

基于 UKF 的自组织直觉模糊神经网络

徐小来¹, 雷英杰¹, 谢文彪²

(1. 空军工程大学导弹学院, 陕西西安 710049; 2. 长沙理工大学电气与信息工程学院, 湖南长沙 410076)

摘 要: 模糊集在语义描述上存在不足, 因此, 如何对模糊神经网络进行扩展是当前模糊神经网络研究的热点, 针对这一问题, 本文提出了基于 UKF 的自组织直觉模糊神经网络. 首先, 给出了直觉模糊神经网络的结构和各层的含义; 其次, 推导了直觉模糊神经网络的学习算法, 用 LLS 和 UKF 分别学习线性 and 非线性参数; 然后, 给出了模糊规则生成的准则, 并用误差下降率方法作为规则修剪的策略, 删除作用不大的规则; 最后, 通过典型的函数逼近、系统辨识和时间序列预测实例, 表明本文算法得到的直觉模糊神经网络的结构更为紧凑, 泛化性能也更佳.

关键词: 直觉模糊集合; UKF; 自组织模糊神经网络; 系统辨识; 函数逼近; 时间序列预测

中图分类号: TP183 **文献标识码:** A **文章编号:** 0372-2112 (2010) 03-0638-08

Self-Organising Intuitionistic Fuzzy Neural Networks Based on UKF

XU Xiao-lai¹, LEI Ying-jie¹, XIE Wen-biao²

(1. Missile Institute, Air Force Engineering University, Xi'an, Shaanxi 710049, China;

2. School of Electrical & Information Engineering, Changsha University of Science and Technology, Changsha, Hunan 410078, China)

Abstract: Because fuzzy sets exist deficiency on semantic description, much of the current research interest in neuro-fuzzy hybrid systems is focused on how to extend fuzzy neural networks. To deal with this problem, a self-organizing intuitionistic fuzzy networks based on UKF is presented. Firstly, structure of intuitionistic fuzzy networks and meanings of each layer is proposed. Secondly, training algorithm is deduced, and LLS and UKF are used to learn linear and non-linear parameters respectively. Thirdly, guideline of how to generate a new rule is given, and method of error descending rate is used as fuzzy rule pruning strategy, so that rule which plays an unimportant role in the system is deleted. At last, typical experiments of function approximation, system identification and prediction of time-series indicate that a fuzzy network obtained by the proposed algorithm has a more tighten structure and better generalization than other algorithms.

Key words: intuitionistic fuzzy sets; unscented Kalman filter; self-organising fuzzy neural networks (SOFNN); system identification; function approximation; time-series prediction

1 引言

由于自组织模糊神经网络通过控制模糊规则的自动生成和修剪, 可得到一个规则数更小和泛化能力更佳的网络, 从而成为模糊神经网络研究的热点. Wu 提出了动态模糊神经网络 (Dynamic Fuzzy Neural Networks, D-FNN), 结构辨识与参数估计同时进行, 采用误差下降率作为修剪策略, 得到了较为紧凑的结构, 但是假设一条规则中输入变量的所有高斯隶属函数的宽度是相同的, 与现实不符^[1]; Wu 后来又提出了广义动态模糊神经网络 (Generalized Dynamic Fuzzy Neural Networks, GD-FNN), 克服了 D-FNN 的不足, 学习效率和性能均优于 D-FNN^[2]; Gang 提出了自组织模糊神经网络 (Self-Organising Fuzzy Neural Networks, SOFNN), 其本质与 GD-FNN 是一致的, 只是在参数学习时采用递推最小二乘法 (Re-

cursive Least Square, RLS), 结构较 GD-FNN 更为紧凑^[3]; 文献[1~3]主要存在以下两个问题: (1) 只对线性部分的参数进行了学习, 而对非线性部分的参数只是有规律地扩大, (2) 模糊集在语义描述上存在不足, 要求隶属函数的参数是确定的, 这在实际中是很难满足的. 针对问题(1), 文献[4]在第二阶段通过遗传算法和 BP 算法对非线性参数进行了学习, 算法非常复杂且容易陷入局部最优, 适用性不强; 文献[5]采用 EKF 对 D-FNN 的非线性参数进行了学习, 性能有所提高, 但 EKF 精度不高, 只是在初始误差和噪声不大时, 才表现较好. 针对文献[1~3]存在的问题, Hani 等研究了区间二型模糊神经网络^[6~8], Juang 研究了区间二型模糊神经网络的自进化问题, 但没有采用规则修剪策略, 以致规则生成后就不能剔除, 采用 BP 算法对非线性参数进行学习容易陷入局部最优^[9].

直觉模糊集是对模糊集最有影响的扩展,在语义描述上具有独到的优势^[10];为了改善非线性滤波的效果,Julier 等人提出了 UKF (Unscented Kalman Filter, UKF) 算法,由于不要计算非线性变换的导数而隐含具有二阶滤波器的精度,计算量与 EKF 同阶次^[11],因此,UKF 在非线形滤波领域得到了广泛应用^[12~15]. 本文提出了自组织直觉模糊神经网络(Self-Organising Intuitionistic Fuzzy Neural Networks, SOIFNN)的结构,模糊规则生成和修剪的策略,并用线性最小二乘(Linear Least Square, LLS)和 UKF 分别学习 SOIFNN 线性和非线性参数的,提出了 UKF-SOIFNN 训练算法,得到了一个较 GD-FNN 和 SOFNN 更为紧凑的结构,泛化性能也更佳.

2 UKF-SOIFNN 的结构

定义 1 (直觉模糊集) 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 $\gamma_A(x): X \rightarrow [0, 1]$ 分别代表 A 的隶属函数 $\mu_A(x)$ 和非隶属函数 $\gamma_A(x)$, 且对于 A 上的所有 $x \in X, 0 \leq \mu_A(x) + \gamma_A(x) \leq 1$ 成立.

本文将直觉模糊集合都转换为区间值模糊集合, 即 $A = (\mu_A, \gamma_A) \mapsto \tilde{A} = [\mu_A, 1 - \gamma_A]$.

