

# 基于直觉模糊 Petri 网的知识表示和推理

孟飞翔<sup>1</sup>, 雷英杰<sup>1</sup>, 余晓东<sup>1</sup>, 雷 阳<sup>2</sup>

(1. 空军工程大学防空反导学院, 陕西西安 710051; 2. 武警工程大学, 陕西西安 710086)

**摘要:** 针对模糊 Petri 网存在隶属度单一的问题, 将直觉模糊集理论与 Petri 网理论相结合, 构建直觉模糊 Petri 网(Intuitionistic Fuzzy Petri Nets, IFPN)模型, 用于知识的表示和推理. 首先构建了 IFPN 模型, 并将其应用于知识的表示, 通过在模型中引入抑止转移弧, 解决了否命题的表示问题. 其次提出了基于矩阵运算的 IFPN 推理算法, 通过修改变迁触发后 token 值的传递规则, 解决了推理过程中的事实的保留问题; 通过修改变迁的触发规则, 抑制了变迁的重复触发. 最后对推理算法进行了分析, 并举例验证了提出的 IFPN 模型及其推理算法的可行性, 结果表明 IFPN 是对 FPN 的有效扩充和发展, 其对推理结果的描述更加细腻、全面.

**关键词:** 直觉模糊 Petri 网; 直觉模糊产生式规则; 知识表示; 直觉模糊推理

**中图分类号:** TP18      **文献标识码:** A      **文章编号:** 0372-2112 (2016)01-0077-10

**电子学报 URL:** <http://www.ejournal.org.cn>      **DOI:** 10.3969/j.issn.0372-2112.2016.01.012

## Knowledge Representation and Reasoning Using Intuitionistic Fuzzy Petri Nets

MENG Fei-xiang<sup>1</sup>, LEI Ying-jie<sup>1</sup>, YU Xiao-dong<sup>1</sup>, LEI Yang<sup>2</sup>

(1. Air and Missile Defense College, Air Force Engineering University, Xi'an, Shaanxi 710051, China;

2. Engineering University of Armed Police Force, Xi'an, Shaanxi 710086, China)

**Abstract:** Aimed at fuzzy Petri nets (FPN) existing membership single issue, intuitionistic fuzzy Petri nets (IFPN) was constructed for knowledge representation and reasoning by combining intuitionistic fuzzy sets theory and Petri net theory. Firstly, IFPN model was constructed for knowledge representation, and the issue of negative proposition representation was solved by introducing inhibitive transfer arcs into the model. Secondly, the algorithm based on matrix operation was presented, the issue of fact reservation in reasoning procedure was solved by modifying token value's transfer rules after transitions being fired, and the issue of transitions repeatedly being fired was inhibited by modifying firing rules of transitions. Lastly, the algorithm was analyzed, and the feasibility of proposed IFPN model and algorithm was proved through an example. The result indicates that IFPN is an effective extension and development of FPN, and it describes the reasoning result more delicately and comprehensively.

**Key words:** intuitionistic fuzzy Petri nets (IFPN); intuitionistic fuzzy production rules; knowledge representation; intuitionistic fuzzy reasoning

## 1 引言

FPN 是基于模糊产生式规则 (Fuzzy Production Rules, FPRs) 的知识系统的良好建模工具<sup>[1]</sup>, 自 C G Looney<sup>[2]</sup> 1988 年提出 FPN, 国内外学者对基于 FPN 的知识表示及其推理方法进行了深入的研究. Shyi-Ming

Chen 提出了基于 FPN 的知识表示方法和推理算法<sup>[3]</sup>, 并将权值引入 FPN 中, 提出了加权模糊 Petri 网 (Weighted Fuzzy Petri Nets, WFPN)<sup>[4]</sup>. Gao Meimei 等<sup>[5]</sup> 研究了否命题在模糊推理 Petri 网 (Fuzzy Reasoning Petri Nets, FRPNs) 中的表示方法, 并提出了基于矩阵运算的 FRPNs 推理算法. 汪洋等<sup>[6]</sup> 指出了文献[5]中的否命题表

示方法存在不一致性的问题,并提出了一种含有命题逻辑的一致性模糊 Petri 网 (Consistent Fuzzy Petri Nets, CFPN),用同一库所同时表示原命题和否命题.贾立新等<sup>[7]</sup>提出了基于矩阵运算的 FPN 形式化推理算法,充分利用了 Petri 网的并行运算能力,但计算出的结果命题可信度可能大于 1.

随着知识的表示日益复杂,传统的 FPN 并不能很好的满足知识表示的需求.许多学者对其进行了扩展,提出了多种扩展 FPN 模型. Xiaoou Li 等<sup>[8,9]</sup>提出了一种具有学习能力的自适应模糊 Petri 网 (Adaptive Fuzzy Petri Nets, AFPN). Hu-Chen Liu 等<sup>[10,11]</sup>提出了一种动态自适应模糊 Petri 网 (Dynamic Adaptive Fuzzy Petri Nets, DAFPN),该模型具有动态适应能力,因而能更准确地表示复杂的基于知识的专家系统. Victor R L Shen<sup>[12]</sup>提出了一种高级模糊 Petri 网 (High-Level Fuzzy Petri Nets, HLFPN),能同时表示 IF-THEN 和 IF-THEN-ELSE 规则,从而具有了解决否定问题的能力. Witold Pedrycz<sup>[13]</sup>在 FPN 中加入时间因素,提出了模糊时间 Petri 网 (fuzzy timed Petri Nets, ftPNs).

近年来,FPN 被广泛应用于知识的表示和推理<sup>[10,11]</sup>、建模仿真<sup>[14]</sup>以及故障诊断<sup>[15]</sup>等领域,但对 FPN 的研究始终局限于将 Zadeh 模糊集 (Zadeh Fuzzy Sets, ZFS) 理论和 Petri 网理论相结合. ZFS 采用单一标度 (即隶属度或隶属函数) 定义模糊集,只能描述“亦此亦彼”的“模糊概念”,无法表示中立状态,FPN 在继承 ZFS 优点的同时,也继承了其隶属度单一缺点.而直觉模糊集 (Intuitionistic Fuzzy Sets, IFS)<sup>[16]</sup>由于增加了一个新的属性参数——非隶属度函数,因此可以描述“中立”的概念,它更加细腻地刻画了客观世界的模糊本质,是对 ZFS 最有影响的一种扩充和发展.为此,本文将 IFS 理论与 Petri 理论相结合构建 IFPN 模型用于知识的表示和推理.通过在 IFPN 模型中引入抑制转移弧,修改变迁触发后库所 token 值的传递规则以及变迁的触发条件,解决了否命题的表示,事实的保留和变迁的重复触发等问题.

## 2 直觉模糊集及基本运算

Atanassov 对直觉模糊集给出如下定义.

**定义 1** (直觉模糊集<sup>[16]</sup>) 设  $X$  是一个给定论域,则  $X$  上的一个直觉模糊集  $A$  为

$$A = \{ \langle x, \mu_A(x), \gamma_A(x) \rangle \mid x \in X \} \quad (1)$$

其中  $\mu_A(x): X \rightarrow [0, 1]$  和  $\gamma_A(x): X \rightarrow [0, 1]$  分别表示  $A$  的隶属函数和非隶属函数,且对于  $A$  上的所有  $x \in X$ ,满足  $0 \leq \mu_A(x) + \gamma_A(x) \leq 1$ . 由隶属度  $\mu_A(x)$  和非隶属度

$\gamma_A(x)$  所组成的有序区间对  $\langle \mu_A(x), \gamma_A(x) \rangle$  为直觉模糊数.

当  $X$  为连续空间时,

$$A = \int_X \langle \mu_A(x), \gamma_A(x) \rangle / x, x \in X \quad (2)$$

当  $X = \{x_1, x_2, \dots, x_n\}$  为离散空间时,

$$A = \sum_{i=1}^n \langle \mu_A(x_i), \gamma_A(x_i) \rangle / x_i, x_i \in X \quad (3)$$

对于  $X$  中的直觉模糊子集  $A$ , 称  $\pi_A(x) = 1 - \mu_A(x) - \gamma_A(x)$  为  $A$  中  $x$  的直觉指数 (Intuitionistic Index), 它是  $x$  对  $A$  的犹豫程度 (Hesitancy degree) 的一种测度. 显然, 对于每一个  $x \in X, 0 \leq \pi_A(x) \leq 1$ , 且对于  $X$  中的每一个一般模糊子集  $A, \pi_A(x) = 1 - \mu_A(x) - [1 - \mu_A(x)] = 0, \forall x \in X$ .

