 THEN s9 ($\mu6=0.95$)

该知识库对应的 FPN 如图 5 所示。

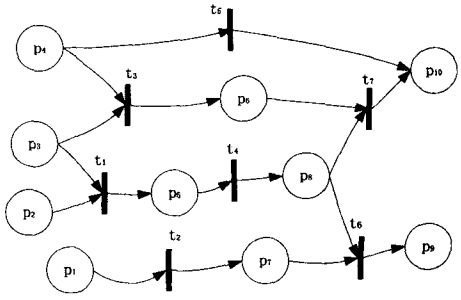

图 5　知识库对应的 FPN

命题 s1，s2，s3，s4 的真实度分别为 0.8，0.8，0.9，0.7。拟求 s9，s10 的真实度。

步骤 1　按目标库所集建立目标的节点集。P_9，p_{10} 节点表示为 $(p_9,\alpha(p_9),\{t_6\},0,1)$，$(p_{10},\alpha(p_{10}),\{t_5,t_7\},0,1)$。

步骤 2　以节点 p_{10} $(p_{10},\alpha(p_{10}),\{t_5,t_7\},0,1)$ 为例，寻找 SNBP(t_5) 和 SNBP(t_7)，并将每一个库所以节点表示，$p_4\in$ SNBP(t_5)，由于 p_4 是种子库所，其输入变迁集用{−}表示，路径深度不计(数值为 0，但不代表其路径深度)，节点 p_4 记为 $(p_4,0.7,\{-\},1,0)$；p_6，$p_8\in$ SNBP(t_7)，节点 p_6 记为 $(p_6,\alpha(p_6),\{t_3\},0,2)$，节点 p_8 记为 $(p_8,\alpha(p_8),\{t_4\},0,2)$；寻找 SNBP($t_3$) 和 SNBP($t_4$)，$p_3$，$p_4\in$ SNBP(t_3)，由于 p_3，p_4 是种子库所，节点 p_3 记为 $(p_3,0.9,\{-\},1,0)$；$p_5\in$ SNBP(t_4)，节点 p_5 记为 $(p_5,\alpha(p_5),\{t_1\},0,3)$；寻找 SNBP($t_1$)，$p_2$，$p_3\in$ SNBP(t_1)，由于 p_2，p_3 是种子库所，节点 p_2 记为 $(p_2,0.8,\{-\},1,0)$。步骤 2 结束。

步骤 3　步骤 2 经历的库所及变迁是 p_{10} 的求解空间。

步骤 4　在未知节点中，从深度为 2 的节点开始求解，$\alpha(p_5)=\min(\alpha(p_2),\alpha(p_3))\times\mu1=\min(0.8,0.9)\times0.95=0.76$；深度为 1 的节点，$\alpha(p_8)=\alpha(p_5)\times\mu4=0.76\times0.9=0.684$，$\alpha(p_6)=\min(\alpha(p_3),\alpha(p_4))\times\mu3=\min(0.9,0.7)\times0.95=0.665$；目标节点 $\alpha(p_{10})=\max(\alpha(p_4)\times\mu5,\min(\alpha(p_6),\alpha(p_8))\times\mu7)=\max(0.7\times0.9,\min(0.665,0.684)\times0.85)=0.63$。

步骤 5　同理按步骤 2 到步骤 4，$\alpha(p_9)\approx0.650$。

结束语　本文针对模糊产生式规则的知识库，在已知决策目标的前提下，设计了该知识库的 FPN 模型及基于递归的逆向知识推理算法，并以实例对该算法进行了验证。该算法的优点在于逻辑性强，利于计算机实现，而且逆向推理策略能有效减少计算空间，提高计算效率。下一步工作在于将此算法编程实现，在知识系统中予以实用。

参 考 文 献

[1] Chen Shyi-ming, Ke Jyh-sheng, Chang Jin-fu. Knowledge representation using fuzzy Petri nets [J]. IEEE Transaction on Knowledge and Data Engineering, 1990, 2(3): 311

[2] Looney C G. Petri fuzzy nets for rule-based decision-making[J]. IEEE Transactions on system, man, and cybemetics-part A, 1998, 18(1): 178-183

[3] 袁崇义. Petri 网原理[M]. 北京：电子工业出版社，1998

[4] Wang Shyue-liang, Wu Yi-huey. Reasoning in Fuzzy Production Systems when Input Inlormation is Incomplete [C] // Proceedings of IEEE International Fuzzy Systems Conference. 1999: 1557-1561

[5] 门鹏，段振华. 基于代数的模糊 petri 网逆向推理算法[J]. 系统仿真学报，2007，19(1)