区间值 Takagi-Sugeno-Kan (TSK) 模糊规则具有以下形式:

Rule i : IF x_1 is \tilde{A}_{1i}, \dots , and x_r is \tilde{A}_{ri} , then y is $\tilde{a}_{0i} + \sum_{j=1}^r \tilde{a}_{ji}$

$x_j, i = 1, \dots, u$

其中 $\tilde{A}_{ji}, j = 1, \dots, u$ 是区间值模糊集, u 是模糊规则数, $\tilde{a}_{ji} = [c_{ji} - s_{ji}, c_{ji} + s_{ji}]$ 是区间值.

UKF-SOIFNN 的结构是一个六层的网络结构, 如图 1 所示.

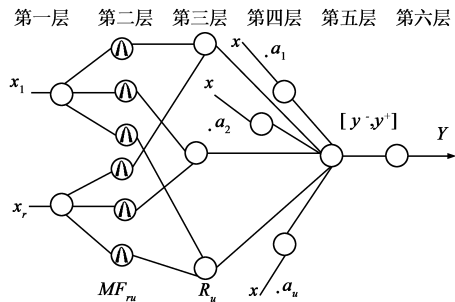


图1 SOIFNN 的结构

第一层:称为为输入层,每个节点分别表示一个输入的语言变量.

第二层:称为隶属函数层,每个节点分别表示一个区间值隶属函数. 区间值隶属函数的上界和下界都采用高斯函数,且具有相同的中心和不同的宽度,如下式

所示:

$$\mu_{ij}^+(x_i) = \exp\left(-\left(\frac{x_i - c_{ij}}{\sigma_{ij}^+}\right)^2\right), i = 1, \dots, r; j = 1, \dots, u \quad (2)$$

$$\mu_{ij}^-(x_i) = \exp\left(-\left(\frac{x_i - c_{ij}}{\sigma_{ij}^-}\right)^2\right), i = 1, \dots, r; j = 1, \dots, u \quad (3)$$

其中, c_{ij} 是高斯隶属函数的中心, σ_{ij}^- 和 σ_{ij}^+ ($\sigma_{ij}^- \leq \sigma_{ij}^+$) 是高斯隶属函数的宽度. 因此, 节点的输出是一区间值模糊集 $[\mu_{ij}^-, \mu_{ij}^+]$, 如图 2 所示, 虚线为 μ_{ij}^- , 实线为 μ_{ij}^+ .

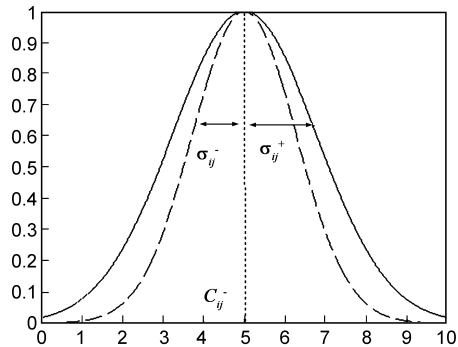


图2 高斯隶属函数

第三层:称为 T -范数层,每个节点分别代表一条模糊规则中的 IF 部分,第 j 个节点 R_j 的输出为一区间值模糊集 $\Phi_j = [\varphi_j^-, \varphi_j^+]$:

$$\varphi_j^- = \prod_{i=1}^r \mu_{ij}^-, \quad \varphi_j^+ = \prod_{i=1}^r \mu_{ij}^+ \quad (4)$$

第四层:称为结论层,每个节点分别代表一条模糊规则中的 then 部分,第 j 个节点的输出为一区间值模糊集 $w_j = [w_j^-, w_j^+]$:

$$[w_j^-, w_j^+] = [c_{j0} - s_{j0}, c_{j0} + s_{j0}] + \sum_{i=1}^r [c_{ji} - s_{ji}, c_{ji} + s_{ji}] \cdot x_i \quad (5)$$

第五层:称为输出处理层,输出为 $y = [y^-, y^+]$, y^-, y^+ 可通过面积中心法进行求解,如下式所示:

$$y^- = \frac{\sum_{j=1}^u \varphi_j^- w_j^-}{\sum_{j=1}^u \varphi_j^-}, \quad y^+ = \frac{\sum_{j=1}^u \varphi_j^+ w_j^+}{\sum_{j=1}^u \varphi_j^+} \quad (6)$$

第六层:称为输出层,每一个节点都与一个语言变量相关联,通过去模糊化算子计算输出.

$$y = \frac{y^- + y^+}{2} \quad (7)$$

3 UKF-SOIFNN 的学习算法

UKF-SOIFNN 的学习算法包括结构辨识和参数学习. 结构辨识试图通过自组织的方法建立一个紧凑的网络,使模糊规则可以动态生成和删除;参数学习采用交替学习的策略,首先,通过 LLS 学习线性参数,然后,

通过 UKF 学习非线性参数,使 UKF-SOIFNN 更快地收敛,获得更好的泛化性能.

3.1 参数的学习

式(6)可以写成

$$\begin{aligned} y^- &= \phi^- w^-, & \phi^- &\in R^{1 \times u} \\ y^+ &= \phi^+ w^+, & \phi^+ &\in R^{1 \times u} \end{aligned} \quad (8)$$

其中 $\phi^- = [\phi_1^-, \dots, \phi_u^-]$, $\phi_j^- = \varphi_j^- / (\sum_{j=1}^u \varphi_j^-)$,

$$\phi^+ = [\phi_1^+, \dots, \phi_u^+], \phi_j^+ = \varphi_j^+ / (\sum_{j=1}^u \varphi_j^+).$$

将式(8)代入式(7)得

$$y = \frac{1}{2}(\phi^- w^- + \phi^+ w^+) = \frac{1}{2}[\phi^- \quad \phi^+] \begin{bmatrix} w^- \\ w^+ \end{bmatrix}$$

$$= \frac{1}{2}[\phi_1^- \dots \phi_u^- \phi_1^+ \dots \phi_u^+] \begin{bmatrix} \sum_{i=0}^r c_{1i} x_i - \sum_{i=0}^r s_{1i} |x_i| \\ \vdots \\ \sum_{i=0}^r c_{ui} x_i - \sum_{i=0}^r s_{ui} |x_i| \\ \sum_{i=0}^r c_{1i} x_i + \sum_{i=0}^r s_{1i} |x_i| \\ \vdots \\ \sum_{i=0}^r c_{ui} x_i + \sum_{i=0}^r s_{ui} |x_i| \end{bmatrix} \quad (9)$$