**定义 2** (直觉模糊集基本运算<sup>[16]</sup>) 设  $A$  和  $B$  是论域  $X$  上的直觉模糊子集, 则有

- (1)  $A \cap B = \{ \langle x, \mu_A(x) \wedge \mu_B(x), \gamma_A(x) \vee \gamma_B(x) \rangle \mid \forall x \in X \}$
- (2)  $A \cup B = \{ \langle x, \mu_A(x) \vee \mu_B(x), \gamma_A(x) \wedge \gamma_B(x) \rangle \mid \forall x \in X \}$
- (3)  $\bar{A} = A^c = \{ \langle x, \gamma_A(x), \mu_A(x) \rangle \mid x \in X \}$
- (4)  $A \subseteq B \Leftrightarrow \forall x \in X, [\mu_A(x) \leq \mu_B(x) \wedge \gamma_A(x) \geq \gamma_B(x)]$
- (5)  $A \subset B \Leftrightarrow \forall x \in X, [\mu_A(x) < \mu_B(x) \wedge \gamma_A(x) > \gamma_B(x)]$
- (6)  $A = B \Leftrightarrow \forall x \in X, [\mu_A(x) = \mu_B(x) \wedge \gamma_A(x) = \gamma_B(x)]$

## 3 基于 IFPN 模型的知识表示

### 3.1 直觉模糊产生式规则

FPRs 是产生式规则与 ZFS 理论相结合的产物, 它既具有产生式规则知识表示直观的优点又具有模糊推理的功能, 但存在隶属度单一的缺点. 为此, 本节将产生式规则与 IFS 理论相结合, 构建直觉模糊产生式规则 (Intuitionistic Fuzzy Production Rules, IFPR) 用于知识的表示.

假设  $R = \{R_1, R_2, \dots, R_n\}$  是一个 IFPR 集, 规则  $R_i$  最基本的形式如下:

$$R_i: \text{IF } d_j \text{ THEN } d_k (CF_i, \lambda_i)$$

其中  $d_j$  和  $d_k$  是直觉模糊命题, 分别表示规则  $R_i$  的前提条件和结论, 它们的真值为  $\theta_j$  和  $\theta_k$ ;  $CF_i$  和  $\lambda_i$  分别表示规则  $R_i$  的可信度和阈值;  $\theta_j, \theta_k, CF_i$  和  $\lambda_i$  为直觉模糊数.

参考文献[3, 5]对 FPRs 的分类, 本文将常见的 IFPR 类型归纳为如下几种:

(1) 类型 1: 简单的 IFPR

$$R_i: \text{IF } d_j \text{ THEN } d_k (CF_i, \lambda_i)$$

假设  $\theta_j = (\mu_j, \gamma_j), \lambda_i = (\alpha_i, \beta_i), CF_i = (C\mu_i, C\gamma_i),$

(i) 当且仅当同时满足  $\mu_j \geq \alpha_i, \gamma_j \leq \beta_i$  时,  $R_i$  才被应用, 此时  $d_k$  的真值为  $\theta_k = (\mu_k, \gamma_k)$ , 其中:

$$\begin{cases} \mu_k = \mu_j \times C\mu_i \\ \gamma_k = \gamma_j + C\gamma_i - \gamma_j \times C\gamma_i \end{cases} \quad (4)$$

(ii) 其他条件下,  $d_k$  的真值不变.

(2) 类型 2: 具有合取式前提条件的 IFPR

$R_i: \text{IF } d_{j_1} \text{ AND } d_{j_2} \text{ AND } \dots \text{ AND } d_{j_n} \text{ THEN } d_k (CF_i, \lambda_i)$  假设  $\theta_{j_m} = (\mu_{j_m}, \gamma_{j_m}) (m = 1, 2, \dots, n)$ ,  $\lambda_i = (\alpha_i, \beta_i)$ ,  $CF_i = (C\mu_i, C\gamma_i)$ ,

(i) 当且仅当同时满足

$\min(\mu_{j_1}, \mu_{j_2}, \dots, \mu_{j_n}) \geq \alpha_i$ ,  $\max(\gamma_{j_1}, \gamma_{j_2}, \dots, \gamma_{j_n}) \leq \beta_i$  时,  $R_i$  才被应用, 此时  $d_k$  的真值为  $\theta_k = (\mu_k, \gamma_k)$ , 其中:

$$\begin{cases} \mu_k = \min(\mu_{j_1}, \mu_{j_2}, \dots, \mu_{j_n}) \times C\mu_i \\ \gamma_k = \max(\gamma_{j_1}, \gamma_{j_2}, \dots, \gamma_{j_n}) + C\gamma_i \\ \quad - \max(\gamma_{j_1}, \gamma_{j_2}, \dots, \gamma_{j_n}) \times C\gamma_i \end{cases} \quad (5)$$

(ii) 其他条件下,  $d_k$  的真值不变.

(3) 类型 3: 具有析取式前提条件的 IFPR

$R_i: \text{IF } d_{j_1} \text{ OR } d_{j_2} \text{ OR } \dots \text{ OR } d_{j_n} \text{ THEN } d_k (CF_i, \lambda_i)$

可以等价如下  $n$  条规则

$$R_{i1}: \text{IF } d_{j_1} \text{ THEN } d_k (CF_i, \lambda_i)$$

$$R_{i2}: \text{IF } d_{j_2} \text{ THEN } d_k (CF_i, \lambda_i)$$

.....

$$R_{in}: \text{IF } d_{j_n} \text{ THEN } d_k (CF_i, \lambda_i)$$

假设  $d_{j_m} (m = 1, 2, \dots, n)$  的真值为  $\theta_{j_m} = (\mu_{j_m}, \gamma_{j_m})$ ,  $\lambda_i = (\alpha_i, \beta_i)$ ,  $CF_i = (C\mu_i, C\gamma_i)$ , 规则  $R_i$  中  $d_k$  的真值为  $\theta_k = (\mu_k, \gamma_k)$ , 等价规则  $R_{im} (m = 1, 2, \dots, n)$  中  $d_k$  的真值为  $\theta_{km} = (\mu_{km}, \gamma_{km})$ ,

(i) 如果等价规则  $R_{i1}, R_{i2}, \dots, R_{in}$  中存在一条或者多条规则被应用, 则规则  $R_i$  被应用,  $R_i$  中的  $d_k$  的最终真值为  $\theta_k = (\mu_k, \gamma_k)$ , 其中:

$$\begin{cases} \mu_k = \max(\mu_{k1}, \mu_{k2}, \dots, \mu_{kn}) \\ \gamma_k = \min(\gamma_{k1}, \gamma_{k2}, \dots, \gamma_{kn}) \end{cases} \quad (6)$$

在等价规则  $R_{im} (m = 1, 2, \dots, n)$  中,

① 当且仅当同时满足  $\mu_{j_m} \geq \alpha_i$ ,  $\gamma_{j_m} \leq \beta_i$  时,  $R_{im}$  才被应用, 此时  $R_{im}$  中的  $d_k$  的真值为  $\theta_{km} = (\mu_{km}, \gamma_{km})$ , 其

$$\begin{cases} \mu_{km} = \mu_{j_m} \times C\mu_i \\ \gamma_{km} = \gamma_{j_m} + C\gamma_i - \gamma_{j_m} \times C\gamma_i \end{cases}$$

② 其他条件下,  $R_{im}$  中的  $d_k$  的真值  $\theta_{km} = (\mu_{km}, \gamma_{km})$  不变.

(ii) 如果等价规则  $R_{i1}, R_{i2}, \dots, R_{in}$  都未被应用, 则规则  $R_i$  未被应用, 此时  $R_i$  中的  $d_k$  的真值  $\theta_k = (\mu_k, \gamma_k)$  不变.

(4) 类型 4: 具有合取式结论的 IFPR

$R_i: \text{IF } d_j \text{ THEN } d_{k1} \text{ AND } d_{k2} \text{ AND } \dots \text{ AND } d_{kn} (CF_i, \lambda_i)$  假设  $d_j$  的真值为  $\theta_j = (\mu_j, \gamma_j)$ ,  $\lambda_i = (\alpha_i, \beta_i)$ ,  $CF_i = (C\mu_i,$

$C\gamma_i)$ ,

(i) 当且仅当同时满足  $\mu_j \geq \alpha_i$ ,  $\gamma_j \leq \beta_i$ ,  $R_i$  才被应用,  $d_{km} (m = 1, 2, \dots, n)$  的真值为  $\theta_{km} = (\mu_{km}, \gamma_{km})$ , 其中:

$$\begin{cases} \mu_{km} = \mu_j \times C\mu_i \\ \gamma_{km} = \gamma_j + C\gamma_i - \gamma_j \times C\gamma_i \end{cases} \quad (7)$$

(ii) 其他条件下,  $d_{km}$  的真值不变.

