

文章编号: 1001-0920(2013)12-1779-07

基于BP算法的IFPN参数优化方法

郑寇全^{1,2}, 雷英杰¹, 王睿¹, 王毅¹, 申晓勇¹

(1. 空军工程大学 防空反导学院, 西安 710051; 2. 中国人民解放军 68331 部队, 陕西 华阴 710042)

摘要: 针对直觉模糊 Petri 网(IFPN)模型自学习能力差的缺陷, 将神经网络中的 BP 误差反传算法引入 IFPN 模型的参数寻优过程, 提出一种基于此的参数优化方法. 该算法通过建立变迁点燃和直觉模糊推理的近似连续函数, 摆脱了参数对经验的依赖, 更加符合实际系统的需求, 同时使得 IFPN 具有较强的泛化能力和自适应功能, 推理结果更加准确可信. 最后通过典型实例验证了该参数优化方法的有效性和优越性.

关键词: 直觉模糊集; 直觉模糊 Petri 网; 产生式规则; BP 算法

中图分类号: TP18

文献标志码: A

Method for parameters optimization of IFPN based on BP algorithm

ZHENG Kou-quan^{1,2}, LEI Ying-jie¹, WANG Rui¹, WANG Yi¹, SHEN Xiao-yong¹

(1. Air Defense and Antimissile Institute, Air Force Engineering University, Xi'an 710051, China; 2. Unit of 68331, PLA, Huayin 710042, China. Correspondent: ZHENG Kou-quan, E-mail: zhengkouquan0421@163.com)

Abstract: In order to improve the self-learning capability of intuitionistic fuzzy Petri nets(IFPN), a novel parameters optimization method is proposed, in which the back propagation algorithm of neural net is introduced to the parameters-optimized process of IFPN. By constructing the approximate continuous function of transition firing and intuitionistic fuzzy reasoning, the method makes the parameters get rid of the dependence upon experience, which makes the parameters adjust the fact instance better. Meanwhile, the IFPN model can own better generalization performance and self-adjusting ability, and the reasoning results are more accurate and reliable as well. Finally, the classical instance verifies the effectiveness and superiority of the proposed method.

Key words: intuitionistic fuzzy set; intuitionistic fuzzy Petri nets; production rule; BP algorithm

0 引言

在客观世界和现实生活中, 由于系统的随机性、缺乏相关属性(参数)或信息不精确等模糊不确定性的广泛存在, 许多知识难以用精确的方法描述. 直觉模糊产生式规则用 If-Then 结构的直觉模糊条件句表示不确定知识, 克服了经典模糊理论隶属度单一的缺陷, 为不确定信息的描述和处理提供了新的思路和方法^[1]. 直觉模糊 Petri 网是基于直觉模糊产生式规则知识库系统的良好建模工具, 其利用直觉模糊数描述各个库所的 Token 值以及变迁的可信度和阈值, 在输入和输出矩阵中加入权重系数和可信度因子, 不仅解决了各输入库所对变迁作用程度不同的问题, 而且将不确定知识的表达与直觉模糊推理融为一体, 从而更加符合人类的思维和认知方式, 系统的可靠性得到了进一步提高, 但自学习调整能力差是其固有的缺点, 这

在很大程度上降低了 IFPN 模型的知识推理和泛化性能^[2]. 因此, 如何经过样本数据的学习和训练, 建立相适应的直觉模糊推理算法, 使 IFPN 的所有参数摆脱对经验的依赖, 更加符合不确定性推理的实际情况便成为亟待研究解决的重要问题.

神经网络具有较强的自适应和自学习能力, 其 BP 误差反传算法可逼近任意的非线性映射关系, 并能快速收敛到相应曲面的极值点, 是一种常用的首选优化策略^[3]. 模糊 Petri 网(FPN)成功地将 BP 网络算法引入到模型参数优化过程中, 较大幅度地提高了系统的泛化能力, 取得了一系列的研究成果: Tsang 等^[4]通过修改传统的 BP 算法来实现前向推理和反向权值调整, 提出了在系统没有阈值条件下的权值和可信度自学习算法; Li 等^[5-6]利用类神经网络进行权值的调整优化, 建立了自适应 FPN 模型; 文献[3]采用分层

收稿日期: 2012-09-16; 修回日期: 2012-12-19.

基金项目: 国家自然科学基金项目(61272011, 60773209); 国家重点实验室基金项目(2012ADL-DW0301).

作者简介: 郑寇全(1983-), 男, 博士生, 从事智能信息处理的研究; 雷英杰(1956-), 男, 教授, 博士生导师, 从事智能信息处理等研究.

思想提出了基于BP网络的FPN自学习算法,可实现系统参数的整体寻优;文献[7-9]扩大模糊“或”规则的适用范围,进一步提高了FPN模型推理的准确度;文献[10]提出了一种基于元模型的FPN反向传播学习算法;文献[11-14]基于神经网络研究了FPN的自学习能力,并成功应用于故障诊断、预测和评估等领域。IFPN结合了直觉模糊理论较强的模糊描述能力和FPN模型的并行动态推理功能,在提高模型处理实际问题能力的同时,也增加了参数优化的复杂度,而且由于非隶属度函数的作用,如果仅简单借鉴FPN的参数优化方案会使得算法收敛速度慢且易陷入局部最优。鉴于此,本文从构建变迁点燃和直觉模糊推理近似连续函数入手,直觉模糊化BP误差反传算法,提出一种自适应的IFPN参数优化方法,降低了系统的复杂度,提高了模型的泛化能力,使推理结果更准确可靠。最后通过实例验证了该方法的正确性和合理性。

