

一种基于极点配置稳定的新型局部递归神经网络

孙健¹ 柴毅¹ 李华锋¹ 朱智勤¹

摘要 针对局部全局前馈递归动态神经网络的稳定性问题提出了一种新的采用极点配置稳定方法的局部递归全局前馈 (Locally recurrent global forward, LRGF) 神经网络. 由于动态神经元的极点有存在于实轴上和一对共轭复数极点两种情况, 为了避免神经元无限脉冲响应滤波器 (Infinite impulse response filter, IIR) 的系数投影到稳定区域的复杂性, 构造的神经网络将动态神经元分成实数极点 IIR 和共轭复数极点 IIR 两部分, 通过函数权值的方法将这两部分加权输出. 同时针对这种新的神经网络采用了梯度下降的学习算法. 通过仿真对本文提出的神经网络的可靠性和有效性进行验证, 并分析这种新的神经网络在稳定投影计算上的复杂度.

关键词 动态神经网络, 局部递归全局前馈神经网络, 极点配置, 稳定性投影

DOI 10.3724/SP.J.1004.2012.00183

A Novel Stable Locally Recurrent Neural Network with Pole Assignment Projection Approach

SUN Jian¹ CHAI Yi¹ LI Hua-Feng¹ ZHU Zhi-Qin¹

Abstract This paper derives a new stable locally recurrent global forward (LRGF) neural network with pole assignment projection approach. The pole in the hidden neurons of the LRGF neural network can be classified into two situations. One case is that the pole is on the real axis, and the other case is that the pole is a conjugate complex. We divide the dynamic hidden neuron into two parts according to the kind of the pole, so that it can avoid the complexity of the projective computation. A weight function is used to fuse the two parts. The learning method is based on the gradient decent approach, which has been modified to be fit for the proposed neural network. At last, the simulation is given to demonstrate the reliability and effectiveness of the new neural network, and the complexity of the projection computation is also be analyzed.

Key words Dynamic neural network, locally recurrent global forward (LRGF) neural network, pole assignment, stable projection

动态神经网络, 由于在网络中引入反馈, 相对于传统神经网络, 能更好地解决一些实际问题. 如非线性系统的辨识^[1-3]、信号的预测和识别^[4-6]、系统的控制^[7-9] 以及动态系统的故障诊断^[10-12] 等.

局部递归全局前馈 (Locally recurrent global forward, LRGF) 神经网络作为动态神经网络的一个分支, 通常可分为三种: 1) 将局部递归引入到神经元输入上; 2) 在神经元内引入递归; 3) 在神经元和其激发函数输出间引入递归^[13]. 在这三种神经网络的神经元引入递归, 使其具有有限脉冲响应滤

波器 (Finite impulse response filter, FIR) 或无限脉冲响应滤波器 (Infinite impulse response filter, IIR) 的形式, 通过对 LRGF 神经网络训练, 将多个 FIR 或 IIR 组合, 便可以建立动态过程模型.

在神经网络的训练中, 网络的稳定性十分重要. 动态神经网络在训练时, 很容易因为网络的动态特性不稳定导致训练算法的不稳定和数值计算病态问题, 从而使神经网络的训练算法崩溃. 在讨论动态神经网络的稳定性时, 通常可采用李亚普诺夫方法^[14-15]、状态矩阵估计方法^[16]、Ostrowski 理论^[17] 等. 李亚普诺夫方法有通用性较强的优点, 能分析各种神经网络的稳定性, 但寻找李亚普诺夫函数的过程复杂, 假设条件多. 而状态矩阵估计方法能定性的分析动态神经网络的全局稳定性和局部稳定性以及定量的分析其动态特性, 然而由于在估计的过程中使用了大量不等式, 限制了神经网络参数在稳定区域中的调节范围, 因此较难应用. 在文献 [17] 中, 采用 Ostrowski 理论的方法能有效判断神经网络的稳定性, 不过此方法中参数的设定对应用 Ostrowski 理论的效果有较大影响, 而且推导过程复杂, 选择合

收稿日期 2011-08-31 录用日期 2011-10-19
Manuscript received August 31, 2011; accepted October 19, 2011

国家自然科学基金 (60974090), 国家教育部博士点基金 (102063720090013), 中央高校基本科研业务费 (GDJXS10170010) 资助
Supported by National Natural Science Foundation of China (60974090), Doctoral Fund of Ministry of Education of China (102063720090013), and Fundamental Research Funds for the Central Universities (GDJXS10170010)

本文责任编辑 王聪
Recommended by Associate Editor WANG Cong

1. 重庆大学自动化学院 重庆 400044
1. College of Automation, Chongqing University, Chongqing 400044

适的参数较难. 此外, 文献 [18] 通过在训练中引入稳定投影方法使神经网络稳定. 该方法具有较强的一般性, 但参数的稳定性投影计算较为复杂, 限制条件较多. 针对该问题, 本文提出了一种新的局部递归动态神经网络模型, 该模型采用了新设计的极点配置投影 (Pole assignment projection, PAP) 技术以及改造后的神经元结构来解决局部递归动态神经网络的稳定性问题. 相对于文献 [18] 提出的最短距离投影 (Minimal distance projection, MDP) 方法, 本文提出的 PAP 方法无需求解不等式方程组, 因此在神经网络训练时, 参数到稳定域的投影更加简单, 学习收敛速度快, 更有利于应用到现实问题中.

1 LRGF 神经网络

为介绍 LRGF 神经网络原理, 以含有一个隐层的多输入多输出 LRGF 神经网络为例来说明. 假设神经网络拥有 n 个输入, v 个隐层神经元, m 个输出, 并且在隐层中引入局部反馈. 则神经网络的示意图如图 1 所示.