(5) 类型 5: 具有析取式结论的 IFPR

$R_i: \text{IF } d_j \text{ THEN } d_{k1} \text{ OR } d_{k2} \text{ OR } \dots \text{ OR } d_{kn} (CF_i, \lambda_i)$

由于该类型的规则推理结果不确定, 不允许出现在规则库中, 本文不做讨论.

### 3.2 否命题的表示

在一个规则集中, 命题的原命题和否命题可能同时存在. 如规则集  $S_1 = \{R_1, R_2\}$ , 其中

$R_1: \text{IF } d_1 \text{ AND } \neg d_2 \text{ THEN } \neg d_4$

$R_2: \text{IF } d_2 \text{ AND } d_3 \text{ THEN } d_4 \text{ AND } \neg d_5$

为合理表示原命题与否命题, 文献[9]通过引入负权值表示否定的含义; 文献[17]用两个不同的库所分别表示原命题和否命题, 但这会增加模型的复杂度; 文献[5]用抑止弧和新库所分别表示前提条件和结论中的否命题, 但存在模型表示不唯一和推理结果矛盾等问题. 文献[6]将原命题和否命题分别理解为该命题在规则中起促进和阻碍作用(例如  $R_1$  中  $\neg d_2$  在推理中阻碍了  $d_4$  的发生), 通过在模型的转移弧上加入标志“-”来区分促进和阻碍作用(没有标志的转移弧为正转移弧, 表示促进作用; 带有“-”的为抑制转移弧, 表示阻碍作用), 成功地用同一库所同时表示原命题和否命题, 避免了改变原模型的结构.

本文将这种表示方法引入 IFPN 模型中, 用于表示模型中的原命题和否命题. 规则集  $S_1$  可用图 1 表示.

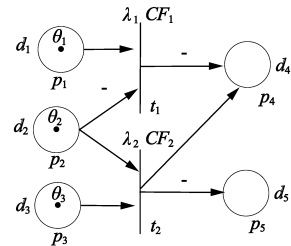


图1 规则集  $S_1$  的 IFPN 模型

### 3.3 IFPN 的定义

定义 3 (直觉模糊 Petri 网)

IFPN 可定义为一个 11 元组

$\text{IFPN} = (P, T, D, I, IN, O, ON, \delta, \theta, Th, CF)$ , 其中

(1)  $P = \{p_1, p_2, \dots, p_n\}$  是一个有限库所集合;

(2)  $T = \{t_1, t_2, \dots, t_m\}$  是一个有限变迁集合;

(3)  $D = \{d_1, d_2, \dots, d_n\}$  是一个有限命题集合,  $|P|$

$$= |D|, P \cap T \cap D = \emptyset;$$

(4)  $I: P \times T \rightarrow \{0, 1\}$  是一个表示从库所到变迁(从命题到规则)的  $n \times m$  维输入正转移矩阵, 其矩阵元素  $I(p_i, t_j)$  满足如下条件: 当存在由  $p_i$  到  $t_j$  的正转移弧时,  $I(p_i, t_j) = 1$ ; 否则  $I(p_i, t_j) = 0$ , 其中  $i = 1, 2, \dots, n, j = 1, 2, \dots, m$ ;

(5)  $IN: P \times T \rightarrow \{0, 1\}$  是一个表示从库所到变迁(从命题到规则)的  $n \times m$  维输入抑制转移矩阵, 其矩阵元素  $IN(p_i, t_j)$  满足如下条件: 当存在由  $p_i$  到  $t_j$  的抑制转移弧时,  $IN(p_i, t_j) = 1$ ; 否则  $IN(p_i, t_j) = 0$ , 其中  $i = 1, 2, \dots, n, j = 1, 2, \dots, m$ ;

(6)  $O: P \times T \rightarrow \{0, 1\}$  是一个表示从变迁到库所(从规则到结论)的  $n \times m$  维输出正转移矩阵, 其矩阵元素  $O(p_i, t_j)$  满足如下条件: 当存在由  $t_j$  到  $p_i$  的正转移弧时,  $O(p_i, t_j) = 1$ ; 否则  $O(p_i, t_j) = 0$ , 其中  $i = 1, 2, \dots, n, j = 1, 2, \dots, m$ ;

(7)  $ON: P \times T \rightarrow \{0, 1\}$  是一个表示从变迁到库所(从规则到结论)的  $n \times m$  维输入抑制转移矩阵, 其矩阵元素  $ON(p_i, t_j)$  满足如下条件: 当存在由  $t_j$  到  $p_i$  的抑制转移弧时,  $ON(p_i, t_j) = 1$ ; 否则  $ON(p_i, t_j) = 0$ , 其中  $i = 1, 2, \dots, n, j = 1, 2, \dots, m$ ;

(8)  $\delta: P \rightarrow D$  表示库所与命题之间的一一对应关系, 其中  $\delta(p_i) = d_i$ ;

(9)  $\theta = (\theta_1, \theta_2, \dots, \theta_n)^T$  是一个  $n$  维列向量, 表示库所的 token 值(即命题的模糊真值), 其中  $\theta_i = (\mu_i, \gamma_i)$  是一个直觉模糊数, 表示库所  $p_i$  中的命题  $d_i$  的真值; 命题的初始真值记作  $\theta^0 = (\theta_1^0, \theta_2^0, \dots, \theta_n^0)^T = ((\mu_1^0, \gamma_1^0), (\mu_2^0, \gamma_2^0), \dots, (\mu_n^0, \gamma_n^0))^T$ ;

(10)  $Th = (\lambda_1, \lambda_2, \dots, \lambda_m)^T$  是一个  $m$  维列向量, 表示变迁的阈值(即规则启动的条件), 其中  $\lambda_j = (\alpha_j, \beta_j)$  是一个直觉模糊数, 表示变迁  $t_j$  的阈值(即规则  $R_j$  启动的条件);

(11)  $CF = \text{diag}(CF_1, CF_2, \dots, CF_m)$  为一个  $m \times m$  维的矩阵, 表示规则的可信度, 其中  $CF_j = (C\mu_j, C\gamma_j)$  为一直觉模糊数, 表示规则  $R_j$  的可信度;  $C\mu_j$  表示规则  $R_j$  可信度的支持程度, 称作置信度,  $C\gamma_j$  表示规则  $R_j$  可信度的反对程度, 称作非置信度。

**定理 1** FPN 是 IFPN 的特例。

**证明** 当库所  $p_i$  中的命题  $d_i (i = 1, 2, \dots, n)$  的直觉模糊指数  $\pi_i(x) = 0$  时, 即满足条件  $\mu_i(x) + \gamma_i(x) = 1$ , 此时库所中的直觉模糊命题变成了模糊命题, IFPN 即简化成了 FPN, 所以 FPN 是 IFPN 的特例。

### 3.4 基于 IFPN 的知识表示方法

IFPR 集与 IFPN 模型的对应关系如下, 可用表

1 表示。

表 1 IFPR 集与 IFPN 模型的对应关系

IFPR 集	IFPN 模型
推理规则 $R_j$	变迁 $t_j$
$R_j$ 的前提条件	$t_j$ 的输入库所
$R_j$ 的结论	$t_j$ 的输出库所
命题 $d_k$	库所 $p_k$
$d_k$ 的真值	$p_k$ 的 token 值
$R_j$ 的阈值	$t_j$ 的阈值
$R_j$ 的可信度	$t_j$ 的可信度
$R_j$ 的应用	$t_j$ 的触发

按照表 1 所示的对应关系, 可以将 IFPR 集映射为 IFPN。

#### (1) 简单的 IFPR 的 IFPN 模型

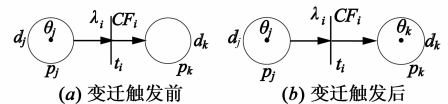


图 2 简单的 IFPR 的 IFPN 模型

其中, 库所  $p_j$  和  $p_k$  分别表示规则的前提条件命题  $d_j$  和结论命题  $d_k$ ;  $p_j$  和  $p_k$  中的 token 值  $\theta_j = (\mu_j, \gamma_j)$  和  $\theta_k = (\mu_k, \gamma_k)$  分别表示  $d_j$  和  $d_k$  的真值;  $\lambda_i = (\alpha_i, \beta_i)$  表示变迁  $t_i$  的阈值(即规则  $R_i$  的阈值);  $CF_i = (C\mu_i, C\gamma_i)$  表示变迁  $t_i$  的可信度(即规则  $R_i$  的可信度),  $\theta_j, \theta_k, \lambda_i$  和  $CF_i$  为直觉模糊数。