其中 $x_0 = 1$, 式(9)可进一步写成

$$y = W\phi \quad (10)$$

$$W = [c_{10} \dots c_{1r} \dots c_{u0} \dots c_{ur} \dots s_{10} \dots s_{1r} \dots s_{u0} \dots s_{ur}] \quad (11)$$

$$\phi = [\phi_1^c x_0 \dots \phi_1^c x_r \dots \phi_u^c x_0 \dots \phi_u^c x_r, \phi_1^s |x_0| \dots \phi_1^s |x_r| \dots \phi_u^s |x_0| \dots \phi_u^s |x_r|] \quad (12)$$

其中 $\phi_j^c = (\phi_j^- + \phi_j^+)/2$, $\phi_j^s = (\phi_j^+ - \phi_j^-)/2$, ϕ_j^c 取值比较大,对输出起主要作用,而 ϕ_j^s 是区间模糊 $\Phi^j = [\phi_j^-, \phi_j^+]$ 的犹豫度的一半,取值比较小.

假定 n 个观测数据产生了 u 条模糊规则,理想的输出为 $T = [t_1, t_2, \dots, t_n]$, 则 N 节点的输出可以由式(10)得到,写成矩阵形式:

$$Y = W\Psi \quad (13)$$

$$Y = [y_1, y_2, \dots, y_n] \quad (14)$$

$$\Psi = \begin{bmatrix} \phi_{11}^c x_{01} & \dots & \phi_{1n}^c x_{0n} \\ \vdots & \vdots & \vdots \\ \phi_{11}^c x_{r1} & \dots & \phi_{1n}^c x_{rm} \\ \vdots & \vdots & \vdots \\ \phi_{u1}^c x_{01} & \dots & \phi_{un}^c x_{0n} \\ \vdots & \vdots & \vdots \\ \phi_{u1}^c x_{r1} & \dots & \phi_{un}^c x_{rm} \\ \vdots & \vdots & \vdots \\ \phi_{11}^s |x_{01}| & \dots & \phi_{1n}^s |x_{0n}| \\ \vdots & \vdots & \vdots \\ \phi_{11}^s |x_{r1}| & \dots & \phi_{1n}^s |x_{rm}| \\ \vdots & \vdots & \vdots \\ \phi_{u1}^s |x_{01}| & \dots & \phi_{un}^s |x_{0n}| \\ \vdots & \vdots & \vdots \\ \phi_{u1}^s |x_{r1}| & \dots & \phi_{un}^s |x_{rm}| \end{bmatrix} \quad (15)$$

W 可以通过 LLS 进行估计:

$$W^* = T(\Psi^T \Psi)^{-1} \Psi^T \quad (16)$$

此外, W 也可以通过 Kalman 滤波进行估计:

$$\begin{aligned} W(k+1) &= W(k+1) + S(k+1)\phi(k+1) \\ &\quad (t(k+1) - \phi(k+1)W(k)) \end{aligned} \quad (17)$$

$$S(k+1) = \frac{1}{\lambda} [S(k) - \frac{S(k)\phi(k+1)\phi^T(k)S(k)}{\lambda + \phi^T(k+1)S(k)\phi(k+1)}] \quad (18)$$

初始条件为 $W=0$ 和 $S(0) = \chi I$. $S(k)$ 是第 k 个观测数据的误差协方差矩阵, $0 \leq \lambda \leq 1$ 是遗忘因子, λ 越小,表明数据遗忘越快.

在 W 和模糊规则的中心参数 $C = [c_{11}, \dots, c_{1u}, \dots, c_{r1}, \dots, c_{ru}]$ 为已知的前提下, 式(10)可以写成非线性动力系统的形式:

$$\begin{aligned} z_k &= z_{k-1} \\ t_k &= h_k(z_k) + v_k \end{aligned} \quad (19)$$

其中, h_k 为非线性函数, v_k 为随机量测噪声, $z_k = [\sigma_{11}^-, \dots, \sigma_{1u}^-, \dots, \sigma_{r1}^-, \dots, \sigma_{ru}^-, \sigma_{11}^+, \dots, \sigma_{1u}^+, \dots, \sigma_{r1}^+, \dots, \sigma_{ru}^+]_k$ 为系统状态.

假设量测噪声为互不相关零均值白噪声, 协方差矩阵为 R_k , 初始状态 z_0 与噪声独立, 先验均值和协方差阵是 $E(z_0) = \hat{z}_{0|0}$, $\text{cov}(z_0) = P_{0|0} = \chi I$. 因此, 可用 UKF 对式(19)进行滤波, 步骤如下:

(1) 计算 σ 点 $\xi_k^{(0)}, i=0, 1, \dots, 2n$ 即

$$\begin{cases} \xi_{k-1|k-1}^{(0)} = \hat{z}_{k-1|k-1} \\ \xi_{k-1|k-1}^{(i)} = \hat{z}_{k-1|k-1} + (\sqrt{(n+\lambda)P_{k-1|k-1}})_i, & i=1, 2, \dots, n \\ \xi_{k-1|k-1}^{(i)} = \hat{z}_{k-1|k-1} - (\sqrt{(n+\lambda)P_{k-1|k-1}})_{i-n}, & i=n+1, \dots, 2n \end{cases} \quad (20)$$

其中 $\lambda = \alpha^2(n + \kappa) - n$, α 决定 σ 点的散布程度, $(\sqrt{(n+\lambda)P_{k-1|k-1}})_i$ 表示矩阵平方根第 i 列, 本文中 $\alpha = 0.5, \kappa = 3 - n$.