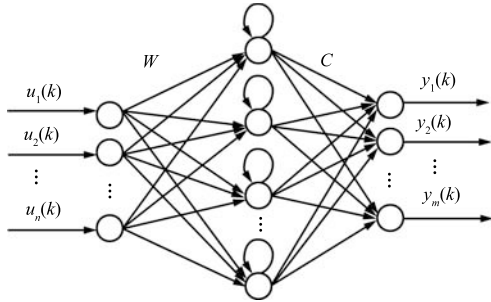


图 1 LRGF 神经网络结构图

Fig. 1 The structure of LRGF neural network

图 1 中, $\mathbf{u}(k) = [u_1(k), u_2(k), \dots, u_n(k)]^T$ 和 $\mathbf{y}(k) = [y_1(k), y_2(k), \dots, y_m(k)]^T$ 分别为神经网络在时刻 k 的输入、输出向量; W 和 C 分别为输入输出权值矩阵. 假设隐层对应的输入输出函数为 $\Phi(\cdot)$, 则神经网络的输入输出表达式为

$$\mathbf{y}(k) = C\Phi(W\mathbf{u}(k)) \quad (1)$$

1.1 LRGF 神经网络的隐层神经元

为了找到神经网络的状态空间表达式, 首先对隐层神经元进行分析. 图 1 中隐层神经元的结构如图 2 所示.

图 2 中, $\sigma(\cdot)$ 为神经元的激励函数, 在本文中取双曲正切函数. 则隐层神经元的输入输出表达式为

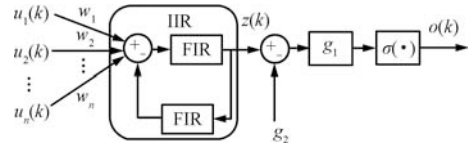


图 2 LRGF 神经网络的隐层神经元原理图

Fig. 2 The structure of the hidden neuron of LRGF neural network

$$\begin{cases} z(k) = \sum_{i=0}^r b_i \left(\sum_{j=1}^n w_j u_j(k-i) \right) - \sum_{i=1}^r a_i z(k-i) \\ o(k) = \sigma(g_1(z(k) - g_2)) \end{cases} \quad (2)$$

在上述函数中, 对神经网络的输入 $\mathbf{u}(k)$ 进行加权求和, 再通过由两个 FIR 滤波器组成的 IIR 滤波器后输出 $z(k)$. 其中前馈 FIR 和反馈 FIR 的阶次均为 r , 神经网络通过调整这两个 FIR 滤波器的系数, 实现对不同系统的动态特性描述. $o(k)$ 为隐层神经元的输出, g_1 为神经元激励函数输入的放大系数, g_2 为神经元激励函数输入的偏移. 对于图 2 隐层神经元结构, 第 i 个隐层神经元的状态空间表达为

$$\mathbf{x}_i(k+1) = A_i \mathbf{x}_i(k) + W_i \mathbf{u}(k) \quad (3)$$

$$o_i(k) = \sigma(g_1^i (b_i^T \mathbf{x}_i(k) + d_i^T \mathbf{u}(k) - g_2^i)) \quad (4)$$

式 (3) 中, $\mathbf{x}_i(k) = [x_1^i(k), x_2^i(k), \dots, x_r^i(k)]^T$ 为状态向量, $W_i = \mathbf{1}w_i^T$ 为状态方程输入矩阵 ($\mathbf{1} = [1, 1, 1, \dots, 1]^T$, $\mathbf{1} \in \mathbf{R}^{r \times 1}$), w_i 为输入层连接第 i 个隐层神经元的权值所构成的列向量. 状态转移矩阵 A_i 为

$$A_i = \begin{bmatrix} -a_1^i & -a_2^i & \cdots & -a_{r-1}^i & -a_r^i \\ 1 & 0 & \cdots & 0 & 0 \\ 0 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & 0 \end{bmatrix}$$

式 (4) 中, $o_i(k)$ 为神经元的输出, g_1^i 为神经元激励函数输入的放大系数, g_2^i 为神经元激励函数输入的偏移. 其中 $g_1^i > 0$, $\mathbf{b}_i = [b_1^i - b_0^i a_1^i, b_2^i - b_0^i a_2^i, \dots, b_r^i - b_0^i a_r^i]^T$ 为第 i 个隐层神经元中 IIR 权值系数构成的向量, $\mathbf{d}_i = [b_0^i w_i^1, \dots, b_0^i w_i^n]^T$.

1.2 LRGF 神经网络的状态空间表达

根据第 1.1 节中隐层神经元的原理以及 LRGF 神经网络结构, 可得神经网络的状态空间表达式为

$$\begin{cases} \mathbf{x}(k+1) = \mathbf{A}\mathbf{x}(k) + \mathbf{W}\mathbf{u}(k) \\ \mathbf{y}(k) = \mathbf{C}\sigma[\mathbf{G}_1(\mathbf{B}\mathbf{x}(k) + \mathbf{D}\mathbf{u}(k) - \mathbf{G}_2)] \end{cases} \quad (5)$$

令 $N = v \times r$, 则 $\mathbf{x} \in \mathbf{R}^{N \times 1}$ 为状态向量. $\mathbf{u} \in \mathbf{R}^{n \times 1}$, $\mathbf{y} \in \mathbf{R}^m$ 分别为输入输出向量. $\mathbf{A} \in \mathbf{R}^{N \times N}$, $\mathbf{A} = \text{diag}\{A_1, \dots, A_v\}$, 其对角块 A_1, \dots, A_v 为各隐层神经元的状态转移矩阵. $\mathbf{W} \in \mathbf{R}^{N \times n}$, $\mathbf{C} \in \mathbf{R}^{m \times v}$ 分别为输入输出权值矩阵. $\mathbf{B} \in \mathbf{R}^{v \times N}$ 为各隐层神经元 IIR 中的系数构成的块对角阵. $\mathbf{D} \in \mathbf{R}^{v \times n}$, $\mathbf{G}_1 \in \mathbf{R}^{v \times v}$, $\mathbf{G}_2 \in \mathbf{R}^{v \times 1}$, 非线性向量激发函数 $\sigma: \mathbf{R}^{v \times 1} \rightarrow \mathbf{R}^{v \times 1}$. $\mathbf{W}, \mathbf{B}, \mathbf{D}, \mathbf{G}_1, \mathbf{G}_2$ 矩阵形式为

$$\mathbf{W} = \begin{bmatrix} W_1 \\ W_2 \\ \vdots \\ W_v \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} b_1^T & 0 & \cdots & 0 \\ 0 & b_2^T & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & b_v^T \end{bmatrix}$$

$$\mathbf{D} = \begin{bmatrix} b_0^1 w_1^T \\ b_0^2 w_2^T \\ \vdots \\ b_0^v w_v^T \end{bmatrix}, \quad \mathbf{G}_1 = \begin{bmatrix} g_1^T & 0 & \cdots & 0 \\ 0 & g_2^T & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & g_v^T \end{bmatrix}$$

$$\mathbf{G}_2 = \begin{bmatrix} g_2^1 & g_2^2 & \cdots & g_2^v \end{bmatrix}^T$$