当且仅当同时满足  $\mu_j \geq \alpha, \gamma_j \leq \beta$ , 变迁  $t_i$  才能触发(即规则  $R_i$  应用),  $p_k$  的 token 值(即  $d_k$  的真值)  $\theta_k = (\mu_k, \gamma_k)$ , 其中:

$$\begin{cases} \mu_k = \mu_j \times C\mu_i \\ \gamma_k = \gamma_j + C\gamma_i - \gamma_j \times C\gamma_i \end{cases} \quad (8)$$

#### (2) 具有合取式前提条件的 IFPR 的 IFPN 模型

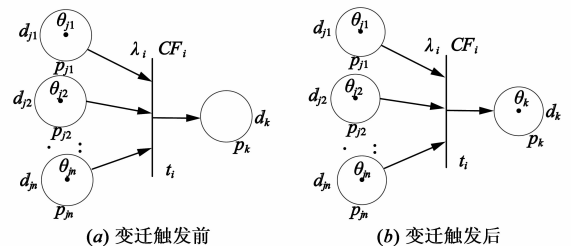


图 3 具有合取式前提条件的 IFPR 的 IFPN 模型

假设  $d_{j_m} (m = 1, 2, \dots, n)$  和  $d_k$  的真值分别为  $\theta_{j_m} = (\mu_{j_m}, \gamma_{j_m})$  和  $\theta_k = (\mu_k, \gamma_k)$ ,  $\lambda_i = (\alpha_i, \beta_i)$ ,  $CF_i = (C\mu_i, C\gamma_i)$ 。

当且仅当同时满足  $\min(\mu_{j_1}, \mu_{j_2}, \dots, \mu_{j_n}) \geq \alpha_i$ ,  $\max(\gamma_{j_1}, \gamma_{j_2}, \dots, \gamma_{j_n}) \leq \beta_i$  时,  $t_i$  才能触发(即  $R_i$  应用),  $p_k$  的 token 值(即  $d_k$  的真值)  $\theta_k = (\mu_k, \gamma_k)$ , 其中:

$$\begin{cases} \mu_k = \min(\mu_{j_1}, \mu_{j_2}, \dots, \mu_{j_n}) \times C\mu_i \\ \gamma_k = \max(\gamma_{j_1}, \gamma_{j_2}, \dots, \gamma_{j_n}) + C\gamma_i - \max(\gamma_{j_1}, \gamma_{j_2}, \dots, \gamma_{j_n}) \times C\gamma_i \end{cases} \quad (9)$$

(3) 具有析取式前提条件的 IFPR 的 IFPN 模型

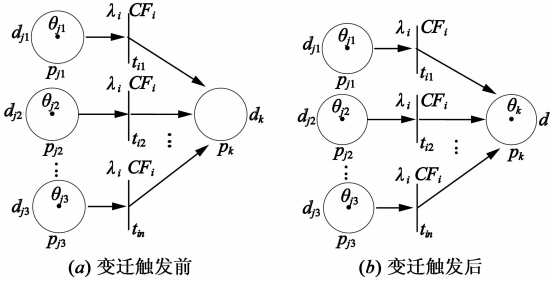


图4 具有析取式前提条件的IFPR的IFPN模型

假设  $d_{j_m}$  ( $m=1, 2, \dots, n$ ) 和  $d_k$  的真值分别为  $\theta_{j_m} = (\mu_{j_m}, \gamma_{j_m})$  和  $\theta_k = (\mu_k, \gamma_k)$ ,  $\lambda_i = (\alpha_i, \beta_i)$ ,  $CF_i = (C\mu_i, C\gamma_i)$ .

如果  $p_{j_1}, p_{j_2}, \dots, p_{j_n}$  的中有  $m$  个库所  $p_{j_1}, p_{j_2}, \dots, p_{j_m}$  ( $1 \leq m \leq n$ ) 的 token 值同时满足  $\mu_{j_m} \geq \alpha_i$ ,  $\gamma_{j_m} \leq \beta_i$ , 则变迁  $t_{j_1}, t_{j_2}, \dots, t_{j_m}$  触发(此时  $R_i$  应用),  $p_k$  的 token 值(即  $d_k$  的真值)  $\theta_k = (\mu_k, \gamma_k)$ , 其中:

$$\begin{cases} \mu_k = \max(\mu_{j_1} \times C\mu_i, \mu_{j_2} \times C\mu_i, \dots, \mu_{j_m} \times C\mu_i) \\ \gamma_k = \min(\gamma_{j_1} + C\gamma_i - \gamma_{j_1} \times C\gamma_i, \gamma_{j_2} + C\gamma_i - \gamma_{j_2} \times C\gamma_i, \dots, \gamma_{j_m} + C\gamma_i - \gamma_{j_m} \times C\gamma_i) \end{cases} \quad (10)$$

(4) 具有合取式结论的 IFPR 的 IFPN 模型

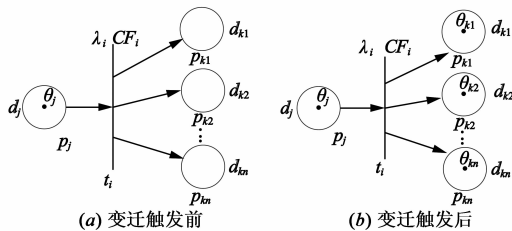


图5 具有合取式结论的IFPR的IFPN模型

假设  $p_j, p_{k_1}, p_{k_2}, \dots, p_{k_n}$  的 token 值分别为  $\theta_j, \theta_{k_1}, \theta_{k_2}, \dots, \theta_{k_n}$ , 其中  $\theta_j = (\mu_j, \gamma_j)$ ,

$\theta_{k_i} = (\mu_{k_i}, \gamma_{k_i})$  ( $i=1, 2, \dots, n$ ),  $\lambda_i = (\alpha_i, \beta_i)$ ,  $CF_i = (C\mu_i, C\gamma_i)$ .

当且仅当同时满足  $\mu_j \geq \alpha_i$ ,  $\gamma_j \leq \beta_i$ ,  $t_i$  才能触发(即  $R_i$  应用),  $p_{k_i}$  的 token 值(即  $d_{k_i}$  的真值)  $\theta_{k_i} = (\mu_{k_i}, \gamma_{k_i})$ , 其中:

$$\begin{cases} \mu_{k_i} = \mu_j \times C\mu_i \\ \gamma_{k_i} = \gamma_j + C\gamma_i - \gamma_j \times C\gamma_i \end{cases} \quad (11)$$

## 4 基于 IFPN 的推理算法

### 4.1 IFPN 的扩展

为更好地利用 IFPN 进行模糊推理,需要解决以下两个问题:事实的保留和变迁的重复触发.

#### (1) 事实的保留

事实的保留是指在 IFPN 推理过程中当变迁  $t$  触发后,输出库所的 token 值更新,而输入库所的 token 值保持不变.传统的 FPN 描述的是信息(token 值)的流动,当信息从输入库所流到输出库所,便在输入库所中消失了,这相当于在推理过程中推理的前提条件随着结论的出现而消失,显然不符合知识处理的要求.

为了保留初始事实, Nazareth<sup>[18]</sup> 提出从变迁到它的所有输入库所都增加一条额外的有向弧,该方法会增加模型的复杂度. C G Looney<sup>[2]</sup> 通过复制 token 值,将初始 token 留在输入库所而将 token 的副本放入到输出库所,从而解决了事实保留的问题. Gao Meimei 等<sup>[5]</sup> 通过修改推理算法达到了保留了事实的目的. 本文通过修改变迁触发后库所 token 值的传递规则,解决 IFPN 推理过程中事实保留的问题.

假设在变迁  $t_k$  触发前,库所中的 token 值为  $\theta^k = (\theta_1^k, \theta_2^k, \dots, \theta_n^k)^T = ((\mu_1^k, \gamma_1^k), (\mu_2^k, \gamma_2^k), \dots, (\mu_n^k, \gamma_n^k))^T$ , 其中  $\theta_i^k$  表示库所  $p_i$  中的 token 值. 变迁  $t_k$  触发后,库所中的 token 值为

$\theta^{k'} = (\theta_1^{k'}, \theta_2^{k'}, \dots, \theta_n^{k'})^T = ((\mu_1^{k'}, \gamma_1^{k'}), (\mu_2^{k'}, \gamma_2^{k'}), \dots, (\mu_n^{k'}, \gamma_n^{k'}))^T$ , 其中

$\theta_i^{k'} = \begin{cases} \theta_i^k, & p_i \in O(t_k) \cap ON(t_k) \\ \theta_i^k, & p_i \notin O(t_k) \cap ON(t_k) \end{cases}$ ,  $O(t_k)$  和  $ON(t_k)$  表示

变迁  $t_k$  的输出库所集合.