1 直觉模糊Petri网及其推理函数

1.1 直觉模糊Petri网

定义1 (IFPN模型)^[15] 直觉模糊Petri网的模型结构可用如下的六元组表示:

$$\text{IFPN} = (P, T, I, O, \tau, \theta). \quad (1)$$

其中: $P = \{p_1, p_2, \dots, p_n\}$ 表示有限库所集合, 每个库所对应一个命题; $T = \{t_1, t_2, \dots, t_m\}$ 表示有限变迁集合, 每个变迁对应一条规则; $I: P \times T \rightarrow [0, 1]$ 指 $n \times m$ 维加权输入矩阵, 其元素 a_{ij} 满足: 若 p_i 是 T_j 的输入, 则 $a_{ij} = \omega_{ij}$, 其中 ω_{ij} 表示输入库所 p_i 对变迁 T_j 产生影响的权值, 且 $\sum_{i=1}^n \omega_{ij} = 1, j \in [1, m]$, 否则 $a_{ij} = 0$; $O: P \times T \rightarrow [0, 1]$ 指 $n \times m$ 维输出矩阵, 其元素 b_{ij} 满足: 若 p_i 是 T_j 的输出, 则 $b_{ij} = C_j$, 其中 $C_j = \langle C\mu_j, C\gamma_j \rangle$ 表示变迁 T_j 的可信度, $C\mu_j$ 和 $C\gamma_j$ 分别表示可信度的置信度和非置信度, 否则 $b_{ij} = \langle 0, 1 \rangle$; $\tau = \{\tau_1, \tau_2, \dots, \tau_m\}$ 表示变迁阈值, 即启动推理规则的条件, 其中 $\tau_j = \langle \alpha_j, \beta_j \rangle$, 且 $0 < \alpha_j + \beta_j \leq 1, 0 < \alpha_j \leq 1, 0 < \beta_j \leq 1, \alpha_j$ 为置信度阈值, β_j 为非置信度阈值; $\theta: P \rightarrow [0, 1]$ 为库所 P 的关联函数, 表示命题的直觉模糊真值, 初始Token值记作 $\theta^0 = \{\theta_1^0, \theta_2^0, \dots, \theta_n^0\}$, 其中 $\theta_i^0 = \langle \theta\mu_i^0, \theta\gamma_i^0 \rangle$ 为直觉模糊数。

产生式规则表示法是一种常用的因果关系知识表达形式, 它通过变迁是否点燃来判断规则的激活, 直观、自然, 便于推理。文献[15]给出了基于直觉模糊“合”取和“析”取产生式规则的IFPN模型。为了将判断变迁是否使能转化成一个连续函数自变量是否满足一定要求的问题, 并使得直觉模糊推理结果是便于求导的连续函数, 本文直觉模糊化文献[5]中的Sigmoid函数, 建立直觉模糊推理函数。

1.2 直觉模糊推理函数

设 $f(x, y)$ 是一个二元函数, b 为常量, $f(x, y)$ 的表达式可表示为

$$f(x, y) = [(1 + e^{-b(x-\tau_1)})(1 + e^{b(y-\tau_2)})]^{-1}. \quad (2)$$

其中: 当 b 足够大时, 若 $x > \tau_1$ 且 $y < \tau_2$, $e^{-b(x-\tau_1)} \rightarrow 0, e^{b(y-\tau_2)} \rightarrow 0$, 则 $f(x, y) \rightarrow 1$; 若 $x < \tau_1$ 或 $y > \tau_2$, $e^{-b(x-\tau_1)} \rightarrow \infty$ 或 $e^{b(y-\tau_2)} \rightarrow \infty$, 则 $f(x, y) \rightarrow 0$ 。显然, 连续函数 $f(x, y)$ 的这种二值性可作为判断变迁是否使能的标志。

1.2.1 变迁点燃连续函数

参照文献[15]中直觉模糊“合”取产生式规则对应的IFPN模型, 令 $\tau_1 = \tau\mu_i, \tau_2 = \tau\gamma_i, x = x_\mu = \sum_{j=1}^n M_\mu(p_{ij}) \times \omega_{ij}, y = x_\gamma = \sum_{j=1}^n M_\gamma(p_{ij}) \times \omega_{ij}$ 。根据式(2), 函数 $f(x_\mu, x_\gamma)$ 可建立变迁使能的判断, 即当 b 足够大时, 若 $x_\mu > \tau\mu_i$ 且 $x_\gamma < \tau\gamma_i$, 则表示变迁 t_i 使能; 否则, 变迁 t_i 没有使能。因此, 变迁点燃连续函数可表示为

$$F = f(x_\mu, x_\gamma) \cdot C \cdot \sum_{j=1}^n M_\mu(p_{ij}) \times \omega_{ij}, \quad (3)$$

其输出库所 p 的Token值可定义为

$$\begin{cases} M_\mu(p) = f(x_\mu, x_\gamma) \cdot C \cdot \sum_{j=1}^n M_\mu(p_{ij}) \times \omega_{ij}, \\ M_\gamma(p) = 1 - M_\mu(p) - M_\pi(p). \end{cases} \quad (4)$$

其中: $M_\mu(p)$ 、 $M_\gamma(p)$ 和 $M_\pi(p)$ 分别为库所 p 的隶属度、非隶属度和犹豫度值, 且 $M_\mu(p)$ 、 $M_\gamma(p)$ 、 $M_\pi(p) \in [0, 1]$; $C_i = (1 + C\mu_i - C\gamma_i)/2$ 。

对于直觉模糊“析”取规则对应的IFPN模型, 同样可以建立最大/最小运算连续函数。

1.2.2 最大/最小运算连续函数

假设 $y(x) = 1/(1 + e^{-b(x-k)})$, 若 x_1, x_2, x_3 为变迁使能时的输出值, 则当 b 足够大时, 显然有

$$\begin{aligned} t &= \max(x_1, x_2) \approx \frac{x_1}{1 + e^{-b(x_1-x_2)}} + \frac{x_2}{1 + e^{-b(x_2-x_1)}}, \\ h &= \max(x_1, x_2, x_3) = \max(t, x_3) \approx \frac{t}{1 + e^{-b(t-x_3)}} + \frac{x_3}{1 + e^{-b(x_3-t)}}. \end{aligned}$$

同理, 假设 $\Phi(x) = 1/(1 + e^{b(x-k)})$, 当 b 足够大时, 下列结论显然成立:

$$\begin{aligned} t &= \min(x_1, x_2) \approx \frac{x_1}{1 + e^{b(x_1-x_2)}} + \frac{x_2}{1 + e^{b(x_2-x_1)}}, \\ h &= \min(x_1, x_2, x_3) = \min(t, x_3) \approx \frac{t}{1 + e^{b(t-x_3)}} + \frac{x_3}{1 + e^{b(x_3-t)}}. \end{aligned}$$

由此可见, 经过每个变迁 t_i , IFPN 输出库所 p 的Token值可定义为

$$\begin{cases} M_\mu(p) = f(x_\mu, x_\gamma) \cdot C \cdot M_\mu(p_{ij}), \\ M_\gamma(p) = 1 - M_\mu(p) - M_\pi(p). \end{cases} \quad (5)$$

以此类推, 当有多个变迁使能时, 利用最大/最小运算连续函数, 可使输出库所 p 总对应一个最大隶属度和最小非隶属度的直觉模糊函数值.