根据线性离散动态系统的稳定性条件和状态空间表达式 (5) 可知, 为使动态系统稳定, 式 (5) 中的状态转移矩阵 \mathbf{A} 的特征值必须在复平面单位圆内. 即对于方程

$$\det(z\mathbf{I} - \mathbf{A}) = 0 \quad (6)$$

\mathbf{A} 的特征值必须满足

$$\forall |z_i| < 1 \quad (7)$$

其中, $\mathbf{I} \in \mathbf{R}^{N \times N}$ 单位阵. 由于 \mathbf{A} 是块对角矩阵, 式 (6) 可以写成

$$\prod_{i=1}^v \det(z_i \mathbf{I} - \mathbf{A}_i) = 0 \quad (8)$$

式 (8) 中, $\mathbf{I} \in \mathbf{R}^{r \times r}$.

因此对于这种 LRGF 神经网络, 其稳定性是由每个包含递归的隐层神经元决定的, 即对于每个隐层神经元, 为了保证神经网络的稳定, 其 IIR 的极点必须在单位圆内. 根据这一原理, 本文提出了一种新的局部递归神经网络——基于极点配置稳定的 LRGF 神经网络. 该网络具有参数到稳定区域投影计算简单的特性.

2 基于极点配置稳定的 LRGF 神经网络

为清楚地阐述基于极点配置稳定的 LRGF 神经

网络, 本节设神经网络有 n 个输入, v 个隐层神经元, m 个输出, 并且动态神经元中的 IIR 滤波器的阶次为 2.

根据第 1 节对 LRGF 神经网络的稳定性分析可知, LRGF 神经网络稳定的充分必要条件为所有隐层神经元内的 IIR 稳定, 因此本节将首先展开讨论隐层神经元内 IIR 的极点配置问题, 并通过改造隐层神经元的结构达到使其能够配置隐层神经元极点的目的.

2.1 基于极点配置原理的隐层神经元

根据第 1.1 节, LRGF 神经网络第 i 个隐层神经元的表达式可表述为

$$\begin{cases} z_i(k) = b_i^0 \sum_{l=1}^n w_{li} u_l(k) + b_i^1 \sum_{l=1}^n w_{li} u_l(k-1) + \\ b_i^2 \sum_{l=1}^n w_{li} u_l(k-2) - a_i^1 z_i(k-1) - \\ a_i^2 z_i(k-2) \\ o_i(k) = \sigma(g_i^1(z_i(k) - g_i^2)) \end{cases} \quad (9)$$

上式中, w_{li} 为输入层第 i 输入到第 l 隐层神经元的权值, u_i 为神经网络的第 i 输入, a_i^1, a_i^2 和 b_i^0, b_i^1, b_i^2 分别为神经元 IIR 滤波器的系数, z_i 为滤波器的输出, $\sigma(\cdot)$ 为神经元的激励函数, g_i^1 和 g_i^2 为激励函数输入的偏移和放大倍数, o_i 为神经元的输出. 将上述神经元的 IIR 滤波器写成传递函数形式有

$$z_i(k) = \frac{b_i^0 + b_i^1 q^{-1} + b_i^2 q^{-2}}{1 + a_i^1 q^{-1} + a_i^2 q^{-2}} \sum_{l=0}^n w_{li} u_l(k) \quad (10)$$

上式中 q 为时间延迟, 从动态系统稳定性原理可知, 为了判别系统是否稳定, 需求解以下方程

$$z^2 + a_i^1 z + a_i^2 = 0 \quad (11)$$

假设求取到的极点为实数 c_i^1 和 c_i^2 , 则上述动态系统中的参数为

$$\begin{cases} a_i^1 = -c_i^1 - c_i^2 \\ a_i^2 = c_i^1 c_i^2 \end{cases} \quad (12)$$

若极点存在复数的情况, 则它们是共轭成对出现的. 设为 $d_i + h_i j$ 和 $d_i - h_i j$. 动态系统中 IIR 中的系数为

$$\begin{cases} a_i^1 = -2d_i \\ a_i^2 = (d_i)^2 + (h_i)^2 \end{cases} \quad (13)$$

因此当在极点为复数的情况下, 对神经网络训练的时候只需用学习算法估计极点的实部 d_i 和虚部 h_i 即可.

依据上述极点的两种情况, 基于极点配置原理的隐层神经元被设计为两个 IIR 的组合结构, 一个 IIR 包含两个实数极点, 另一个 IIR 包含一对共轭复数极点, 在这两种极点的 IIR 之间通过加权的方式切换, 如图 3 所示.

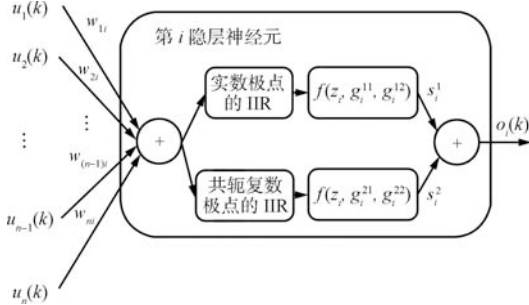


Fig. 3 The structure of the hidden neuron of stable LRGF neural network with pole assignment approach