通过修改变迁触发后库所 token 值的传递规则,既保留了事实又没有改变原 IFPN 的结构,从而避免了降低推理效率.

#### (2) 抑制变迁的重复触发(避免规则重复使用)

在基于 IFPR 的推理过程中,如果规则的前提条件没有改变,那么该规则最多只能应用一次.相应地在基于 IFPN 的推理过程中,若变迁的输入库所没有变化,则该变迁最多只能触发一次.而事实得到保留后(即输入库所的 token 值得到保留),变迁的触发条件就一直满足,这会导致已经触发过的变迁重复触发,从而导致反复执行同一条规则,增加不必要的计算.为此 Nazareth<sup>[18]</sup> 提出为每个变迁增加一个只包含单个 token 值的库所,在变迁触发后,这个库所的 token 值消失,从而避免了变迁的重复触发,但这会增加库所的数目,导致模

型变的更加复杂.

本文通过在基于 IFPN 的推理算法中增加“判断每个变迁的等效输入是否大于先前的输入”这一步骤,来避免变迁的重复触发. 对任意变迁,只要其所有输入库所的等效输入不大于先前的等效输入,该变迁就没有必要触发.

## 4.2 基于 IFPN 的推理算法

### 4.2.1 定义算子

为了简洁地表示推理算法,定义如下算子:

(1) 乘法算子  $\otimes$ :  $C = A \otimes B$ , 其中

$$C = (c_{ij})_{m \times n} = ((\mu a_{ij}, \gamma a_{ij}))_{m \times n},$$

$$A = (a_{ij})_{m \times n},$$

$$B = (b_{ij})_{n \times 1} = ((b\mu_j, b\gamma_j))_{n \times 1},$$

$$\mu a_{ij} = \max \left\{ x_i \mid x_i = \begin{cases} b\mu_j, & a_{ij} = 1 \\ 0, & a_{ij} = 0 \end{cases} \right\},$$

$$\gamma a_{ij} = \min \left\{ y_i \mid y_i = \begin{cases} b\gamma_j, & a_{ij} = 1 \\ 1, & a_{ij} = 0 \end{cases} \right\}$$

(2) 加法算子  $\oplus$ :  $C = A \oplus B$ , 其中

$$C = (c_{ij})_{m \times n} = ((\mu c_{ij}, \gamma c_{ij}))_{m \times n},$$

$$A = (a_{ij})_{m \times n} = ((\mu a_{ij}, \gamma a_{ij}))_{m \times n},$$

$$B = (b_{ij})_{m \times n} = ((\mu b_{ij}, \gamma b_{ij}))_{m \times n},$$

$$c_{ij} = \max(a_{ij}, b_{ij}) = (\max(\mu a_{ij}, \mu b_{ij}), \min(\gamma a_{ij}, \gamma b_{ij}))$$

(3) 比较算子  $\ominus$ :  $C = A \ominus B$ , 其中

$$C = (c_{ij})_{m \times n} = ((\mu c_{ij}, \gamma c_{ij}))_{m \times n},$$

$$A = (a_{ij})_{m \times n} = ((\mu a_{ij}, \gamma a_{ij}))_{m \times n},$$

$$B = (b_{ij})_{m \times n} = ((\mu b_{ij}, \gamma b_{ij}))_{m \times n},$$

$$c_{ij} = (\mu c_{ij}, \gamma c_{ij}) = \begin{cases} (\mu a_{ij}, \gamma a_{ij}), & \mu a_{ij} > \mu b_{ij} \text{ and } \gamma a_{ij} < \gamma b_{ij} \\ (0, 1), & \text{other} \end{cases}$$

(4) 比较算子  $\otimes$ :  $C = A \otimes B$ , 其中

$$C = (c_1, c_2, \dots, c_m)^T = ((\mu c_1, \gamma c_1), (\mu c_2, \gamma c_2), \dots, (\mu c_m, \gamma c_m))^T,$$

$$A = (a_1, a_2, \dots, a_m)^T$$

$$= ((\mu a_1, \gamma a_1), (\mu a_2, \gamma a_2), \dots, (\mu a_m, \gamma a_m))^T,$$

$$B = (b_1, b_2, \dots, b_m)^T$$

$$= ((\mu b_1, \gamma b_1), (\mu b_2, \gamma b_2), \dots, (\mu b_m, \gamma b_m))^T,$$

$$c_i = (\mu c_i, \gamma c_i) = \begin{cases} (\mu a_i, \gamma a_i), & \mu c_i \geq \mu a_i \text{ and } \gamma c_i \leq \gamma a_i \\ (0, 1), & \text{other} \end{cases}$$

(5) 直乘算子  $\odot$ :  $C = A \odot B$ , 其中

$$C = (c_{ij})_{m \times n} = ((\mu c_{ij}, \gamma c_{ij}))_{m \times n},$$

$$A = (a_{ij})_{m \times l} = ((\mu a_{ij}, \gamma a_{ij}))_{m \times l},$$

$$B = (b_{ij})_{l \times n} = ((\mu b_{ij}, \gamma b_{ij}))_{l \times n},$$

$$c_{ij} = (c_{ij})_{m \times n} = (\mu a_{ij} \times \mu b_{ij}, \gamma a_{ij} + \gamma b_{ij} - \gamma a_{ij} \times \gamma b_{ij})$$

(6) 向量否定算子 neg

已知  $\theta = (\theta_1, \theta_2, \dots, \theta_n)^T = ((\mu_1, \gamma_1), (\mu_2, \gamma_2), \dots, (\mu_n, \gamma_n))^T$ , 则

$$\begin{aligned} \text{neg}\theta &= \text{neg}(\theta_1, \theta_2, \dots, \theta_n)^T \\ &= \text{neg}((\mu_1, \gamma_1), (\mu_2, \gamma_2), \dots, (\mu_n, \gamma_n))^T \\ &= ((\gamma_1, \mu_1), (\gamma_2, \mu_2), \dots, (\gamma_n, \mu_n))^T = \bar{\theta} \end{aligned}$$

### 4.2.2 基于 IFPN 的推理算法

非循环网是指没有回路和环形的网,在大多数实际应用的知识库中几乎不存在循环<sup>[5]</sup>. 因此本文假设构建的基于 IFPR 的 IFPN 模型是一个非循环网,即模型中不存在回路.

**定义 4** (直接可达集,可达集<sup>[1,3]</sup>) 在 IFPN 模型中,假设  $t_i$  是一个变迁,  $p_i, p_j, p_k$  是库所, 如果  $p_i \in I(t_i) \cap IN(t_i)$  并且  $p_j \in O(t_i) \cap ON(t_i)$ , 则称从  $p_i$  直接可达  $p_j$ . 如果从  $p_i$  直接可达  $p_j$ , 且从  $p_j$  直接可达  $p_k$ , 则称从  $p_i$  可达  $p_k$ . 所有从  $p_i$  直接可达的库所构成的集合称为  $p_i$  的直接可达集 (immediate reachability set), 记为  $IRS(p_i)$ . 所有从  $p_i$  可达的库所构成的集合称为  $p_i$  的可达集 (reachability set), 记为  $RS(p_i)$ .

假设 IFPR 集  $S$  中有  $n$  个命题,  $m$  条规则, 对应的 IFPN 模型有  $n$  个库所,  $m$  个变迁, 则基于 IFPN 的推理算法如下:

#### 算法 1 基于 IFPN 的推理算法

Input: 输入正转移矩阵  $I$ , 输入抑制转移矩阵  $IN$ , 输出正转移矩阵  $O$ , 输出抑制转移矩阵  $ON$ , 变迁的阈值  $Th$ , 规则的可信度  $CF$ , 命题的初始真值  $\theta^0$ .

Output: 库所的 token 值 (即命题的真值), 迭代次数  $k$ .

预处理 (判断 IFPN 模型中是否有回路): 在 IFPN 模型中, 若  $\exists p_i \in RS(p_i)$ , 则模型中存在回路. 该模型不能应用该推理算法, 退出.