2 BP 误差反传算法在 IFPN 中的应用

IFPN 模型可看作是由以变迁为中心的节点构成的. 仿照神经网络中的 BP 算法, 在每一层上对每个 IFPN 节点用误差反传方法设计调整参数. 设 IFPN 模型分为 l 层, 有 n 个库所, d 个变迁, 其中有 b 个终止库所 $p_k, k = 1, 2, \dots, b$, 用 r 批样本数据进行学习, 每批输入样本为 s , 其对应的误差函数为

$$E_s = \frac{\sum_{k=1}^b [(M_\mu(p_k)^s - M_\mu^*(p_k)^s)^2 + (M_\gamma(p_k)^s - M_\gamma^*(p_k)^s)^2]}{2}. \quad (6)$$

其中: $\langle M_\mu(p_k)^s, M_\gamma(p_k)^s \rangle, \langle M_\mu^*(p_k)^s, M_\gamma^*(p_k)^s \rangle$ 分别为样本 s 作用下终止库所 p_k 的隶属度和非隶属度函数的实际和期望输出. 通过求均值便可计算 r 批样本的平均误差.

2.1 直觉模糊扩展的 BP 误差反传算法

IFPN 模型各参数应按 E_s 函数梯度变化的反方向进行调整, 从而使网络收敛. 根据式 (4)~(6), 按照最速下降法, 可得到第 l 层各参数的修正公式为

$$\Delta\omega_{ij}^l = -\eta \frac{\partial E_s}{\partial \omega_{ij}^l} = -\eta \left(\frac{\partial E_s}{\partial M_\mu^l(p_k)^s} \frac{\partial M_\mu^l(p_k)^s}{\partial \omega_{ij}^l} + \frac{\partial E_s}{\partial M_\gamma^l(p_k)^s} \frac{\partial M_\gamma^l(p_k)^s}{\partial \omega_{ij}^l} \right); \quad (7)$$

$$\begin{cases} \Delta C\mu_i^l = -\eta \frac{\partial E_s}{\partial C\mu_i^l} = \\ -\eta \left(\frac{\partial E_s}{\partial M_\mu^l(p_k)^s} \frac{\partial M_\mu^l(p_k)^s}{\partial C\mu_i^l} + \frac{\partial E_s}{\partial M_\gamma^l(p_k)^s} \frac{\partial M_\gamma^l(p_k)^s}{\partial C\mu_i^l} \right), \\ \Delta C\gamma_i^l = -\eta \frac{\partial E_s}{\partial C\gamma_i^l} = \\ -\eta \left(\frac{\partial E_s}{\partial M_\mu^l(p_k)^s} \frac{\partial M_\mu^l(p_k)^s}{\partial C\gamma_i^l} + \frac{\partial E_s}{\partial M_\gamma^l(p_k)^s} \frac{\partial M_\gamma^l(p_k)^s}{\partial C\gamma_i^l} \right); \end{cases} \quad (8)$$

$$\begin{cases} \Delta\tau\mu_i^l = -\eta \frac{\partial E_s}{\partial \tau\mu_i^l} = \\ -\eta \left(\frac{\partial E_s}{\partial M_\mu^l(p_k)^s} \frac{\partial M_\mu^l(p_k)^s}{\partial \tau\mu_i^l} + \frac{\partial E_s}{\partial M_\gamma^l(p_k)^s} \frac{\partial M_\gamma^l(p_k)^s}{\partial \tau\mu_i^l} \right), \\ \Delta\tau\gamma_i^l = -\eta \frac{\partial E_s}{\partial \tau\gamma_i^l} = \\ -\eta \left(\frac{\partial E_s}{\partial M_\mu^l(p_k)^s} \frac{\partial M_\mu^l(p_k)^s}{\partial \tau\gamma_i^l} + \frac{\partial E_s}{\partial M_\gamma^l(p_k)^s} \frac{\partial M_\gamma^l(p_k)^s}{\partial \tau\gamma_i^l} \right). \end{cases} \quad (9)$$

其中: $M_\mu^l(p_k)^s$ 和 $M_\gamma^l(p_k)^s$ 表示第 l 层终止库所 p_k 的隶属度与非隶属度函数值; $\frac{\partial E_s}{\partial \omega_{ij}^l}$ 为 E_s 关于 ω_{ij}^l 的增长率, 为使误差减小, 取 $\Delta\omega_{ij}^l$ 与它成反比, 若 $\frac{\partial E_s}{\partial \omega_{ij}^l} > 0$, 则系统当前所处位置在极小点的右侧, ω_{ij}^l 的值应减小, 否则系统所处位置在极小点的左侧, ω_{ij}^l 的值应增大; $\frac{\partial E_s}{\partial C\mu_i^l}, \frac{\partial E_s}{\partial C\gamma_i^l}$ 分别为 E_s 关于置信度与非置信度的增长率; $\frac{\partial E_s}{\partial \tau\mu_i^l}, \frac{\partial E_s}{\partial \tau\gamma_i^l}$ 分别为 E_s 关于置信度阈值与非置信度阈值的增长率; $\eta \in (0, 1)$ 为学习系数.

假设 $t_i^l \in T_l (i=1, 2, \dots, d)$ 是 IFPN 第 l 层的一个变迁, t_i^l 输入弧上的权重系数分别为 $\omega_{i1}^l, \omega_{i2}^l, \dots, \omega_{im}^l$, 其阈值与可信度分别表示为 $t_i^l = \langle \tau\mu_i^l, \tau\gamma_i^l \rangle, C_i^l = \langle C\mu_i^l, C\gamma_i^l \rangle$. 若 $p_k^l \in O(t_i^l)$, 则 p_k^l 为终止库所, 从而可以计算各参数一阶梯度.