神经网络的输入向量经过加权求和后, 分别经过隐层神经元的两个 IIR 滤波器 (实数极点的 IIR 和共轭复数极点的 IIR), 然后通过函数 f 输出两路数据. 隐层神经元的输出为两路数据经过 s_i^1 和 s_i^2 加权后的结果. 函数 f 的表达式为

$$f(z, g_1, g_2) = \tanh(g_1(z - g_2)) \quad (14)$$

g_i^{11} 和 g_i^{12} 分别为第 i 个隐层神经元实极点 IIR 部分激励函数的输入放大系数和输入偏移; g_i^{21} 和 g_i^{22} 分别为第 i 个隐层神经元共轭复数极点 IIR 部分激励函数的输入放大系数和输入偏移. 当隐层神经元的动态特性为实数极点所配置时, s_i^1 趋于 1, 而 s_i^2 趋于 0; 当隐层神经元的动态特性由复平面极点所配置时, s_i^1 趋于 0, 而 s_i^2 趋于 1. 由于神经元内的权值起到切换两种 IIR 的动态特性的功能, 因此本文采用的权值函数的方法, 权值函数应具备开关特性, 使其达到正确选择具有某种极点属性的 IIR 的目的. 这种权值函数通常可以为 0-1 函数, 即

$$f(\beta) = \begin{cases} 1, & \beta > \varepsilon \\ 0, & \beta \leq \varepsilon \end{cases} \quad (15)$$

但 0-1 函数不能求导, 不能用反向传播方法学习神经网络, 因此本文采用另外一种近似于 0-1 函数的 sigmod 函数. 同时在函数内设置一参数 β_i , 通过 sigmod 函数实现神经元内两种 IIR (实数极点 IIR 和共轭复数极点 IIR) 之间的切换, 则第 i 隐层神经元实数极点 IIR 对应输出权值为

$$s_i^1 = \frac{1}{1 + e^{-\beta_i}} \quad (16)$$

共轭复数极点 IIR 的输出权值为

$$s_i^2 = \frac{1}{1 + e^{\beta_i}} \quad (17)$$

以上所述的神经元结构, 使得隐层神经元能通过神经网络的训练, 估计权值函数的参数 β_i 和极点变量 c_i^1 、 c_i^2 、 d_i 、 h_i 的数值, 达到配置隐层神经元极点的属性和复平面中位置的目的. 因此将第 i 个隐层神经元表示为含有极点变量和权值函数参数变量的方程为

$$u_i'(k) = \sum_{l=1}^n w_{il} u_l(k) \quad (18)$$

$$z_i^1(k) = b_i^{01} u_i'(k) + b_i^{11} u_i'(k-1) + b_i^{21} u_i'(k-2) + (c_i^1 + c_i^2) z_i^1(k-1) - (c_i^1 c_i^2) z_i^1(k-2) \quad (19)$$

$$z_i^2(k) = b_i^{02} u_i'(k) + b_i^{12} u_i'(k-1) + b_i^{22} u_i'(k-2) + 2d_i z_i^2(k-1) - ((d_i)^2 + (h_i)^2) z_i^2(k-2) \quad (20)$$

$$o_i(k) = \frac{1}{1 + e^{-\beta_i}} f(z_i^1(k), g_i^{11}, g_i^{12}) + \frac{1}{1 + e^{\beta_i}} f(z_i^2(k), g_i^{21}, g_i^{22}) \quad (21)$$

在式 (18)~(21) 中, $u_i'(k)$ 为时刻 k 第 i 个隐层神经元的输入, 为实数极点 IIR 在时刻 k 的输出, b_i^{01} , b_i^{11} , b_i^{21} 为实数极点 IIR 部分的前馈 FIR 系数, b_i^{02} , b_i^{12} , b_i^{22} 为共轭复数极点 IIR 部分的前馈 FIR 系数, c_i^1 和 c_i^2 为实数极点 IIR 的两个实数极点, 为共轭复数极点 IIR 在时刻 k 的输出, d_i 和 h_i 分别为共轭复数极点 IIR 内极点的实部和虚部, $o_i(k)$ 为 k 时刻第 i 个隐层神经元的输出, g_i^{11} 和 g_i^{12} 分别为实数极点 IIR 所对应的输出激励函数的输入放大倍数和偏移, g_i^{21} 和 g_i^{22} 分别为共轭复数极点 IIR 所对应的输出激励函数的输入放大倍数和偏移.

将这种基于极点配置稳定的 LRGF 神经网络写成状态空间的形式, 第 i 个隐层神经元的表达式为

$$\mathbf{x}_i(k+1) = A_i \mathbf{x}_i(k) + W_i \mathbf{u}(k) \quad (22)$$

其中, $\mathbf{x}_i(k) = [\mathbf{x}_{i1}(k)^T, \mathbf{x}_{i2}(k)^T]^T$, $\mathbf{x}_{i1}(k)$ 为实数极点 IIR 的状态向量, $\mathbf{x}_{i2}(k)$ 为共轭复数极点 IIR 的状态向量. 状态转移矩阵 $A_i = \text{diag}\{A_{i1}, A_{i2}\}$ 为块对角矩阵, 其矩阵块元素为

$$A_{i1} = \begin{bmatrix} 1 & c_i^1 + c_i^2 & -c_i^1 c_i^2 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$$

$$A_{2i} = \begin{bmatrix} 1 & 2d_i & -(d_i)^2 - (h_i)^2 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$$

隐层神经元的输出表达式为

$$o_i(k) = \left[\frac{1}{1+e^{-\beta_i}} \quad \frac{1}{1+e^{\beta_i}} \right] \times \begin{bmatrix} f(\mathbf{b}_{i1}^T \mathbf{x}_i(k) + \mathbf{d}_{i1}^T \mathbf{u}(k), g_i^{11}, g_i^{21}) \\ f(\mathbf{b}_{i2}^T \mathbf{x}_i(k) + \mathbf{d}_{i2}^T \mathbf{u}(k), g_i^{12}, g_i^{22}) \end{bmatrix} \quad (23)$$

其中, $\mathbf{b}_{i1} = [b_i^{11} - b_i^{01}a_i^{11}, b_i^{21} - b_i^{01}a_i^{21}, 0, 0]^T$, b_i^{01} , b_i^{11} , b_i^{21} , a_i^{11} 和 a_i^{21} 为实数极点 IIR 的权值系数, $\mathbf{b}_{i2} = [b_i^{12} - b_i^{02}a_i^{12}, b_i^{22} - b_i^{02}a_i^{22}, 0, 0]^T$, b_i^{02} , b_i^{12} , b_i^{22} , a_i^{12} 和 a_i^{22} 为共轭复数极点 IIR 的权值系数, $\mathbf{d}_{i1} = [b_i^{01}w_{1i}, \dots, b_i^{01}w_{ni}]$, $\mathbf{d}_{i2} = [b_i^{02}w_{1i}, \dots, b_i^{02}w_{ni}]$.