**Step1** 初始化所有输入, 令  $k=1, \theta^{k-1} = \theta^0 = (\theta_1^0, \theta_2^0, \dots, \theta_n^0)^T = ((\mu_1^0, \gamma_1^0), (\mu_2^0, \gamma_2^0), \dots, (\mu_n^0, \gamma_n^0))^T$ , 未知命题的真值用  $(0, 1)$  表示, 初始等效输入  $\rho_{k-1} = \rho_0 = ((0, 1), (0, 1), \dots, (0, 1))^T$ .

**Step2** 计算各个变迁的等效输入, 即将各个变迁的所有输入库所的 token 值等效为单个输入库所的 token 值, 结果为

$$\rho_k = (\rho\theta_1^k, \rho\theta_2^k, \dots, \rho\theta_m^k)^T = ((\rho\mu_1^k, \rho\gamma_1^k), (\rho\mu_2^k, \rho\gamma_2^k), \dots, (\rho\mu_m^k, \rho\gamma_m^k))^T$$

$$\text{其中 } \rho\mu_j^k = \min \left\{ x_i \mid x_i = \begin{cases} \mu_i^{k-1}, & I(p_i, t_j) = 1 \\ \gamma_i^{k-1}, & IN(p_i, t_j) = 1 \end{cases} \right\},$$

$$\rho\gamma_j^k = \max \left\{ y_i \mid y_i = \begin{cases} \gamma_i^{k-1}, & I(p_i, t_j) = 1 \\ \mu_i^{k-1}, & IN(p_i, t_j) = 1 \end{cases} \right\}$$

$$\text{即 } \rho_k = (I^T \otimes \theta_{k-1}^T) \oplus (IN^T \otimes \theta_{k-1}^T) \quad (12)$$

**Step3** (抑制变迁的重复触发) 判断每个变迁的等效输入是否大于先前的输入:

$$\rho'_k = \rho_k \ominus \rho_{k-1} \quad (13)$$

此时  $\rho'_k$  中记录的是有必要触发的变迁的等效输入. 如果  $\rho'_k = ((0,1), (0,1), \dots, (0,1))^T$ , 则推理结束, 输出命题最终值  $\theta^k$ , 此时  $\theta^k = \theta^{k-1}$ ; 否则推理继续.

**Step4** 将变迁等效输入与规则阈值进行比较, 保留可以使变迁触发的输入:

$$\rho_k'' = \rho'_k \otimes Th \quad (14)$$

**Step5** 变迁触发后, 计算结果命题真值:

①  $S_k = CF \odot \rho_k''$ , 其中

$$S_k = (s_1^k, s_2^k, \dots, s_m^k)^T \\ = ((\mu_1^k, \gamma_1^k), (\mu_2^k, \gamma_2^k), \dots, (\mu_n^k, \gamma_n^k))^T$$

$$\begin{cases} \mu_j^k = C\mu_j^k \times \rho\mu_j^k \\ \gamma_j^k = C\gamma_j^k + \rho\gamma_j^k - C\gamma_j^k \times \rho\gamma_j^k \end{cases}$$

②  $Y_k = (y\theta_1^k, y\theta_2^k, \dots, y\theta_n^k)^T$

$$= ((\mu_1^k, \gamma_1^k), (\mu_2^k, \gamma_2^k), \dots, (\mu_n^k, \gamma_n^k))^T$$

其中

$$y\mu_i^k = \max \left\{ x_i \mid x_i = \begin{cases} \mu_j^k, O(p_i, t_j) = 1 \\ \gamma_j^k, ON(p_i, t_j) = 1 \\ 0, O(p_i, t_j) = 0 \text{ and } ON(p_i, t_j) = 0 \end{cases} \right\}$$

$$y\gamma_i^k = \min \left\{ y_i \mid y_i = \begin{cases} \gamma_j^k, O(p_i, t_j) = 1 \\ \mu_j^k, ON(p_i, t_j) = 1 \\ 1, O(p_i, t_j) = 0 \text{ and } ON(p_i, t_j) = 0 \end{cases} \right\}$$

$$Y_k = (O \otimes S_k) \oplus (ON \otimes \bar{S}_k) \\ \text{即} = [O \otimes (CF \odot \rho_k'')] \oplus [ON \otimes \overline{(CF \odot \rho_k'')}] \quad (15)$$

**Step6** 计算所有库所的 token 值 (即所有命题的最终真值):

$$\theta^k = \theta^{k-1} \oplus Y_k = (\theta_1^k, \theta_2^k, \dots, \theta_n^k)^T \\ = ((\mu_1^k, \gamma_1^k), (\mu_2^k, \gamma_2^k), \dots, (\mu_n^k, \gamma_n^k))^T \quad (16)$$

本步骤保留了事实.

**Step7** 判断推理是否结束:

如果  $\theta^k = \theta^{k-1}$ , 推理结束, 输出命题最终值  $\theta^k$ ; 否则, 令  $k = k + 1$ , 转到 Step2.

### 4.3 算法分析

**定义 5** (源库所, 终结库所<sup>[8]</sup>)

如果一个库所没有输入库所, 就称该库所为源库所 (Source Places); 如果一个库所没有输出库所, 就称该库所为终结库所 (Sink Places).

**定义 6** (路径<sup>[8]</sup>)

对一个给定的库所  $p$ , 如果  $p$  可以通过变迁  $t_1, t_2, \dots, t_n$  顺序地从源库所中获取 token 值, 则称变迁序列  $t_1, t_2, \dots, t_n$  为库所  $p$  的一个路径 (route). 如果变迁序列  $t_1, t_2, \dots, t_n$  可以依次被触发, 则称该路径为活动路径.

**定理 2**  $\rho'_k = ((0,1), (0,1), \dots, (0,1))^T$  是推理结束的充分条件, 但不是必要条件.

**证明** (1) 当  $\rho'_k = ((0,1), (0,1), \dots, (0,1))^T$ , 由式 (14~16) 可知  $\rho_k'' = ((0,1), (0,1), \dots, (0,1))^T$ ,  $Y_k = ((0,1), (0,1), \dots, (0,1))^T$ ,  $\theta^k = \theta^{k-1} \oplus Y_k = \theta^{k-1}$ , 此

时推理必然结束, 即  $\rho'_k = ((0,1), (0,1), \dots, (0,1))^T$  是推理结束的充分条件.

(2) 当  $\rho'_k \neq ((0,1), (0,1), \dots, (0,1))^T$ , 如果变迁的等效输入不大于阈值, 由式 (14) 可知  $\rho_k'' = \rho'_k \otimes Th = ((0,1), (0,1), \dots, (0,1))^T$  仍成立, 此时  $Y_k = ((0,1), (0,1), \dots, (0,1))^T$ ,  $\theta^k = \theta^{k-1} \oplus Y_k = \theta^{k-1}$ . 所以  $\rho'_k = ((0,1), (0,1), \dots, (0,1))^T$  不是推理结束的必要条件.

**定理 3**  $\theta^k = \theta^{k-1}$  是推理结束的充要条件.

该定理显然成立, 证明略过.

**定理 4** 该推理算法可以在有限  $k$  次循环后结束, 其中  $1 \leq k \leq h + 1$ ,  $h$  表示 IFPN 模型中的最长路径的变迁数目.

**证明** (1) 先证该推理算法可以在有限  $k$  次循环后结束

假设在第  $k$  次推理结束时,  $\theta^k = \theta^{k-1}$ , 显然根据定理 3, 推理已经结束.

(2) 再证  $1 \leq k \leq h + 1$

(i) 先证  $k = h + 1$

假设  $h$  表示 IFPN 模型中的最长路径的变迁数目, 我们只需证明当  $k = h + 1$  时, 推理结束后  $\rho'_{h+1} = ((0,1), (0,1), \dots, (0,1))^T$  或者  $\theta^{h+1} = \theta^h$ .

已知  $\theta^h = (\theta_1^h, \theta_2^h, \dots, \theta_n^h)^T = ((\mu_1^h, \gamma_1^h), (\mu_2^h, \gamma_2^h), \dots, (\mu_n^h, \gamma_n^h))^T$ ,

$\theta^{h+1} = (\theta_1^{h+1}, \theta_2^{h+1}, \dots, \theta_n^{h+1})^T = ((\mu_1^{h+1}, \gamma_1^{h+1}), (\mu_2^{h+1}, \gamma_2^{h+1}), \dots, (\mu_n^{h+1}, \gamma_n^{h+1}))^T$ .