1) 权重系数

$$\frac{\partial E_s}{\partial \omega_{ij}^l} = \frac{\partial E_s}{\partial M_\mu^l(p_k)^s} \frac{\partial M_\mu^l(p_k)^s}{\partial \omega_{ij}^l} + \frac{\partial E_s}{\partial M_\gamma^l(p_k)^s} \frac{\partial M_\gamma^l(p_k)^s}{\partial \omega_{ij}^l} = G(x_\mu, x_\gamma) \frac{\Delta M_\mu - \Delta M_\gamma}{f(x_\mu)g(x_\gamma)}; \quad (10)$$

2) 可信度

$$\begin{cases} \frac{\partial E_s}{\partial C\mu_i^l} = \frac{\Delta M_\mu x_\mu(t_i)}{2f(x_\mu) \cdot f(x_\gamma)} - \frac{\Delta M_\gamma x_\mu(t_i)}{f(x_\mu) \cdot 2g(x_\gamma)}, \\ \frac{\partial E_s}{\partial C\gamma_i^l} = -\frac{\Delta M_\mu x_\mu(t_i)}{2f(x_\mu) \cdot f(x_\gamma)} + \frac{\Delta M_\gamma x_\mu(t_i)}{f(x_\mu) \cdot 2g(x_\gamma)}; \end{cases} \quad (11)$$

3) 阈值

$$\begin{cases} \frac{\partial E_s}{\partial \tau\mu_i^l} = \frac{-bC_i x_\mu(t_i)(\Delta M_\mu - \Delta M_\gamma)}{(f(x_\mu))^2 g(x_\gamma)} e^{-b(x_\mu(t_i) - \tau\mu_i)}, \\ \frac{\partial E_s}{\partial \tau\gamma_i^l} = \frac{bC_i x_\mu(t_i)(\Delta M_\mu - \Delta M_\gamma)}{f(x_\mu)(g(x_\gamma))^2} e^{b(x_\gamma(t_i) - \tau\gamma_i)}. \end{cases} \quad (12)$$

其中

$$\begin{aligned} \Delta M_\mu &= M_\mu(p_k)^s - M_\mu^*(p_k)^s, \\ \Delta M_\gamma &= M_\gamma(p_k)^s - M_\gamma^*(p_k)^s, \\ f(x_\mu) &= 1 + e^{-b(x_\mu(t_i) - \tau\mu_i)}, \\ g(x_\gamma) &= 1 + e^{b(x_\gamma(t_i) - \tau\gamma_i)}, \\ G(x_\mu, x_\gamma) &= \\ &C_i \left[M_\mu(p_k) + \frac{bM_\mu(p_k)}{f(x_\mu)} e^{-b(x_\mu(t_i) - \tau\mu_i)} - \right. \\ &\left. \frac{bM_\gamma(p_k)}{g(x_\gamma)} e^{b(x_\gamma(t_i) - \tau\gamma_i)} \right]. \end{aligned}$$

若 $t_i^{l-1} \in T_{l-1}$ 是 IFPN 第 $l-1$ 层的一个变迁, 则

$p_x^{l-1} \in O(t_i^{l-1})$; 如果 p_x^{l-1} 为终止库所, 则可参照上述方法计算, 否则, $\exists t_i^l \in T_l, p_x^{l-1} \in I(t_i^l), p_k^l \in O(t_i^l), p_k^l$ 为终止库所. 令 $\delta_\mu^{(l)} = \frac{\partial E_s}{\partial M_\mu^l(p_k)^s}, \delta_\gamma^{(l)} = \frac{\partial E_s}{\partial M_\gamma^l(p_k)^s}$, 则第 $l-1$ 层变迁各参数的一阶梯度可表示如下:

1) 权重系数

$$\frac{\partial E_s}{\partial \omega_{ij}^l} = \frac{\partial M_\mu^{l-1}(p_k)^s}{\partial \omega_{ij}^{l-1}} \delta_\mu^{(l-1)} + \frac{\partial M_\gamma^{l-1}(p_k)^s}{\partial \omega_{ij}^{l-1}} \delta_\gamma^{(l-1)};$$

2) 可信度

$$\begin{cases} \frac{\partial E_s}{\partial C\mu_i^l} = \frac{\partial M_\mu^{l-1}(p_k)^s}{\partial C\mu_i^{l-1}} \delta_\mu^{(l-1)} + \frac{\partial M_\gamma^{l-1}(p_k)^s}{\partial C\mu_i^{l-1}} \delta_\gamma^{(l-1)}, \\ \frac{\partial E_s}{\partial C\gamma_i^l} = \frac{\partial M_\mu^{l-1}(p_k)^s}{\partial C\gamma_i^{l-1}} \delta_\mu^{(l-1)} + \frac{\partial M_\gamma^{l-1}(p_k)^s}{\partial C\gamma_i^{l-1}} \delta_\gamma^{(l-1)}; \end{cases}$$

3) 阈值

$$\begin{cases} \frac{\partial E_s}{\partial \tau\mu_i^l} = \frac{\partial M_\mu^{l-1}(p_k)^s}{\partial \tau\mu_i^{l-1}} \delta_\mu^{(l-1)} + \frac{\partial M_\gamma^{l-1}(p_k)^s}{\partial \tau\mu_i^{l-1}} \delta_\gamma^{(l-1)}, \\ \frac{\partial E_s}{\partial \tau\gamma_i^l} = \frac{\partial M_\mu^{l-1}(p_k)^s}{\partial \tau\gamma_i^{l-1}} \delta_\mu^{(l-1)} + \frac{\partial M_\gamma^{l-1}(p_k)^s}{\partial \tau\gamma_i^{l-1}} \delta_\gamma^{(l-1)}. \end{cases}$$

其中

$$\delta_\mu^{(l-1)} = \frac{\partial M_\mu^l(p_k)^s}{\partial M_\mu^{l-1}(p_k)^s} \delta_\mu^{(l)}, \delta_\gamma^{(l-1)} = \frac{\partial M_\gamma^l(p_k)^s}{\partial M_\gamma^{l-1}(p_k)^s} \delta_\gamma^{(l)}.$$

在模型推理过程中存在 $p_x \rightarrow p_k$ 的连续函数, 因此 $\frac{\partial M_\mu^l(p_k)^s}{\partial M_\mu^{l-1}(p_k)^s}$ 有解. 以此类推, 依次对 $h = l-2, l-3, \dots, 1$ 层反向递推, 计算 $\frac{\partial E_s}{\partial \omega_{ij}^h}, \frac{\partial E_s}{\partial C\mu_i^h}, \frac{\partial E_s}{\partial C\gamma_i^h}, \frac{\partial E_s}{\partial \tau\mu_i^h}$ 和 $\frac{\partial E_s}{\partial \tau\gamma_i^h}$. 求得所需一阶梯度后, 每个变迁的参数调整学习过程可定义如下:

1) 权重系数

$$\omega_{ij}^h(q+1) = \omega_{ij}^h(q) - \eta \frac{\partial E_s}{\partial \omega_{ij}^h};$$