2.2 基于极点配置稳定的 LRGF 神经网络的状态空间表达

由第 2.1 节中对隐层神经元的分析, 可得到基于极点配置稳定的 LRGF 神经网络的表达式为

$$\mathbf{x}(k+1) = A\mathbf{x}(k) + W\mathbf{u}(k) \quad (24)$$

$$\mathbf{y}(k) = G_3 C \Lambda f(B\mathbf{x}(k) + D\mathbf{u}(k), G_1, G_2) - G_4 \quad (25)$$

令 $N = v \times r$, 则 $\mathbf{x} \in \mathbf{R}^{2N \times 1}$ 为状态向量. $\mathbf{u} \in \mathbf{R}^{n \times 1}$, $\mathbf{y} \in \mathbf{R}^m$ 分别为输入输出向量. $A \in \mathbf{R}^{2N \times 2N}$, $A = \text{diag}\{A_1, \dots, A_v\}$, 其对角块 A_1, \dots, A_v 为各隐层神经元的状态转移矩阵. $W \in \mathbf{R}^{2N \times n}$, $C \in \mathbf{R}^{m \times v}$ 分别为输入输出权值矩阵. $B \in \mathbf{R}^{2v \times 2N}$ 为各隐层神经元 IIR 中的系数构成的块对角阵. $D \in \mathbf{R}^{2v \times n}$, $G_1 \in \mathbf{R}^{v \times 2}$, $G_2 \in \mathbf{R}^{v \times 2}$, 非线性向量激发函数 $f: \mathbf{R}^{2v \times 1} \rightarrow \mathbf{R}^{2v \times 1}$. Λ 为块对角矩阵, 块元素为隐层神经元内两种极点 IIR 所对应激励函数的输出权值构成的向量. 其中,

$$\Lambda = \begin{bmatrix} \varepsilon_1 & & & 0 \\ & \varepsilon_2 & & \\ & & \ddots & \\ 0 & & & \varepsilon_v \end{bmatrix}, \quad \varepsilon_i = \begin{bmatrix} 1 & 1 \\ 1 + e^{-\beta_i} & 1 + e^{\beta_i} \end{bmatrix}$$

B 也为块对角矩阵, B 和 D 的表达式为

$$B = \begin{bmatrix} b_1^T & & & 0 \\ & b_2^T & & \\ & & \ddots & \\ 0 & & & b_v^T \end{bmatrix}, \quad b_i = \begin{bmatrix} b_{i1} & 0 \\ 0 & b_{i2} \end{bmatrix}$$

$$D = \begin{bmatrix} b_1^0 w_1^T \\ b_2^0 w_2^T \\ \vdots \\ b_v^0 w_v^T \end{bmatrix}, \quad b_i^0 = [b_i^{01} \quad b_i^{02}]^T$$

激励函数的输入放大倍数和偏移参数分别构成的矩阵 G_1 和 G_2 , 以及神经网络输出层的放大倍数 G_3 和偏移 G_4 为

$$G_1 = \begin{bmatrix} g_1^{11} & g_1^{21} \\ g_2^{11} & g_2^{21} \\ \vdots & \vdots \\ g_v^{11} & g_v^{21} \end{bmatrix}, \quad G_2 = \begin{bmatrix} g_1^{21} & g_1^{22} \\ g_2^{21} & g_2^{22} \\ \vdots & \vdots \\ g_v^{21} & g_v^{22} \end{bmatrix}$$

$$G_3 = \begin{bmatrix} g_1^3 & & & 0 \\ & g_2^3 & & \\ & & \ddots & \\ 0 & & & g_m^3 \end{bmatrix}, \quad G_4 = \begin{bmatrix} g_1^4 \\ g_2^4 \\ \vdots \\ g_m^4 \end{bmatrix}$$

由式 (24) 可知, 状态转移矩阵 A 决定了这种神经网络的稳定性. A 内的块对角矩阵元素为

$$A_i = \begin{bmatrix} A_{i1} & 0 \\ 0 & A_{i2} \end{bmatrix}, \quad i = 1, 2, \dots, v$$

所以 $\det(Iz - A) = 0$ 可以写成

$$\prod_{i=1}^v [\det(Iz - A_{i1}) \det(Iz - A_{i2})] = 0 \quad (26)$$

根据第 1 节阐述的稳定性条件, 即 A 内的对角块元素矩阵的特征值必须在复平面单位圆内. 因此由式 (26) 可知, 每个隐层神经元内的 IIR 的稳定性决定了神经网络的动态稳定性.

3 基于极点配置稳定的 LRGF 神经网络的学习算法

训练神经网络时, 通常可以采用 BP 算法、共轭梯度下降法、遗传算法、粒子群算法以及 SPSA 算法等. 然而实验表明 BP 算法相对于其他几种训练算法可以获得更好的训练结果, 因此这里使用基于反向传播原理的算法来训练本文提出的新型 LRGF 神经网络.

定义训练的目标函数为

$$J = \frac{1}{2L} \sum_{k=1}^L \|\mathbf{y}(k) - \hat{\mathbf{y}}(k)\|_2 + \lambda_1 \|W\|_F + \lambda_2 \|C\|_F \quad (27)$$

L 为训练数据的样本数, $\mathbf{y}(k)$ 为实际对象的输出向量, $\hat{\mathbf{y}}(k)$ 为动态神经网络的输出向量, k 为离散时

间. 由于过大的权值矩阵对于减小神经网络模型误差没有贡献, 且可能导致神经网络在计算时数值超出计算机表达范围, 因此本文采用调整系数的方法来避免此问题, 该方法能使得神经网络训练目标函数高维曲面更加平滑, 有益于目标函数收敛到最小值上. λ_1 和 λ_2 分别为神经网络输入权值矩阵 W 和输出权值矩阵 C 的调整系数, 其取值通常通过正交检验的方法来决定. 通过正交测试, 选取测试集中使得神经网络的模型误差最小的那对数值作为调整系数. 式 (27) 中矩阵 W 和 C 的范数为 F 范数, 即 $\|W\|_F = \text{tr}(W_T W)$, $\|C\|_F = \text{tr}(C_T C)$.