假设  $p_i$  为该最长路径的终结库所, 对应的变迁为  $t_i$ , 则在推理进行到第  $h$  步和第  $h + 1$  步, 库所  $p_j (j = 1, 2, 3, \dots, n, j \neq i)$  中的 token 值  $\theta_j^h$  和  $\theta_j^{h+1}$  完全相同, 即  $\theta^h$  和  $\theta^{h+1}$  中除了终结库所的 token 值不同, 其他的库所的 token 值完全相同, 而各个变迁的等效输入只与它的输入库所的 token 值相关, 与它的输出库所 token 值无关.

所以综上所述,  $k = h$  和  $k = h + 1$  时, 各个变迁的等效输入  $\rho_h$  和  $\rho_{h+1}$  相同. 由式 (13) 可知  $\rho'_{h+1} = \rho_{h+1} \ominus \rho_h = ((0,1), (0,1), \dots, (0,1))^T$ , 根据定理 2, 此时推理结束.

(ii) 再证  $k < h + 1$  成立

当  $k = j, j < h + 1$  时, 如果各个未触发变迁的等效输入小于对应变迁的阈值, 则有  $\rho_k'' = \rho'_k \otimes Th = ((0,1), (0,1), \dots, (0,1))^T$ , 此时变迁不再触发. 根据式 (14~16) 可知  $\theta^k$  不再变化, 所以  $k < h + 1$  时, 推理也可能结束.

综上所述, 定理得证.

定理 5 该推理算法的复杂度为  $O(nm^2)$ .

证明 假设 IFPN 模型中不存在回路,那么在一般情况下,推理算法的复杂度为  $O(knm)$ ,其中  $k$  为推理算法循环的次数,考虑最坏的情况,即推理循环了  $h + 1$  次( $h$  表示 IFPN 模型中的最长路径的变迁数目),那么总的算法复杂度为  $O((h + 1)nm) = O((m + 1)nm)$ ,即为  $O(nm^2)$ .

所以,假设 IFPR 集  $S$  中有  $n$  个命题, $m$  条规则,对应的 IFPN 模型有  $n$  个库所, $m$  个变迁,则基于 IFPN 的推理算法的复杂度为  $O(nm^2)$ .

### 5 实例验证及分析

规则集  $S_2$  如下:

$R_1$ : IF  $d_1$  AND  $d_2$  AND  $d_3$  THEN  $\neg d_6$  ( $\lambda_1 = (0.2, 0.6)$ ,  $CF_1 = (0.7, 0.2)$ )

$R_2$ : IF  $d_3$  AND  $d_4$  THEN  $d_7$  ( $\lambda_2 = (0.3, 0.6)$ ,  $CF_2 = (0.8, 0.1)$ )

$R_3$ : IF  $d_5$  THEN  $\neg d_7$  AND  $d_8$  ( $\lambda_3 = (0.1, 0.7)$ ,  $CF_3 = (0.6, 0.2)$ )

$R_4$ : IF  $d_6$  AND  $\neg d_7$  THEN  $d_9$  ( $\lambda_4 = (0.2, 0.5)$ ,  $CF_4 = (0.7, 0.1)$ )

$R_5$ : IF  $d_8$  OR  $\neg d_9$  THEN  $d_{10}$  ( $\lambda_5 = (0.1, 0.8)$ ,  $CF_5 = (0.5, 0.3)$ )

$R_6$ : IF  $\neg(\neg d_6)$  THEN  $d_6$  ( $\lambda_6 = (0.5, 0.4)$ ,  $CF_6 = (1, 0)$ )

其中  $R_5$  可以等效为如下两条规则:

$R_5^1$ : IF  $d_8$  THEN  $d_{10}$  ( $\lambda_5^1 = (0.1, 0.8)$ ,  $CF_5^1 = (0.5, 0.3)$ )

$R_5^2$ : IF  $\neg d_9$  THEN  $d_{10}$  ( $\lambda_5^2 = (0.1, 0.8)$ ,  $CF_5^2 = (0.5, 0.3)$ )

$R_6$  在图 6 所示的 IFPN 模型中,可以忽略. 规则集  $S_2$  的 IFPN 模型如图 6 所示.

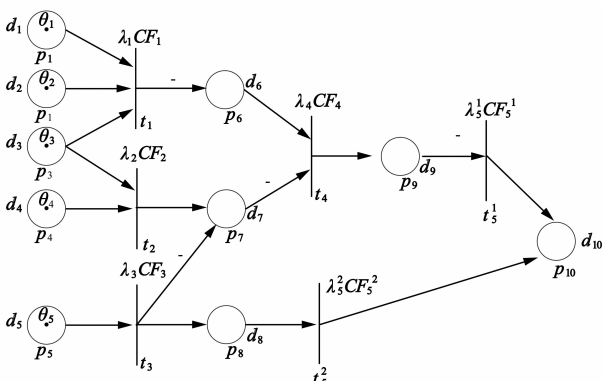


图6 规则集  $S_2$  的 IFPN 模型

已知  $n = 10, m = 6$ ,

$\theta^0 = (\theta_1^0, \theta_2^0, \dots, \theta_n^0)^T$   
 $= ((\mu_1^0, \gamma_1^0), (\mu_2^0, \gamma_2^0), \dots, (\mu_n^0, \gamma_n^0))^T$

$= ((0.6, 0.3), (0.7, 0.1), (0.5, 0.3), (0.7, 0.1), (0.8, 0.1), (0, 1), (0, 1), (0, 1), (0, 1), (0, 1))^T$

$I = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix}^T$

$IN = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T$

$O = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}^T$

$ON = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T$

$Th = (\lambda_1, \lambda_2, \lambda_3, \lambda_4, \lambda_5^1, \lambda_5^2)^T$   
 $= ((0.2, 0.6), (0.3, 0.6), (0.1, 0.7), (0.2, 0.5), (0.1, 0.8), (0.1, 0.8))^T$

$CF = \text{diag}(CF_1, CF_2, CF_3, CF_4, CF_5^1, CF_5^2)$   
 $= \text{diag}((0.7, 0.2), (0.8, 0.1), (0.6, 0.2), (0.7, 0.1), (0.5, 0.3), (0.5, 0.3))$

$\rho_0 = ((0, 1), (0, 1), \dots, (0, 1))^T$

推理过程如下:

(1) 推理开始, 令  $k = 1$

$\rho_1 = (\rho\theta_1^1, \rho\theta_2^1, \dots, \rho\theta_m^1)^T = ((0.5, 0.3), (0.5, 0.3), (0.8, 0.1), (0, 1), (0, 1), (0, 1))^T$ ,  $\rho_1' = \rho_1 \ominus \rho_0 = ((0.5, 0.3), (0.5, 0.3), (0.8, 0.1), (0, 1), (0, 1), (0, 1))^T$ ,  $\rho_1' \neq ((0, 1), (0, 1), \dots, (0, 1))^T$ , 推理继续.

$\rho_1'' = \rho_1' \otimes Th = ((0.5, 0.3), (0.5, 0.3), (0.8, 0.1), (0, 1), (0, 1), (0, 1), (0, 1))^T$ ,  $Y_1 = ((0, 1), (0, 1), (0, 1), (0, 1), (0, 1), (0, 1), (0.35, 0.44), (0.4, 0.37), (0.48, 0.28), (0, 1), (0, 1))^T$ ,  $\theta^1 = ((0.6, 0.3), (0.7, 0.1),$



$(0.5, 0.3), (0.7, 0.1), (0.8, 0.1), (0.35, 0.44), (0.4, 0.37), (0.48, 0.28), (0, 1), (0, 1))^T, \theta^1 \neq \theta^0$ , 推理继续.

(2) 此时  $k=2$

$\rho_2 = ((0.5, 0.3), (0.5, 0.3), (0.8, 0.1), (0.35, 0.44), (0, 1), (0.48, 0.28))^T, \rho'_2 = \rho_2 \ominus \rho_1 = ((0, 1), (0, 1), (0, 1), (0.35, 0.44), (0, 1), (0.48, 0.28))^T, \rho'_2 \neq ((0, 1), (0, 1), \dots, (0, 1))^T$  推理继续.

$\rho''_2 = \rho'_2 \otimes Th = ((0, 1), (0, 1), (0, 1), (0.35, 0.44), (0, 1), (0.48, 0.28))^T, Y_2 = ((0, 1), (0, 1), (0, 1), (0, 1), (0, 1), (0, 1), (0.245, 0.496), (0.24, 0.496))^T, \theta^2 = ((0.6, 0.3), (0.7, 0.1), (0.5, 0.3), (0.7, 0.1), (0.8, 0.1), (0.35, 0.44), (0.4, 0.37), (0.48, 0.28), (0.245, 0.496), (0.24, 0.496))^T, \theta^2 \neq \theta^1$ , 推理继续.