(2) 计算 σ 点 $\xi_k^{(i)}$, 通过状态演化方程的传播, 即

$$\begin{cases} \omega_0^{(m)} = \lambda / (n + \lambda) \\ \omega_0^{(c)} = \lambda / (n + \lambda) + (1 - \alpha^2 + \beta) \\ \omega_i^{(m)} = \omega_i^{(c)} = 0.5 / (n + \lambda), & i=1, \dots, 2n \end{cases} \quad (21)$$

$$\begin{cases} \xi_k^{(i)} = \xi_{k-1|k-1}^{(i)}, & i=0, 1, \dots, 2n \\ \hat{z}_{k|k-1} = \sum_{i=0}^{2n} \omega_i^{(m)} \xi_k^{(i)} \\ P_{k|k-1} = \sum_{i=0}^{2n} \omega_i^{(c)} (\xi_k^{(i)} - \hat{z}_{k|k-1})(\xi_k^{(i)} - \hat{z}_{k|k-1})^T + Q_{k-1} \end{cases} \quad (22)$$

其中 β 用来描述 x 的分布信息,本文中 $\beta = 2$, $\omega_i^{(m)}$ 和 $\omega_i^{(c)}$ 分别为求一阶和二阶统计特性时的权系数.

(3) 计算 σ 点 $\hat{z}_{k|k-1}$ 、 $P_{k|k-1}$ 通过量测方程对 z_k 的传播,即

$$\begin{cases} \xi_k^{(0)} = \hat{z}_{k|k-1} \\ \xi_k^{(i)} = \hat{z}_{k|k-1} + (\sqrt{(n+\lambda)P_{k|k-1}})_i, i = 1, 2, \dots, n \\ \xi_k^{(i)} = \hat{z}_{k|k-1} - (\sqrt{(n+\lambda)P_{k|k-1}})_{i-n}, i = n+1, \dots, 2n \end{cases} \quad (23)$$

(4) 计算输出的一步提前预测,即

$$\begin{cases} \zeta_k^{(i)} = h_k(\xi_k^{(i)}), i = 0, 1, \dots, 2n \\ \hat{t}_{k|k-1} = \sum_{i=0}^{2n} \omega_i^{(m)} \zeta_k^{(i)} \\ P_{\hat{t}_k} = \sum_{i=0}^{2n} \omega_i^{(c)} (\zeta_k^{(i)} - \hat{t}_{k|k-1}) (\zeta_k^{(i)} - \hat{t}_{k|k-1})^T + R_k \\ P_{z_{\hat{t}_k}} = \sum_{i=0}^{2n} \omega_i^{(c)} (\xi_k^{(i)} - \hat{z}_{k|k-1}) (\zeta_k^{(i)} - \hat{t}_{k|k-1})^T \end{cases} \quad (24)$$

(5) 在获得新的量测 y_k 后,进行滤波更新.

$$\begin{cases} \hat{z}_{k|k} = \hat{z}_{k|k-1} + K_k(t_k - t_{k|k-1}) \\ K_k = P_{\hat{z}_k} P_{t_k}^{-1} \\ P_{k|k} = P_{k|k-1} - K_k P_{t_k}^{-1} K_k^T \end{cases} \quad (25)$$

式中 K_k 是滤波增益阵.

UKF 通过选取一个最小样本点集合来近似高斯随机变量的状态分布,这些样本点能够完全捕获高斯随机变量的真实均值和方差,UKF 对于任何非线性系统精度都能达到二阶以上.

3.2 增加一条模糊规则

对于第 k 个观测数据 $(X(k), t(k))$, $k = 1, 2, \dots, n$, 其中, k 是全部训练数据的数量, $X(k)$ 是第 k 个输入样本向量, $t(k)$ 是第 k 个期望的输出,用式(10)计算当前结构下的 SOIFNN 输出 $y(k)$. 定义系统误差为:

$$e(k) = (t(k) - y(k)) \quad (26)$$

由式(4)可以计算每条规则激活程度的区间模糊值,取他们的平均值作为规则的激活度.

$$\varphi_j^c = \frac{\varphi_j^- + \varphi_j^+}{2} \quad (27)$$

对于每一个新到的数据 $X(k) = [x_{1k}, \dots, x_{rk}]$, 由式(27)计算规则的激活度,找到

$$J = \arg \max_{1 \leq j \leq u} \varphi_j^c(X(k)) \quad (28)$$

如果 $e(k) > k_e$ 且 $\varphi_j^c > k_\phi$, 则增加一条模糊规则, k_e 和 k_ϕ 是预先定义的阈值,此时,参数的学习须从第一个观测样本开始. 新的模糊规则的初始参数按照如下方式分配:

$$\text{Dist}_i = \begin{bmatrix} \text{dist}_{1k}(h_{m1}) \\ \vdots \\ \text{dist}_{rk}(h_{mr}) \end{bmatrix} = \begin{bmatrix} \min_{j=1 \dots i-1} (|x_{1k} - c_{1j}|) \\ \vdots \\ \min_{j=1 \dots i-1} (|x_{rk} - c_{rj}|) \end{bmatrix} \quad (29)$$

其中 $\text{dist}_{qk}(h_{mq})$ 是第 k 个观测数据的输入 $X = [x_{1q}, \dots, x_{rq}]$ 的第 q 个输入与第 m_q 个神经元的第 h_{mq} 个隶属函数的中心之间的距离.

如果 $\text{dist}_{qk}(h_{mq}) \leq k_w(q)$, $k_w(q)$ 是预定常数,则有

$$\begin{cases} C_{qi} = C_{qh_{mq}} \\ \sigma_{qi}^- = \sigma_{qh_{mq}} \\ \sigma_{qi}^+ = 1.1 * \sigma_{qi}^- \end{cases} \quad (30)$$

如果 $\text{dist}_{qk}(h_{mq}) > k_w(q)$, 则有

$$\begin{cases} C_{qi} = X(k) \\ \sigma_{qi}^- = k * \text{dist}_{qk}(h_{mq}) \\ \sigma_{qi}^+ = 1.1 * \sigma_{qi}^- \end{cases} \quad (31)$$

只有 $e_i > k_e$ 且 $\varphi_j^c > k_\phi$, 才增加一条新的模糊规则, 其它情况下, 只需用 LLS 和 UKF 更新参数即可.

引入“分级学习”的思想^[1], 即 k_e 是一个线性函数, 如式(32)所示:

$$k_e = \max[e_{\max} * \beta^i, e_{\min}] \quad (32)$$

其中 e_{\max} 是预先定义好的最大误差, e_{\min} 是期望的精度, $\beta = e_{\min}/e_{\max}$ 是衰减常数.