2) 可信度

$$\begin{cases} C\mu_i^h(q+1) = C\mu_i^h(q) - \eta \frac{\partial E_s}{\partial C\mu_i^h}, \\ C\gamma_i^h(q+1) = C\gamma_i^h(q) - \eta \frac{\partial E_s}{\partial C\gamma_i^h}; \end{cases}$$

3) 阈值

$$\begin{cases} \tau\mu_i^h(q+1) = \tau\mu_i^h(q) - \eta \frac{\partial E_s}{\partial \tau\mu_i^h}, \\ \tau\gamma_i^h(q+1) = \tau\gamma_i^h(q) - \eta \frac{\partial E_s}{\partial \tau\gamma_i^h}. \end{cases}$$

其中: q 表示学习次数; $\omega_{ij}^h(q)$ 表示第 q 次学习后第 h 层对应变迁 t_i 的输入库所 p_x 的权重, 且 $\sum_{j=1}^m \omega_{ij}^h(q+1) = 1$, 其他参数含义类似.

2.2 学习率调整

在 IFPN 模型中应用 BP 网络算法的主要缺陷是收敛速度慢. 因此, 在训练过程中引进自适应学习速率法, 以增强算法的鲁棒性, 减少个别受扰点误差大的影响. 根据网络的输出误差, 动态调整学习速率为

$$\eta(q) = \eta_0 / [1 - SE_s(q)]. \quad (13)$$

其中: η_0 为初始学习率, $SE_s(q)$ 为第 q 次学习后终止库所的误差和.

在参数学习过程中, 学习率 η 的选择很重要. η 大, 收敛速度快, 但可能不稳定引起振荡; η 小, 虽然可避免不稳定, 但收敛速度慢. 解决这一问题的最简便方法是加入“动量项”. 因此, 上节的修正公式可调整为

$$\omega_{ij}^h(q+1) = \omega_{ij}^h(q) - \eta \frac{\partial E_s}{\partial \omega_{ij}^h} + \xi \omega_{ij}^h(q); \quad (14)$$

$$\begin{cases} C\mu_i^h(q+1) = C\mu_i^h(q) - \eta \frac{\partial E_s}{\partial C\mu_i^h} + \xi C\mu_i^h(q), \\ C\gamma_i^h(q+1) = C\gamma_i^h(q) - \eta \frac{\partial E_s}{\partial C\gamma_i^h} + \xi C\gamma_i^h(q); \end{cases} \quad (15)$$

$$\begin{cases} \tau\mu_i^h(q+1) = \tau\mu_i^h(q) - \eta \frac{\partial E_s}{\partial \tau\mu_i^h} + \xi \tau\mu_i^h(q), \\ \tau\gamma_i^h(q+1) = \tau\gamma_i^h(q) - \eta \frac{\partial E_s}{\partial \tau\gamma_i^h} + \xi \tau\gamma_i^h(q). \end{cases} \quad (16)$$

其中 $\xi \in [0, 1)$ 为调节因子. 为避免振荡, 同时加快收敛速度, 若 $\xi \leq 0$, 则取 $\xi = 0$.

2.3 IFPN 模型学习和训练算法

下面根据直觉模糊化的 BP 误差反传算法, 结合 IFPN 推理过程给出 IFPN 模型的学习和训练步骤. 本文提出的算法适用于无回路 IFPN 模型, 而无回路的 IFPN 模型可以转化为层次结构^[15]. 因此, 首先给出 IFPN 模型分层算法.

算法 1 IFPN 模型分层算法.

输入: 库所 P , 变迁 T , 输入矩阵 I 和输出矩阵 O ;
输出: $T_i, \langle x_\mu(t_i), x_\gamma(t_i) \rangle$.

Step 1: 若 $p_k \in R(p_k)$, 则该 IFPN 模型中有回路, 算法终止, 退出, 否则运行 Step 2.

Step 2: 初始化各输入变量, 给定集合 $H = \{p_k\}$, p_k 为初始库所. $\forall p_k \in \{P - H\}$, 令 $\langle M_\mu(p_k), M_\gamma(p_k) \rangle = \langle 0, 1 \rangle$, 设 i 为模型的分层数, 初始值为 1.

Step 3: 建立变迁集合 $T_i = \{t_j \in T \mid \forall p_k \in I(t_j), p_k \in H\}$.

Step 4: 若 $t_j \in T_i, t \in \{T - T_i\}, \exists p \in \{O(t_j) \cap O(t)\}$, 则 $T_i = \{T_i - \{t_j\}\}$.

Step 5: 令 $H = H \cup \{p \mid p \in O(t_j), t_j \in T_j\}$, 即将 T_i 中所有变迁的输出库所插在集合 H 中.

Step 6: 若 $T_i = \emptyset$, 则需对模型添加虚库所和虚变迁, 返回 Step 3, 否则 $T = \{T - \{T_i\}\}$.

Step 7: 若 $T = \emptyset$, 则 $i = l$, 否则 $i = i + 1$, 返回 Step 2 循环执行.

Step 8: 利用式 (4) 和 (5) 逐层点燃 T_i 中的所有变迁, 计算除初始库所以外所有库所连续函数表达式,

算法终止.

算法 1 用来对 IFPN 模型分层, 并给出各层库所 Token 值的计算函数表达式, 通过分析可知其算法的时间复杂度为 $O(n^2)$. 模型按照变迁结构分为 l 层, 每一层的变迁集合为 T_i ; 虚库所和虚变迁在模型中只起一个中间过渡作用, 它们的增加不会影响规则库系统, 虚变迁对应的阈值为 $\langle 0, 1 \rangle$, 确信度为 $\langle 1, 0 \rangle$. 对于分好层的 IFPN 模型, 可以给出基于 BP 算法的 IFPN 模型参数优化方法.

算法 2 BP-IFPN 算法.

输入: 初始 Token 值 $\theta^{(0)}$, 权重系数, 变迁阈值以及各参数的初始值, 输出阈值 ε ;

输出: 终止 Token 值 θ 以及各参数的学习值, 学习次数 q .

Step 1: 对需要学习的参数赋予初值, 预处理 r 批样本数据, 令 $q = 0$.

Step 2: 对 r 批样本数据, 利用算法 1 的输出结果计算各个库所的 Token 值.

Step 3: 参照式 (6) 计算样本平均误差函数 E , 若 $E < \varepsilon$, 则跳到 Step 5, 否则转 Step 4.