通过对目标函数求偏导, 得到目标函数梯度下降的方向, 从而调节式 (24) 和式 (25) 里的神经网络参数 $W, C, B, D, G_1, G_2, G_3, G_4$ 和 Λ , 使目标函数 J 达到最小. 在参数更新中, 为了加快神经网络的学习速度, 防止学习过程发散, 本文采用自适应步长优化方法, 即经过一次参数更新后, 若目标函数减小, 则增大学习步长 α , 若目标函数增大, 则减小学习步长 α . 在参数更新中, 同时采用动量学习方法来避免神经网络参数陷入局部极小区域. 由此可得参数更新方程为

$$\Delta p(t+1) = \gamma \Delta p(t) - \alpha(1-\gamma) \frac{\partial J}{\partial p} \quad (28)$$

式中, p 为神经网络内的参数, α 为学习速率, γ 为动量系数 ($0 < \gamma < 1$), t 为训练批次.

3.1 神经网络内各参数的梯度方向求取

3.1.1 输出层

对式 (27) 中的 G_3 和 G_4 求偏导, 写成标量形式为

$$\frac{\partial J}{\partial g_i^3} = \frac{1}{L} \sum_{k=1}^L (\hat{y}_i(k) - y_i(k)) \times \left(\sum_{j=1}^v c_{ji} o_j(k) - g_i^4 \right) \quad (29)$$

$$\frac{\partial J}{\partial g_i^4} = \frac{1}{L} \sum_{k=1}^L ((\hat{y}_i(k) - y_i(k)) g_i^3) \quad (30)$$

$o_j(k)$ 表示在一次训练中 k 时刻第 j 个隐层神经元输出, $i = 1, \dots, m$. 对式 (27) 中的输出权值求偏导得到

$$\frac{\partial J}{\partial c_{ji}} = \frac{1}{L} \sum_{k=1}^L (\hat{y}_i(k) - y_i(k)) g_i^3 o_j(k) + 2\lambda_2 \sum_{l=1}^m c_{ji} \quad (31)$$

c_{ji} 为第 j 个隐层神经元到第 i 个输出的权值. 目标函数对隐层神经元的输出在时刻 k 的偏导数为

$$\frac{\partial J}{\partial o_j(k)} = -\frac{1}{L} (\hat{y}_i(k) - y_i(k)) c_{ji} g_i^3 \quad (32)$$

3.1.2 隐层

对第 j 个隐层神经元, 目标函数对 $z_j^1(k)$ 和 $z_j^2(k)$ 在时刻 k 的偏导数为

$$\frac{\partial J}{\partial z_j^1(k)} = \frac{\partial J}{\partial o_j(k)} \frac{1}{1 + e^{-\beta_j}} \sigma' (g_j^{11} (z_j^1(k) - g_j^{21})) g_j^{11} \quad (33)$$

$$\frac{\partial J}{\partial z_j^2(k)} = \frac{\partial J}{\partial o_j(k)} \frac{1}{1 + e^{\beta_j}} \sigma' (g_j^{12} (z_j^2(k) - g_j^{22})) g_j^{12} \quad (34)$$

对于实数极点 IIR, 目标函数对其相应激励函数的输入放大系数 g_j^{11} 和输入偏移 g_j^{21} 在时刻 k 的偏导数为

$$\frac{\partial J}{\partial g_j^{11}}(k) = \frac{\partial J}{\partial o_j(k)} \frac{1}{1 + e^{-\beta_j}} \times \sigma' (g_j^{11} (z_j^1(k) - g_j^{21}) (z_j^1(k) - g_j^{21})) \quad (35)$$

$$\frac{\partial J}{\partial g_j^{21}} = \frac{\partial J}{\partial o_j(k)} \frac{1}{1 + e^{-\beta_j}} \sigma' (g_j^{11} (z_j^2(k) - g_j^{21})) g_j^{11} \quad (36)$$

目标函数对具有两个实数极点的 IIR 滤波器中的各参数在 k 时刻的偏导数为

$$\frac{\partial J}{\partial b_j^{i1}}(k) = \frac{\partial J}{\partial o_j(k)} \frac{\partial o_j(k)}{\partial z_j^1(k)} u_j'(k-i) \quad (37)$$

$$\frac{\partial J}{\partial c_j^1}(k) = \frac{\partial J}{\partial o_j(k)} \frac{\partial o_j(k)}{\partial z_j^1(k)} (u_j'(k-1) - c_j^2 u_j'(k-2)) \quad (38)$$

$$\frac{\partial J}{\partial c_j^2}(k) = \frac{\partial J}{\partial o_j(k)} \frac{\partial o_j(k)}{\partial z_j^1(k)} (u_j'(k-1) - c_j^1 u_j'(k-2)) \quad (39)$$

$u_j'(k)$ 为第 j 个隐层神经元在时刻 k 的输入, c_j^i 和 c_j^2 为第 j 个隐层神经元实数极点部分 IIR 内的两个极点, b_j^{i1} 为第 j 个隐层神经元内实数极点 IIR 内前馈 FIR 的第 i 个系数.

同理, 目标函数对共轭复数极点 IIR 内参数的偏导数表达式可表述为

$$\frac{\partial J}{\partial b_j^{i2}}(k) = \frac{\partial J}{\partial o_j(k)} \frac{\partial o_j(k)}{\partial z_j^2(k)} u_j'(k-i) \quad (40)$$

$$\frac{\partial J}{\partial d_j}(k) = \frac{\partial J}{\partial o_j(k)} \frac{\partial o_j(k)}{\partial z_j^2(k)} \times (2u_j'(k-1) - 2d_j u_j'(k-2)) \quad (41)$$

$$\frac{\partial J}{\partial h_j}(k) = \frac{\partial J}{\partial o_j(k)} \frac{\partial o_j(k)}{\partial z_j^2(k)} \left(-2h_j u_j'(k-2) \right) \quad (42)$$