3.3 模糊规则的修剪

在式(13) ~ (15) 中, s_{ij} 和 ϕ_j^s 取值都比较小, 对输出影响不大. 为简便起见, 在模糊规则修剪时, 只考虑 c_{ij} 和 ϕ_j^s , 则有

$$Y = W_1 \Psi_1 \quad (33)$$

$$\Psi_1 = \begin{bmatrix} \phi_{11}^c x_{01} & \cdots & \phi_{1n}^c x_{0n} \\ \vdots & \vdots & \vdots \\ \phi_{11}^c x_{r1} & \cdots & \phi_{1n}^c x_{rm} \\ \vdots & \vdots & \vdots \\ \phi_{u1}^c x_{01} & \cdots & \phi_{ur}^c x_{0n} \\ \vdots & \vdots & \vdots \\ \phi_{u1}^c x_{r1} & \cdots & \phi_{ur}^c x_{rm} \end{bmatrix} \quad (34)$$

$$W_1 = [c_{10} \cdots c_{1r} \cdots c_{u0} \cdots c_{ur}]^T \quad (35)$$

如果在学习进行时, 检测到不活跃的模糊规则并加以剔除, 则可获得更为紧凑的结构. 本文采用误差下降率(Err)^[1,2]方法作为修剪策略. 给定 n 对输入-输出 $\{X(i), t(i), i = 1, 2, \dots, n\}$, 考虑式(36)作为回归模型的一种特殊情况:

$$D = H\theta + E \quad (36)$$

其中 $D = T^T \in R^n$ 为期望输出, $H = \Psi_1^T = (h_1 \cdots h_n) \in R^{n \times v}$, $v = u \times (r+1)$, 称为回归量, $\theta = W_1^T \in R^v$ 包含实参数并且假定 $E \in R^n$ 是与回归量 h_i 不相关的一个误差向量.

对任意矩阵 H , 如果它的行数大于列数, 通过 QR 分解:

$$H = QA \quad (37)$$

其中 $Q = (q_1, q_2, \dots, q_v) \in R^{n \times v}$ 为正交矩阵, $A \in R^{v \times v}$ 为上三角矩阵. 定义误差下降率:

$$\text{err}_i = \frac{(q_i^T D)^2}{q_i^T q_i D^T D}, \quad i = 1, 2, \dots, v \quad (38)$$

把新得到的 $\text{err}_i (i = 1, 2, \dots, (r+1)u)$ 重新排列为矩阵 $\Delta = (\delta_1, \delta_2, \dots, \delta_u) \in R^{(r+1) \times u}$, Δ 的第 i 列 δ_i 就是与第 i 条规则相关的 $(r+1)$ 个误差下降率.

$$\eta_i = \sqrt{\frac{\delta_i^T \delta_i}{r+1}} \quad (39)$$

η_i 反映第 i 条规则的重要性, η_i 值越大, 表示第 i 条规则越重要. 如果 $\eta_i < k_{\text{err}}$, 其中 k_{err} 为预设的阈值, 则第 i 条规则可剔除.

本文算法涉及的参数众多, 并且参数对网格性能影响很大. 给出一般的参数设置原则: k_e 决定了网络的精度, 一般取值比期望的误差稍大; k_ϕ 是决定是否增加一条规则的重要依据, k_ϕ 取值大, 则生成的规则就多,

一般取值在 $0.5 \sim 0.75$; k_{err} 是确定是否删除一条规则的依据, k_{err} 取值大, 则删除的规则就多; R_k 一般取 0.01 , χ 一般取值在 $0.05 \sim 0.5$.

综上所述, UKF-SOIFNN 算法的流程图如图 3 所示.

4 实例分析

我们将通过三个典型的实例来验证 UKF-SOIFNN 算法的有效性. 并与前期的工作, 如 D-FNN、GD-FNN、SOFNN、基于 RBF 适应性模糊系统(RBF-AFS)等做了一些比较.

4.1 函数逼近

例 1 三输入非线性函数如下:

$$t = (1 + x^{0.5} + y^{-1} + z^{-1.5})^2, x, y, z \in [1, 6]$$

在定义域中随机选取 216 个训练样本和 125 个测试样本. $e_{\min} = e_{\max} = 0.2$, $k_\phi = 0.6$, $k = 4$, $k_w = [0.6 \ 0.6 \ 0.6]$, $k_{\text{err}} = 0.004$, $\chi = 0.1$, $R_k = 0.01$.

仿真结果如图 4 所示, 从图 4(a)可以看出, 模糊规则是一条一条地生成的, 当第 60 个样本进入的时候, 显然删除了一条规则(实际是删除了两条规则); 从图 4(b)可以看出, RMSE 陡峭的地方, 表明该处生成了或删除了一条规则, 但有时生成一条规则与剔除另一条规则同时进行, 所以在图上看不出规则的生成和删除; 图 4(c)为训练集的期望值和 UKF-SOIFNN 输出的实际值的比较, 可以看出, 期望值与实际值基本重合在一起; 图 4(d)为非线性函数与 UKF-SOIFNN 输出的实际值之间的误差.

从表 1 可以看出, UKF-SOIFNN 生成的规则数最小, 训练和测试的 APE 最小, 因此, UKF-SOIFNN 生成的网络具有结构最紧凑和泛化性能最佳的特点, 性能显著提高. 究其原因, 高斯隶属函数的宽度对网络的结构和泛化能力影响巨大, 而 GD-FNN 和 SOFNN 对高斯隶属函数的宽度处理很简单, 在模糊规则生成的时候, 赋初值, 然后当新样本到来时, 按照某种准则决定是否成比例地扩大, 这种方式对初值依赖, 按比例扩大没有寻优, 从而使得结构存在冗余. 但是 UKF-SOIFNN 增加了一个用 UKF 学习高斯隶属函数宽度的环节, 减小了对初值的依赖, 并且精度能够达到二阶以上; 此外, UKF-SOIFNN 将高斯隶属函数的宽度定义为一个区间值, 在不确定性表达和处理上更为灵活. 因此, UKF-SOIFNN 训练的网络结构更为紧凑, 泛化性能更佳.