Step 4: 利用算法 1 的输出结果, 参照 2.1 节对参数进行调整. 令 $q = q + 1$, 返回 Step 2.

Step 5: 获取各参数调整后的值、终止库所和总误差函数值以及学习次数, 算法终止.

算法 2 的时间复杂度和样本数、学习次数、模型层数与每层变迁数均相关, 若模型层数已知, 则其时间复杂度为 $O(n^3)$.

3 算例分析

为验证算法的可行性, 参照文献 [8] 的实例给出 IFPN 模型的学习和训练步骤, 并对推理结果进行比较分析.

设已知库所 $p_1, p_2, p_3, p_4, p_5, p_6, p_7$ 和 p_8 各自对应着专家系统中的一个命题, 且相互之间存在如下直觉模糊产生式规则:

- R_1 : If p_1 , Then $p_2(C_2, \tau_2)$;
- R_2 : If p_1 or p_2 , Then $p_3(C_1, \tau_1, C_3, \tau_3)$;
- R_3 : If p_3 and p_4 and p_5 , Then $p_6(\omega_{41}, \omega_{42}, C_4, \tau_4)$;
- R_4 : If p_6 and p_7 , Then $p_8(\omega_{51}, \omega_{52}, C_5, \tau_5)$.

按照 IFPN 模型知识表示方法, 由上述规则可建立图 1 所示的 IFPN 模型.

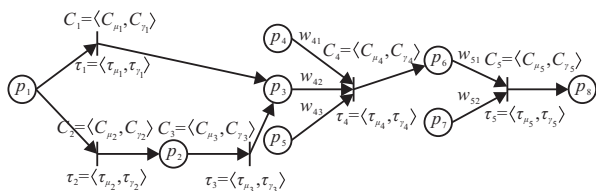


图 1 直觉模糊 Petri 网模型

3.1 模型分层及推理

首先, 利用算法 1 对模型进行分层. 因为 τ_1 和 τ_3 对应着同一个输出库所 p_3 , 所以 Step 4 将第 1 层中的 t_1 去掉, 且 $T_1 \neq \emptyset$, 故不需要增加虚库所和虚变迁, 只将 t_1 和 t_3 划分在同一层. 如图 2 所示, 经过分层, $T_1 = \{t_2\}, T_2 = \{t_1, t_3\}, T_3 = \{t_4\}, T_4 = \{t_5\}$. 为简单起见, 令 $M_\pi = 0$, 则该 IFPN 模型的直觉模糊推理顺序可表示如下:

1) 先点燃 T_1 中的变迁 t_2 , 有 $\langle x_\mu, x_\gamma \rangle = \langle M_\mu(p_1), M_\gamma(p_1) \rangle, \langle \tau_\mu, \tau_\gamma \rangle = \langle \tau\mu_2, \tau\gamma_2 \rangle$, 则

$$\begin{cases} M_\mu(p_2) = \frac{M_\mu(p_1)C_2}{(1 + e^{-b(M_\mu(p_1) - \tau\mu_2)})(1 + e^{b(M_\gamma(p_1) - \tau\gamma_2)})}, \\ M_\gamma(p_2) = 1 - \frac{M_\mu(p_1)C_2}{(1 + e^{-b(M_\mu(p_1) - \tau\mu_2)})(1 + e^{b(M_\gamma(p_1) - \tau\gamma_2)})}. \end{cases}$$

2) 点燃 T_2 中的变迁 t_1 和 t_3 , 令

$$M_{1_\mu}(p_3) = \frac{M_\mu(p_1)C_1}{(1 + e^{-b(M_\mu(p_1) - \tau\mu_1)})(1 + e^{b(M_\gamma(p_1) - \tau\gamma_1)})},$$

$$M_{2_\mu}(p_3) = \frac{M_\mu(p_2)C_3}{(1 + e^{-b(M_\mu(p_2) - \tau\mu_3)})(1 + e^{b(M_\gamma(p_2) - \tau\gamma_3)})},$$

则 $M_{1_\gamma}(p_3) = 1 - M_{1_\mu}(p_3), M_{2_\gamma}(p_3) = 1 - M_{2_\mu}(p_3)$. 由直觉模糊逻辑推理理论可知

$$\begin{cases} M_\mu(p_3) = \frac{M_{1_\mu}(p_3)}{1 + e^{-b(M_{1_\mu}(p_3) - M_{2_\mu}(p_3))}} + \frac{M_{2_\mu}(p_3)}{1 + e^{-b(M_{2_\mu}(p_3) - M_{1_\mu}(p_3))}}, \\ M_\gamma(p_3) = \frac{M_{1_\gamma}(p_3)}{1 + e^{b(M_{1_\gamma}(p_3) - M_{2_\gamma}(p_3))}} + \frac{M_{2_\gamma}(p_3)}{1 + e^{b(M_{2_\gamma}(p_3) - M_{1_\gamma}(p_3))}}. \end{cases}$$

3) 点燃 T_3 中的变迁 t_4 , 有 $x_\mu(t_4) = M_\mu(p_4) \times \omega_{41} + M_\mu(p_3) \times \omega_{42} + M_\mu(p_5) \times \omega_{43}, x_\gamma(t_4) = M_\gamma(p_4) \times \omega_{41} + M_\gamma(p_3) \times \omega_{42} + M_\gamma(p_5) \times \omega_{43}, \langle \tau_\mu, \tau_\gamma \rangle = \langle \tau\mu_4, \tau\gamma_4 \rangle$, 则

$$\begin{cases} M_\mu(p_6) = \frac{x_\mu(t_4)C_4}{(1 + e^{-b(x_\mu(t_4) - \tau\mu_4)})(1 + e^{b(x_\gamma(t_4) - \tau\gamma_4)})}, \\ M_\gamma(p_6) = 1 - \frac{x_\mu(t_4)C_4}{(1 + e^{-b(x_\mu(t_4) - \tau\mu_4)})(1 + e^{b(x_\gamma(t_4) - \tau\gamma_4)})}. \end{cases}$$

4) 点燃 T_4 中的变迁 t_5 , 有 $x_\mu(t_5) = M_\mu(p_6) \times \omega_{51} + M_\mu(p_7) \times \omega_{52}, x_\gamma(t_5) = M_\gamma(p_6) \times \omega_{51} + M_\gamma(p_7) \times \omega_{52}, \langle \tau_\mu, \tau_\gamma \rangle = \langle \tau\mu_5, \tau\gamma_5 \rangle$, 则

$$\begin{cases} M_\mu(p_8) = \frac{x_\mu(t_5)C_5}{(1 + e^{-b(x_\mu(t_5) - \tau\mu_5)})(1 + e^{b(x_\gamma(t_5) - \tau\gamma_5)})}, \\ M_\gamma(p_8) = 1 - \frac{x_\mu(t_5)C_5}{(1 + e^{-b(x_\mu(t_5) - \tau\mu_5)})(1 + e^{b(x_\gamma(t_5) - \tau\gamma_5)})}. \end{cases}$$

运行算法 2, 对模型逐层点燃便可进行 IFPN 参数的学习和修正.