其中, d_j 和 h_j 为第 j 个隐层神经元共轭复数极点 IIR 内极点的实部和虚部。

对于第 j 个隐层神经元内的切换函数, 目标函数对切换权值系数 s_j^1 在 k 时刻的偏导为

$$\frac{\partial J}{\partial s_j^1}(k) = \frac{\partial J}{\partial o_j(k)} \times \left(\sigma(g_j^{11}(z_j^1(k) - g_j^{21})) - \sigma(g_j^{12}(z_j^2(k) - g_j^{22})) \right) \quad (43)$$

而根据切换权值函数表达式 (16) 和 (17) 可知, 两函数轴对称, 权值的关系式为

$$s_j^2 = 1 - s_j^1 \quad (44)$$

因此只需估计第 j 隐层神经元内的 s_j^1 即可得到 s_j^2 。

3.1.3 输入层

由链式法则可知, 目标函数对时刻 k 隐层神经元的输入求偏导数, 其结果为

$$\frac{\partial J}{\partial u_j'(k)} = \frac{\partial J}{\partial o_j(k)} \left(\frac{\partial o_j(k)}{\partial z_j^1(k)} b_j^{01} + \frac{\partial o_j(k)}{\partial z_j^2(k)} b_j^{02} \right) \quad (45)$$

在时刻 k 隐层神经元的输入对神经网络输入层权值 w_{ij} 的偏导数为

$$\frac{\partial u_j'(k)}{\partial w_{ij}} = \frac{\partial}{\partial w_{ij}} \left(\sum_{i=1}^m w_{ij} u_i(k) \right) = u_i(k) \quad (46)$$

因此目标函数对输入层权值在 k 时刻的偏导数为

$$\frac{\partial J}{\partial w_{ij}}(k) = \frac{\partial J}{\partial u_j'(k)} u_i(k) \quad (47)$$

3.2 隐层神经元极点的配置和投影

为了实现配置隐层神经元的极点, 通过第 3.1 节中目标函数对极点参数求偏导, 得到其梯度下降的方向, 然后使用式 (28) 更新极点在复平面中的位置。

需要注意的是对隐层神经元内权值函数中 β_j 参数的更新方式与其他参数的更新方式不同。为了使隐层神经元内权值具有切换开关的功能, 在调整参数时, 根据目标函数对权值的偏导数式 (43) 来调节权值函数内参数 β_j , 从而配置隐层神经元极点的属性。在 $t+1$ 训练批次, β_j 的改变量为

$$\Delta \beta_j(t+1) = \gamma \Delta \beta_j(t) - \alpha(1-\gamma) \sum_{k=1}^L \frac{\partial J}{\partial s_j^1}(k) \quad (48)$$

此外, 因为在隐层神经元实数极点 IIR 部分中极点 c_j^1 和 c_j^2 的目标函数梯度下降方向是相互决定的, 所

以对极点 c_j^1 和 c_j^2 的学习更新与式 (28) 不同。令 $e_i = c_i^1 - c_i^2$, 则

$$e_j(t+1) = (1-\gamma)e_j(t) + \gamma e_j(t-1) + \alpha(1-\gamma) \sum_{k=1}^L \left(\frac{\partial J}{\partial c_j^1}(k) - \frac{\partial J}{\partial c_j^2}(k) \right) \quad (49)$$

将式 (38) 和式 (39) 代入上式得到

$$e_j(t+1) = \left(1-\gamma + \alpha(1-\gamma) \sum_{k=1}^L \left(\frac{\partial J}{\partial o_j(k)} \frac{\partial o_j(k)}{\partial z_j^1(k)} u_j'(k-2) \right) \right) \times e_j(t) + \gamma e_j(t-1) \quad (50)$$

写成矩阵形式为

$$\begin{cases} \begin{pmatrix} e_j(t+1) \\ e_j(t) \end{pmatrix} = H \begin{pmatrix} e_j(t) \\ e_j(t-1) \end{pmatrix} \\ H = \begin{pmatrix} 1-\gamma + \alpha(1-\gamma) \times \sum_{k=1}^L \left(\frac{\partial J}{\partial o_j(k)} \frac{\partial o_j(k)}{\partial z_j^1(k)} u_j'(k-2) \right) & \gamma \\ 1 & 0 \end{pmatrix} \end{cases} \quad (51)$$

当上式中的状态转移矩阵 H 的特征值在单位圆内时 $e_j(t) \rightarrow 0$, 因此在学习的过程中可能存在 $c_j^1 = c_j^2$ 的情况。由式 (38) 和式 (39) 可知, 在 $c_j^1 = c_j^2$ 时, J 对 c_j^1 和 c_j^2 的偏导数相等, 所以若仅按式 (28) 的参数更新方法, c_j^1 将恒等于 c_j^2 , 从而影响神经网络的学习效果。因此在对 c_j^1 和 c_j^2 更新的时候采用参数投影技术, 使

$$|c_j^1 - c_j^2| > \eta \quad (52)$$

其中, η 为极点间的最小距离, 应当取接近 0 的数值。

为了保证神经网络的稳定性, 根据离散系统的稳定性原理, 第 j 个隐层神经元内的极点相关参数必须满足以下不等式条件

$$\begin{cases} |c_j^1| < \psi \\ |c_j^2| < \psi \end{cases} \quad (53)$$

$$|d_j \pm h_j i| < \psi \quad (54)$$

上述不等式 (53) 和 (54) 中, ψ 为学习算法设定的投影区域半径 ($0 < \psi < 1$), 其取值可根据保留稳定余量大小而定, 同时为了避免限制极点的存在区域, ψ 的取值应当接近 1。在图 4 中, 阴影部分为极点稳定投影区域。根据式 (52) ~ (54) 可得, IIR 内的极点参数到稳定区域的投影计算为

$$\begin{cases} c_j^1 = \psi \text{sgn}(c_j^1), & |c_j^1| > \psi \\ c_j^2 = \psi \text{sgn}(c_j^2), & |c_j^2| > \psi \\ d_j = \frac{\psi}{|d_j + h_j i|} d_j, & |d_j + h_j i| > \psi \\ h_j = \frac{\psi}{|d_j + h_j i|} h_j, & |d_j + h_j i| > \psi \end{cases} \quad (55)$$