表 1 UKF-SOIFNN 与其它方法的比较

模型	规则数	参数数目	训练的 APE	测试的 APE
GD-FNN ^[2]	10	64	2.11	1.54
SOFNN ^[3]	9	60	1.1380	1.1244
UKF-SOIFNN	5	50	0.7420	0.9051

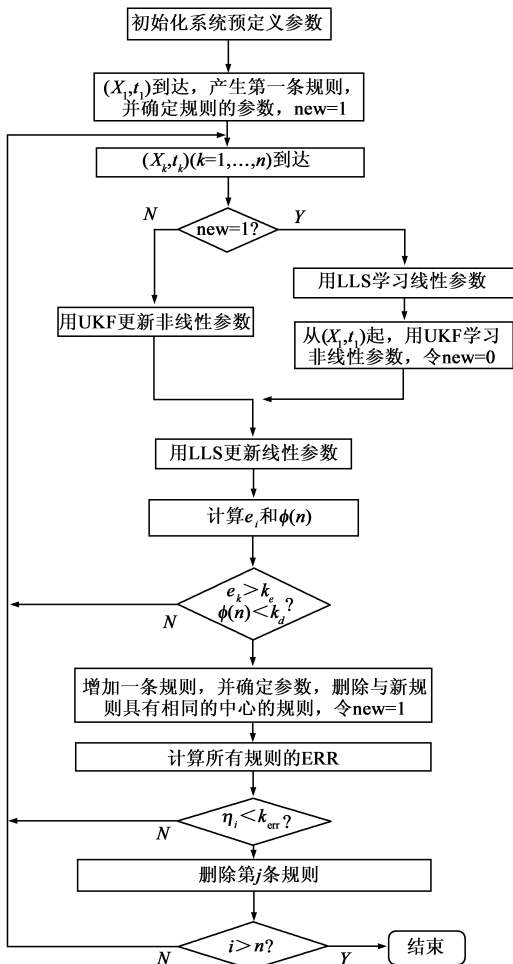


图3 UKF-SOIFNN算法的流程图

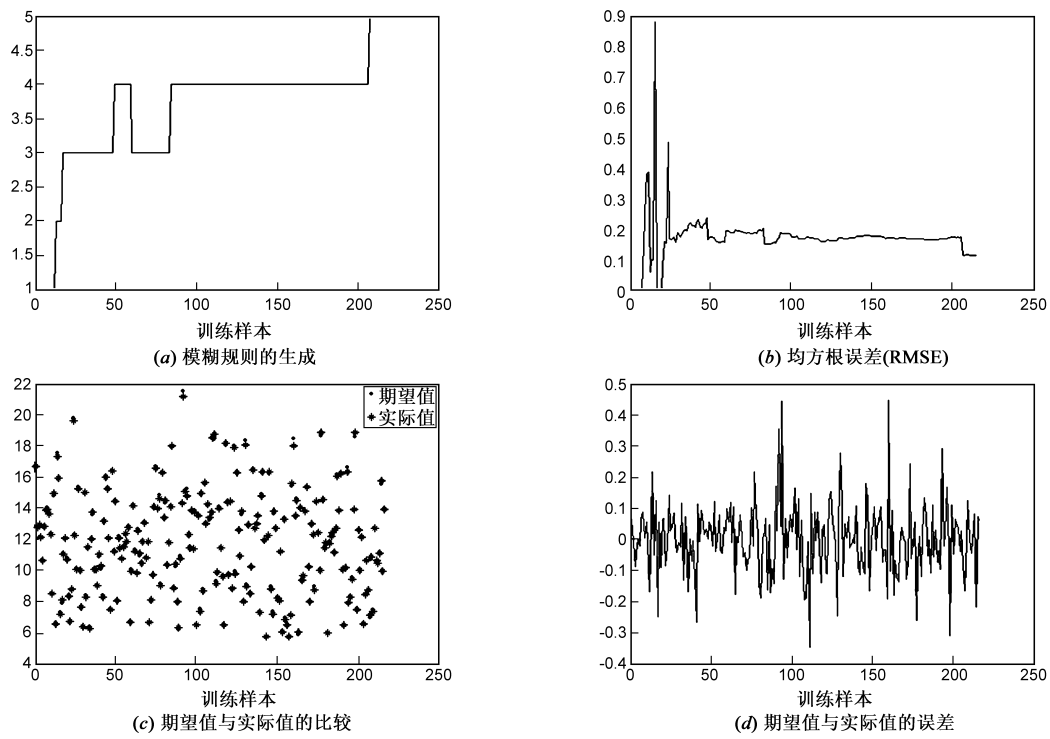


图4 UKF-SOIFNN用于非线性函数逼近

4.2 非线性动态系统的辨识

例 2 被辨识的对象描述如下:

$$y(t+1) = \frac{y(t)y(t-1)[y(t)+2.5]}{1+y^2(t)+y^2(t-1)} + u(t)$$

为了辨识该对象,串-关行模型用下式表示:

$$\hat{y}(t+1) = f(y(t), y(t-1), u(t))$$

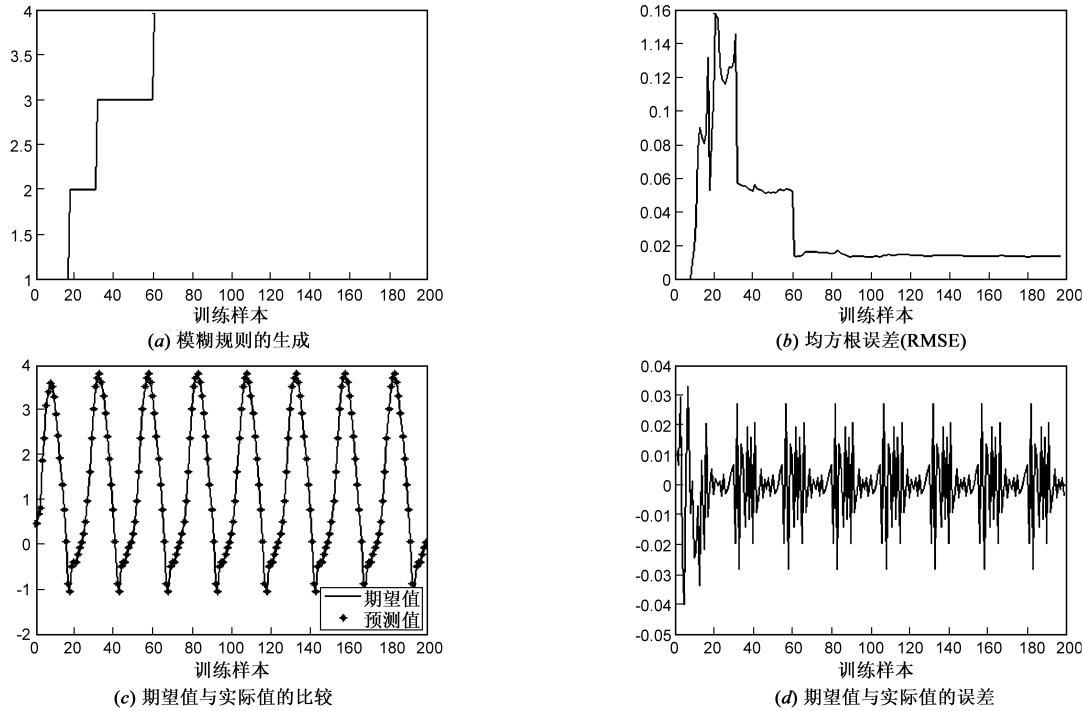


图5 UKF-SIOFNN用于系统辨识

其中, f 是由三输入和单输出的 UKF-SOFNN 实现的函数, $u(t) = \sin(2\pi t/25)$.

在定义域中选取 200 个训练样本,参数设置为: $e_{\max} = 0.3, e_{\min} = 0.02, k_{\phi} = 0.7, k = 1.5, \chi = 0.1, k_w = [0.5 \ 0.5], k_{\text{err}} = 0.003, R_k = 0.01$. 仿真结果如图 5 所示.

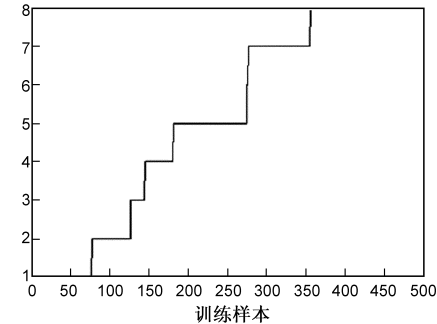
表 2 UKF-SOIFNN 与其它方法的比较

模型	规则数	训练的 RMSE	测试的 RMSE
D-FNN ^[1]	6	0.0283	*
GD-FNN ^[2]	6	0.0241	*
SOFNN ^[3]	5	0.0157	0.0151
SOFNNGA ^[4]	4	0.0159	0.0146
UKF-SOIFNN	4	0.0138	0.0130

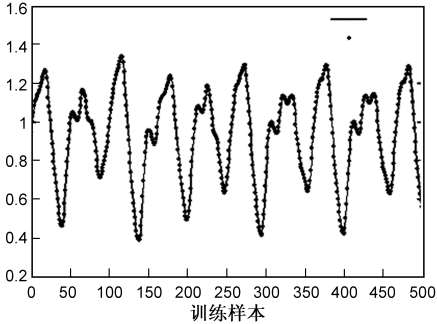
从表 2 可以看出,UKF-SOIFNN 生成的规则数最小,训练的和测度的均方根误差(RMSE)最小,因此,UKF-SOIFNN 生成的网络具有结构最紧凑和泛化性能最佳的特点,性能显著提高.表中的“*”表示原始文献没有提供相关数据.

4.3 Mackey-Glass 时间序列预测

例 3 Mackey-Glass 混沌时间序列是 Mackey 等人



(a) 模糊规则的生成



(c) 期望值与实际值的比较

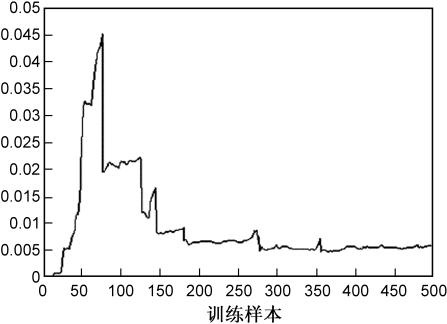
在 1977 年所提出的一个模型,其差分方程如下式所示:

$$x(t+1)=(1-a)x(t)+\frac{bx(t-\tau)}{1+x^{10}(t-\tau)}$$

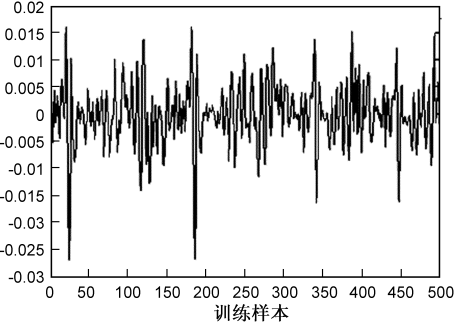
对这种时间序列的预测是一个被许多研究者研究过的典型问题,为了能够在相同的基础之上进行比较,参数选择为: $a=0.1$ 、 $b=0.2$ 和 $\tau=17$,当 $t<0$ 时, $x(t)=0$, $x(0)=1.2$.预测模型为:

$$x(t)=f(x(t-24),x(t-18),x(t-12),x(t-6))$$

用差分方程在 $118\leq t\leq 1117$ 之间选择 1000 个样本,其中,前 500 个为训练样本,后 500 个为测试样本. $e_{\max}=0.3$, $e_{\min}=0.02$, $k_{\varphi}=0.7$, $k=1.5$, $\chi=0.1$, $k_w=[0.5\ 0.5\ 0.5]$, $k_{\text{err}}=0.000025$,仿真结果如图 6 所示.