(3) 此时  $k=3$

$\rho_3 = ((0.5, 0.3), (0.5, 0.3), (0.8, 0.1), (0.35, 0.44), (0.496, 0.245), (0.48, 0.28))^T, \rho'_3 = \rho_3 \ominus \rho_2 = ((0, 1), (0, 1), (0, 1), (0, 1), (0.496, 0.245), (0, 1))^T, \rho'_3 \neq ((0, 1), (0, 1), \dots, (0, 1))^T$  推理继续.

$\rho''_3 = \rho'_3 \otimes Th = ((0, 1), (0, 1), (0, 1), (0, 1), (0.496, 0.245), (0, 1))^T, Y_3 = ((0, 1), (0, 1), (0, 1), (0, 1), (0, 1), (0, 1), (0.248, 0.4715))^T, \theta^3 = ((0.6, 0.3), (0.7, 0.1), (0.5, 0.3), (0.7, 0.1), (0.8, 0.1), (0.35, 0.44), (0.4, 0.37), (0.48, 0.28), (0.245, 0.496), (0.248, 0.4715))^T, \theta^3 \neq \theta^2$ , 推理继续.

(4) 此时  $k=4$

$\rho_4 = ((0.5, 0.3), (0.5, 0.3), (0.8, 0.1), (0.35, 0.44), (0.496, 0.245), (0.48, 0.28))^T, \rho'_4 = \rho_4 \ominus \rho_3 = ((0, 1), (0, 1), (0, 1), (0, 1), (0, 1), (0, 1))^T$ , 推理结束, 命题的最终真值为

$\theta = \theta^3 = ((0.6, 0.3), (0.7, 0.1), (0.5, 0.3), (0.7, 0.1), (0.8, 0.1), (0.35, 0.44), (0.4, 0.37), (0.48, 0.28), (0.245, 0.496), (0.248, 0.4715))^T$

通过实例验证可以发现本文提出的 IFPN 模型及推理算法与现有方法的不同之处主要有以下几点:

(1) 本文采用一个库所同时表示原命题和否命题, 该方法并未增加库所的数目, 避免了增加计算的复杂度.

(2) 算法的 Step3 通过增加“判断每个变迁的等效输入是否大于先前的输入”这一步, 抑制了变迁的重复触发, 避免了重复推理.

(3) 算法的 Step6 保留了事实, 避免了推理过程中

前提条件的丢失, 更符合实际推理过程.

(4) 与基于 FPN 的推理方法相比, 本文提出的基于 IFPN 的推理方法的克服了 FPN 推理结果隶属度单一的缺陷, 推理结果中增加了非隶属度, 对推理结果的表示更加细腻、全面, 更符合客观实际. 如命题  $d_{10}$  的真值为  $(0.248, 0.4715)$ , 表示  $d_{10}$  的隶属度为 0.248, 非隶属度为 0.4715.

## 6 结论

本文将 IFS 理论与 Petri 网理论相结合, 构建了 IFPN 模型用于知识的表示和推理, 解决了 IFPN 模型中否命题的表示, 基于 IFPN 推理过程中事实的保留和变迁的重复触发等问题.

实例验证表明本文构建的 IFPN 克服了现有 FPN 隶属度单一的缺陷, 由于增加了非隶属度, 使得 IFPN 对知识的表示更加准确; 提出的基于矩阵运算的推理算法, 在推理过程中充分利用了 Petri 网的图形描述和并行运算能力, 使得推理能够自动运行并且提高了推理的效率.

## 参考文献

- [1] 鲍培明. 基于 BP 网络的模糊 Petri 网的学习能力[J]. 计算机学报, 2004, 27(5): 695 - 702.  
Bao Peiming. Learning capability in fuzzy Petri nets based on BP net [J]. Chinese Journal of Computers, 2004, 27(5): 695 - 702. (in Chinese)
- [2] C G Looney. Fuzzy Petri nets for rule-based decision making [J]. IEEE Transactions on Systems, Man, and Cybernetics, 1998, 18(1): 178 - 183.
- [3] Shyi-Ming Chen, Jyh-Sheng Ke, Jin-Fu Chang. Knowledge representation using fuzzy Petrinets [J]. IEEE Transactions on Knowledge and Data Engineering, 1990, 2(3): 311 - 319.
- [4] Shyi-Ming Chen. Weighted fuzzy reasoning using weighted fuzzy Petri nets [J]. IEEE Transactions on Knowledge and Data Engineering, 2002, 14(2): 386 - 397.
- [5] Meimei Gao, MengChu Zhou, Xiaoguang Huang, Zhiming Wu. Fuzzy reasoning Petri nets [J]. IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans, 2003, 33(3): 314 - 324.
- [6] 汪洋, 林闯, 曲扬, 等. 含有否定命题逻辑推理的一致性模糊 Petri 网模型 [J]. 电子学报, 2006, 34(11): 1955 - 1960.  
Wang Yang, Lin Chuang, Qu Yang, et al. Consistent fuzzy Petri nets model for logic programs with negation [J]. Acta Electronica Sinica, 2006, 34(11): 1955 - 1960. (in Chi-

nese)

- [7] 贾立新, 薛钧义, 茹峰. 采用模糊 Petri 网的形式化推理算法及其应用[J]. 西安交通大学学报, 2003, 37(12): 1263 - 1266.
- Jia Lixin, Xun Junyi, Ru Feng. Fuzzy Petri net based formalized reasoning algorithm with applications[J]. Journal of Xi'an Jiaotong University, 2003, 37(12): 1263 - 1266. (in Chinese)
- [8] Xiaou Li, Wen Yu, Felipe Lara-Rosano. Dynamic knowledge inference and learning under adaptive fuzzy Petri net framework[J]. IEEE Transactions on Systems, Man, and Cybernetics-Part C: Applications and Reviews, 2000, 30(4): 442 - 450.
- [9] X Li, F Lara-Rosano. Adaptive fuzzy Petri nets for dynamic knowledge representation and inference[J]. Expert Systems with Applications, 2000, 19: 235 - 241.
- [10] Hu-Chen Liu, Long Liu, Qing-Lian Lin, et al. Knowledge acquisition and representation using fuzzy evidential reasoning and dynamic adaptive fuzzy Petri nets[J]. IEEE Transactions on Cybernetics, 2013, 43(3): 1059 - 1072.
- [11] Hu-Chen Liu, Qing-Lian Lin, Ling-Xiang Mao, et al. Dynamic adaptive fuzzy Petri nets for knowledge representation and reasoning[J]. IEEE Transactions on Systems, Man, and Cybernetics: Systems, 2013, 43(6): 1399 - 1410.
- [12] Victor R L Shen. Knowledge representation using high-level fuzzy Petri nets[J]. IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans, 2006, 36(6): 2120 - 2127.
- [13] Witold Pedrycz, Heloisa Camargo. Fuzzy timed Petri nets[J]. Fuzzy Sets and Systems, 2003, 140: 301 - 330.
- [14] Feng Zhou, Roger J Jiao, Qianli Xu, et al. User experience modeling and simulation for product ecosystem design based on fuzzy reasoning Petri nets[J]. IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans, 2012, 42(1): 201 - 212.
- [15] Jing Sun, Shi-Yin Qin, Yong-Hua Song. Faultdiagnosis of electric power systems based on fuzzy Petri nets[J]. IEEE Transactions on Power Systems, 2004, 19(4): 2053 - 2059.
- [16] Atanassov K. Intuitionistic fuzzy sets[J]. Fuzzy Sets and Systems, 1986, 20(1): 87 - 96.
- [17] Stephen JH Yang, Jeffrey JP Tsai, Chyun-Chyi Chen. Fuzzy rule base systems verification using high-level Petri nets[J]. IEEE Transactions on Knowledge and Data Engineering, 2003, 15(2): 457 - 473.
- [18] Derek L Nazareth. Investigating the applicability of Petri nets for rule-based system verification[J]. IEEE Transactions on Knowledge and Data Engineering, 1993, 4(3): 402 - 410.

## 作者简介



**孟飞翔** 男, 1986 年 11 月出生, 河南信阳人, 现为空军工程大学计算机应用专业博士研究生, 主要研究方向为智能信息处理.

E-mail: ttimo@163.com



**雷英杰** 男, 1956 年 11 月出生, 陕西华阴人, IEEE 高级会员. 现为空军工程大学教授, 博士生导师, 主要研究方向为智能信息处理与智能决策.

E-mail: leiyjie@163.com