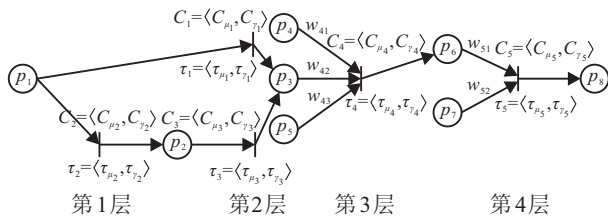


图 2 直觉模糊 Petri 网分层模型

3.2 训练测试

假设 IFPN 模型参数的理想值为 $\omega_{41} = 0.2, \omega_{42} = 0.5, \omega_{43} = 0.3, \omega_{51} = 0.4, \omega_{52} = 0.6, C_1 = \langle 0.7, 0.25 \rangle, C_2 = \langle 0.9, 0.08 \rangle, C_3 = \langle 0.6, 0.36 \rangle, C_4 = \langle 0.8, 0.15 \rangle, C_5 = \langle 0.7, 0.25 \rangle, \tau_1 = \langle 0.3, 0.65 \rangle, \tau_2 = \langle 0.4, 0.56 \rangle, \tau_3 = \langle 0.2, 0.78 \rangle, \tau_4 = \langle 0.5, 0.5 \rangle, \tau_5 = \langle 0.4, 0.56 \rangle$. 根据模型推理算法对产生的 200 个训练样本进行学习训练. 推理函数中常量 $b = 5000$; 初始学习率 $\eta = 0.1$, 其后随网络输出误差根据式 (13) 动态调整; $\varepsilon = 0.06$. 训练前模型原始参数以随机取数的方式初始化. 经过 300 次学习后, 各参数的优化结果如表 1 所示, 各参数优化的均方误差分别为 $MSE(\omega) = 0.3289 \times 10^{-3}, MSE(C) = 2.6973 \times 10^{-3}, MSE(\tau) = 2.6934 \times 10^{-3}$. 样本总平均误差函数值为 $0.058 < 0.06$, 满足要求.

表 1 各参数优化结果

参数	实际输出	参数	实际输出	参数	实际输出
ω_{41}	0.180 2	C_1	$\langle 0.687 5, 0.272 5 \rangle$	τ_1	$\langle 0.342 7, 0.602 4 \rangle$
ω_{42}	0.486 3	C_2	$\langle 0.782 6, 0.147 4 \rangle$	τ_2	$\langle 0.442 5, 0.501 6 \rangle$
ω_{43}	0.333 5	C_3	$\langle 0.518 7, 0.371 3 \rangle$	τ_3	$\langle 0.253 2, 0.732 5 \rangle$
ω_{51}	0.381 4	C_4	$\langle 0.823 4, 0.166 6 \rangle$	τ_4	$\langle 0.531 5, 0.441 6 \rangle$
ω_{52}	0.618 6	C_5	$\langle 0.652 4, 0.287 6 \rangle$	τ_5	$\langle 0.352 6, 0.612 8 \rangle$

3.3 算法性能评估

对于 IFPN 模型而言, 最主要的性能指标是系统本身的自适应能力和泛化功能. 本文取 150 个测试样本, 应用优化的模型进行 IFPN 形式化推理, 将获取终止库所的实际输出与其期望输出进行比较, 结果如图 3 和图 4 所示. 由此可见, 本文提出的优化模型输出

值与期望值基本重合, 即该参数优化学习算法具有很好的泛化性能.

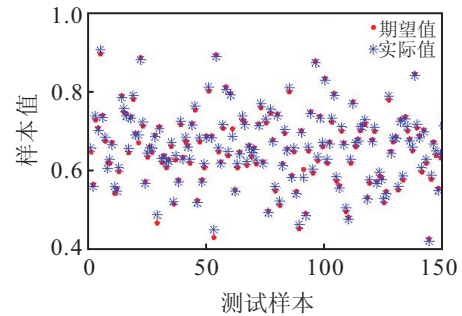


图 3 IFPN 参数期望值与实际值比较

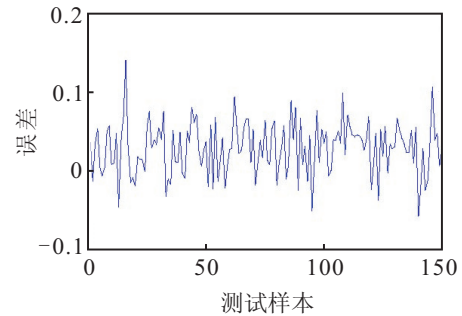


图 4 IFPN 参数测试误差

根据度量标准中的性能评判指标 (APE)^[7] 将本文方法与几种常用的 FPN 模型优化算法进行比较, 结果如表 2 所示. 可见, 本文成功地将 BP 误差反传算法直觉模糊化引入 IFPN 模型, 其参数的均方差均优于 BP-FPN 和 GA-FPN 方法, 虽在可信度和阈值方面稍逊于 ACA-FPN 方法, 但 ACA-FPN 方法在实际应用中经常被赋予很多限制条件, 而且还存在能否合理转化为 ACA 适用范围的问题, 这些都使得它并不能总是收敛到真值, 类似的情况 GA 也存在. 因此, 基于 BP 算法的 IFPN 参数优化方法是比较优越的, 虽然其收敛速度慢, 但本文通过引入动量项和变步长法, 在一定程度上解决了此问题.