(b) 均方根误差(RMSE)



(d) 期望值与实际值的误差

图6 UKF-SOIFNN用于Mackey-Glass时间序列预测

表 3 UKF-SOIFNN 与其它方法的比较

方法	规则数	训练的 RMSE	测试的 RMSE
ANFIS ^[17]	16	0.0016	0.0015
RBF-AFS ^[18]	21	0.0107	0.0128
D-FNN ^[1]	10	0.0082	0.0083
UKF-SOIFNN	8	0.0057	0.0061

从表 3 可以看出,UKF-SOIFNN 具有最小的规则数,而 ANFIS 具有最小的训练和测试的 RMSE.此外,UKF-SOIFNN 的 RMSE 比 D-FNN 和 RBF-AFS 都要小很多,说明 UKF-SOIFNN 网络的结构紧凑,泛化能力稍差.原因是 ANFIS 通过迭代学习的方式训练,从而可以达到整体的最优解.而 UKF-SOIFNN 和 D-FNN 等只能获得次优的结果.

5 结论

本文提出了 UKF-SOIFNN 的结构,并推导了训练算法,结构辨识和参数学习同时进行,采用误差下降率作为模糊规则的修剪策略.UKF-SOIFNN 增加了一个用 UKF 学习高斯隶属函数宽度的环节,减小了对初值的依赖,并且精度能够达到二阶以上;此外,UKF-SOIFNN 将高斯隶属函数的宽度定义为一个区间值,在不确定性表达和处理上更为灵活.因此,UKF-SOIFNN 训练的网络结构更为紧凑,泛化性能更佳.通过函数逼近、系统辨识和时间序列预测实例,表明本文算法训练的神经网络性能提高显著.

参考文献:

- [1] Wu S Q, Er M J. Dynamic fuzzy neural networks: a novel approach to function approximation[J]. IEEE Trans Sys, Man, Cybern Part B, 2000, 30(2): 358 – 364.
- [2] 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 Transactions on Fuzzy Systems, 2001, 9(4): 578 – 594.
- [3] Gang Leng, Girijesh Prasad, Thomas Martin McGinnity. An approach for on-line extraction of fuzzy rules using a self-organising fuzzy neural networks[J]. Fuzzy Sets and Systems, 2005, (150): 211 – 243.
- [4] Gang Leng, Thomas Martin McGinnity, Girijesh Prasad. Design for self-organizing fuzzy neural networks based on genetic algorithms[J]. IEEE Transactions on Fuzzy Systems, 2006, 14(6): 755 – 766.
- [5] Er M J, Wong W M, Wu S Q. A comparative study of different methods for realizing D-FNN algorithm[A]. Proc of the 38th IEEE Conference on Decision and Control[C]. Phoenix, Arizona, USA, 1999.
- [6] Hani A. Hagrass. A hierarchical type-2 fuzzy logic control architecture for autonomous mobile robots[J]. IEEE Transactions on Fuzzy Systems, 2004, 12(4): 524 – 539.
- [7] Jerry M. Mendel. Computing derivatives in interval type-2 fuzzy logic systems[J]. IEEE Transactions on Fuzzy Systems, 2004, 12(1): 84 – 98.
- [8] Haining Shu, Qilian Liang, Jean Gao. Wireless sensor network lifetime analysis using interval type-2 fuzzy logic systems [J]. IEEE Transactions on Fuzzy Systems, 2008, 16(2): 416 – 427.
- [9] Chia-Feng Juang, Yu-Wei Tsao. A self-evolving interval type-2 fuzzy neural network with on-line structure and parameter learning [J]. IEEE Transactions on Fuzzy Systems. 2008, 16(6): 1411 – 1424.
- [10] 雷英杰, 路艳丽, 李兆渊. 直觉模糊神经网络的全局逼近能力[J]. 控制与决策, 2007, 22(5): 597 – 600.
LEI Ying-jie, LU Yan-li, LI Zhao-yuan. Function approximation capabilities of intuitionistic fuzzy reasoning neural networks[J]. Control and Decision, 2007, 22(5): 597 – 600. (in Chinese)
- [11] Simon J Junier, Effrey K Uhlmann. Unscented filtering and nonlinear estimation[J]. Proceedings of the IEEE, 2004, 92(3): 401 – 422.
- [12] Dan Simon. A comparison of filtering approaches for aircraft engine health estimation [J]. Aerospace Science and Technology, 2008 (12): 276 – 284.
- [13] Xiong K, Chan C W, Zhang H Y. Detection of satellite attitude sensor faults using the UKF[J]. IEEE Transactions on Aerospace and Electronic Systems, 2007, 43(2): 480 – 482.
- [14] Ronghui Zhan, Jianwei Wan. Neural network-aided adaptive unscented Kalman filter for nonlinear state estimation [J]. IEEE Signal Processing Letters, 2006, 13(7): 445 – 448.
- [15] 张海涛, 李大字, 靳其兵. 基于无先导卡尔曼滤波的RBFN训练算法研究[J]. 北京化工大学学报, 2007, 34(2): 221 – 224
ZHANG Hai-tao, LI Da-zi, JIN Qi-bing. Training radial basis neural networks with the unscented Kalman filter[J]. Journal of Beijing University of Chemical Technology, 2007, 34(2): 221 – 224. (in Chinese)
- [16] Jang J-S R. ANFIS: Adaptive-network-based fuzzy inference system [J]. IEEE Trans Syst Man Cybern, 1993, 23: 665 – 684.
- [17] Cho K B, Wang B H. Radial basis function based adaptive fuzzy systems and their application to system identification and prediction [J]. Fuzzy Sets and System, 1996, (26): 344 – 354.
- [18] 伍世度, 徐军. 动态模糊神经网络-设计与应用[M]. 北京: 清华大学出版社. 2008. 218 – 224.

作者简介:



徐小来 男, 1980 年 12 月出生于湖南省长沙市, 空军工程大学信号与信息处理专业博士生, 研究方向: 智能信息处理与信息融合.

E-mail: xxl1024@163.com



雷英杰 男, 1956 年出生于陕西华阴市, 现为空军工程大学导弹学院教授、博士生导师, 主要研究方向为智能信息处理与智能决策等. E-mail: leiyingjie@sina.com



谢文彪 男, 1980 年出生于湖南省长沙市, 现为长沙理工大学电气与信息工程学院讲师.