表 2 BP-IFPN 方法与其他方法的比较

方法	权重 MSE	可信度 MSE	阈值 MSE	测试集 APE
BP-FPN ^[9]	$0.699 7 \times 10^{-3}$	$5.504 1 \times 10^{-3}$	$2.776 6 \times 10^{-3}$	0.097 0
GA-FPN ^[7]	0.42×10^{-3}	3.38×10^{-3}	4.76×10^{-3}	0.072 2
ACA-FPN ^[8]	$0.339 1 \times 10^{-3}$	$2.361 2 \times 10^{-3}$	$2.655 5 \times 10^{-3}$	0.035 8
BP-IFPN	$0.328 9 \times 10^{-3}$	$2.457 3 \times 10^{-3}$	$2.693 4 \times 10^{-3}$	0.035 2

由于需要学习的参数较多, IFPN 经过学习得到的调整参数收敛于输入参数的局部极值, 与理想的参数值有一定差异 (见表 1), 尤其是阈值参数差异较大. 但是, IFPN 的参数是有特定意义的, 这是 IFPN 模型与神经网络的最大区别, 并且可以通过对学习后参数的合法性进行分析, 不断地调整输入, 从而使参数接

近于理想值. 在实际应用中还可以考虑如何将多种优化方案结合使用, 取长补短, 不断提高系统参数的优化精度.

4 结 论

本文通过对经典的 BP 误差反传算法进行直觉化扩展, 提出了一种 IFPN 模型参数优化方法. 该算法加

入了非隶属度参数, 不仅使得对不确定信息的描述更加精确, 而且由于非隶属度函数的反制作用, 参数的优化结果更加理想; 同时引入了动量项和变步长法, 减小了学习过程中的振荡趋势, 改善了收敛性. 因此, 该优化模型不仅具有高效的并行处理能力, 而且拥有象BP网络一样的自学习功能, 对直觉模糊产生式规则库系统的建立和维护有着重要意义, 为解决信息融合领域中一些复杂系统结构模型的构建和推理问题提供了理论支持. 如何进一步提高IFPN模型参数的并行优化精度将是今后研究的重点.

参考文献(References)

- [1] 申晓勇, 雷英杰, 华继学, 等. 基于IFTPN的不确定时间知识描述和推理方法[J]. 控制与决策, 2010, 25(10): 1457-1462.
(Shen X Y, Lei Y J, Hua J X, et al. Description and reasoning method of uncertain temporal knowledge based on IFTPN[J]. Control and Decision, 2010, 25(10): 1457-1462.)
- [2] 周创明, 申晓勇, 雷英杰. 基于直觉模糊Petri网的故意图识别方法研究[J]. 计算机应用, 2009, 37(9): 2464-2467.
(Zhou C M, Shen X Y, Lei Y J. Research of intention recognition method based on intuitionistic fuzzy Petri net[J]. J of Computer Applications, 2009, 37(9): 2464-2467.)
- [3] 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 P M. Learning capability in fuzzy Petri nets based on BP net[J]. Chinese J of Computers, 2004, 27(5): 695-702.)
- [4] Eric C C Tsang, Daniel S Yeung, John W T Lee. Learning capability in fuzzy petri nets[C]. IEEE Proc of Int Conf on Systems, Man, and Cybernetics. Tokyo, 1999: 355-360.
- [5] Li Xiao-ou, Yu Wen, Felipe Lara-rosano. Dynamic knowledge inference and learning under adaptive fuzzy Petri net framework[J]. IEEE Trans on System, Man, and Cybernetics, Part C: Applications and Reviews, 2000, 30(4): 442-450.
- [6] Li Xiao-ou, Felipe Lara-rosano. Adaptive fuzzy Petri nets for dynamic knowledge representation and inference[J]. Expert Systems with Applications, 2000, 19(3): 235-241.
- [7] Wu S Q, Er M J, Gao Y. A fast approach for automatic generation of fuzzy rules by generalized dynamic fuzzy neural networks[J]. IEEE Trans on Fuzzy Systems, 2001, 9(4): 578-594.
- [8] 李洋, 乐晓波. 基于遗传-BP算法的FPN参数优化的研究[J]. 计算机工程, 2006, 32(24): 189-191.
(Li Y, Yue X B. Research of parameters optimization of fuzzy Petri nets based on genetic-BP algorithm[J]. Computer Engineering, 2006, 32(24): 189-191.)
- [9] 李洋, 乐晓波. 基于模糊Petri网学习能力问题的最优化算法[J]. 计算机应用与软件, 2010, 27(11): 127-130.
(Li Y, Yue X B. An optimization algorithm based on fuzzy Petri net for the problem of learning capability[J]. Computer Applications and Software, 2010, 27(11): 127-130.)
- [10] 汤新民, 钟诗胜. 基于元模型的模糊Petri网反向传播学习算法[J]. 系统仿真学报, 2007, 19(14): 3163-3165.
(Tang X M, Zhong S S. BP learning algorithm for fuzzy Petri nets based on metal model[J]. J of System Simulation, 2007, 19(14): 3163-3165.)
- [11] 宗群, 马宏波, 王中海. 基于NNFPN模型的电梯故障诊断方法的研究[J]. 控制与决策, 2005, 20(3): 341-344.
(Zong Q, Ma H B, Wang Z H. On elevator fault diagnosis method based on NNFPN model[J]. Control and Decision, 2005, 20(3): 341-344.)
- [12] 孟宪刚, 严洪森. 基于自学习模糊Petri网的知识化制造系统采购预测[J]. 控制与决策, 2009, 24(3): 371-376.
(Meng X G, Yan H S. Purchase forecasting in knowledgeable manufacturing systems based on self-learning fuzzy Petri nets[J]. Control and Decision, 2009, 24(3): 371-376.)
- [13] 姚熊亮, 冯麟涵, 张阿漫. 改进FPN在航空保障系统易损性评估中的应用[J]. 电机与控制学报, 2009, 13(3): 464-470.
(Yao X L, Feng L H, Zhang A M. Systems vulnerability assessment of aircraft guarantee system based on improved FPN[J]. Electric Machines and Control, 2009, 13(3): 464-470.)
- [14] 马敏, 黄建国, 夏侯士戟. 基于自适应模糊Petri网的雷达故障诊断方法研究[J]. 仪器仪表学报, 2008, 29(2): 261-265.
(Ma M, Huang J G, Xiahou S J. Radar fault diagnosis with the adaptive fuzzy Petri net[J]. Chinese J of Scientific Instrument, 2008, 29(2): 261-265.)
- [15] Shen Xiao-yong, Lei Ying-jie, Li Cheng-hai. Intuitionistic fuzzy Petri nets model and reasoning algorithm[C]. 2009 6th Int Conf on Fuzzy Systems and Knowledge Discovery. Tianjin, 2009, 3: 119-122.