防止 c_j^1 和 c_j^2 在学习过程中进入恒等状态的投影计算为

$$\begin{cases} c_j^1 = c_j^2 - \eta_j, & |c_j^1 - c_j^2| < \eta \\ c_j^2 = c_j^1 - \eta_j, & |c_j^1 - c_j^2| < \eta \end{cases} \quad (56)$$

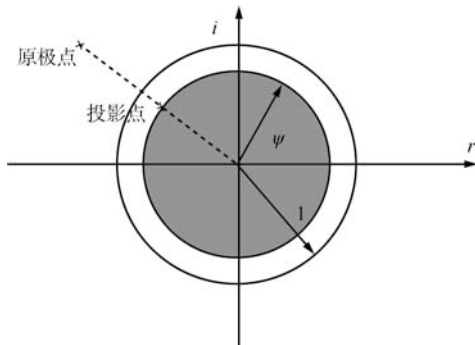


图 4 极点稳定区域投影

Fig. 4 Stable projection of the pole

4 计算分析和计算实例

为验证基于极点配置稳定的 LRGF 神经网络对动态系统逼近学习的可靠性和有效性,下面将对几种同类算法的计算复杂度进行分析和对比,并对线性系统和非线性系统进行逼近学习实验.同时为了进一步说明本文提出的神经网络的优势,最后将文献 [19–20] 中的另外两种神经网络和本文提出的神经网络对直流电机控制系统进行逼近学习实验,并作比较分析.

4.1 计算复杂度分析

常用的动态神经网络稳定投影方法有两种,一种是梯度投影 (Gradient projection, GP) 方法,另一种是最短距离投影 (MDP) 方法^[18].下面将本文提出的极点配置投影 (PAP) 方法和以上两种方法做计算复杂度对比.通过对三种投影方法的计算复杂度详细分析,得到表 1~3.

从表 1~3 可以看出,无论是在一阶 IIR 还是在二阶 IIR 的情况下, PAP 方法相对于 GP 和 MDP 方法的计算复杂度都要小.使用 PAP 方法在投影计算时,加法和乘法计算操作次数较少.在条件判断时,二阶 IIR 情况下, GP 方法的最大计算操作次数要小于 PAP 方法,但是从平均计算操作次数上

来说, PAP 方法仍然在计算复杂度上要小于 GP 方法. PAP 方法在计算复杂度上相对于 GP 方法优势不很明显,但 PAP 投影方法却具有 MDP 方法最短距离投影的功能,而 GP 方法不具有此功能,这对于下面将要验证的逼近学习性能至关重要.同时从表 2 和表 3 可以看到 PAP 方法在稳定区域参数投影计算复杂度上要远小于 MDP 方法.这得益于 PAP 方法在参数投影计算时,直接进行极点配置,避免了对稳定性约束方程组^[14]的求解.

表 1 GP 方法计算复杂度分析

Table 1 Computational complexity analysis of GP

计算类别	一阶 IIR			二阶 IIR		
	最小	最大	平均	最小	最大	平均
条件判断	1	1	1	3	3	3
设置操作	1	2	1.5	2	5	3.5
加法计算	1	1	1	2	2	2
乘法计算	1	1	1	2	2	2
总和	4	5	4.5	9	12	10.5

表 2 MDP 方法计算复杂度分析

Table 2 Computational complexity analysis of MDP

计算类别	一阶 IIR			二阶 IIR		
	最小	最大	平均	最小	最大	平均
条件判断	1	2	1.5	2	3	2.5
设置操作	0	1	0.5	0	2	1
加法计算	0	2	1	0	16	8
乘法计算	0	1	0.5	0	16	8
总和	1	6	3.5	2	37	19.5

表 3 PAP 方法计算复杂度分析

Table 3 Computational complexity analysis of PAP

计算类别	一阶 IIR			二阶 IIR		
	最小	最大	平均	最小	最大	平均
条件判断	1	2	1.5	1	4	2.5
设置操作	0	1	0.5	0	2	1
加法计算	0	0	0	0	1	0.5
乘法计算	0	0	0	0	6	3
总和	1	3	2	1	13	7

为测试本文提出的基于极点配置稳定的 LRGF 神经网络的学习计算速度,这里引用文献 [18] 的一个二阶线性动态过程计算实例来说明.这里将二阶

线性动态过程写成离散的形式

$$y(k) = A_1 y(k-1) + A_2 y(k-2) + B_1 u(k-1) + B_2 u(k-2) \quad (57)$$

上式中, $A_1 = 0.374861$, $A_2 = -0.367879$, $B_1 = 0.200281$, $B_2 = 0.140827$. 动态系统的输入数据为正弦信号组合序列, 具体的输入 $u(k)$ 为

$$u(k) = \begin{cases} \sin\left(\frac{\pi k}{25}\right), & k \leq 250 \\ 1, & 250 < k \leq 500 \\ -1, & 500 < k \leq 750 \\ 0.3\sin\left(\frac{\pi k}{25}\right) + 0.1\sin\left(\frac{\pi k}{32}\right) + 0.6\sin\left(\frac{\pi k}{10}\right), & 750 < k \leq 1000 \end{cases} \quad (58)$$

为比较三种稳定投影方法的耗时, 对拥有相同个数的隐层神经元在不同学习次数时, 以及相同学习次数而拥有不同个数的隐层神经元时分别进行了实验比较. 该实验通过 Visual studio 2010 C++ 编程, 在 PC 平台下 (2 G 内存, Intel E6550 CPU) 运行测试. 具体试验结果如表 4 所示 (表中的数据是经过 10 次实验后取得的平均值). 其中 v 为隐层神经元个数, r 为隐层神经元的阶次, n_{\max} 为最大训练次数.

表 4 计算实验性能对比

Table 4 Computational performance comparison

神经网络配置	稳定投影方法 (s)		
	GP	MDP	PAP
$v = 4, r = 2, n_{\max} = 500$	2.217	2.374	2.254
$v = 8, r = 2, n_{\max} = 500$	3.977	4.011	3.591
$v = 16, r = 2, n_{\max} = 500$	6.914	6.923	6.717
$v = 8, r = 2, n_{\max} = 5000$	40.233	40.245	39.142
$v = 8, r = 2, n_{\max} = 10000$	79.312	78.272	78.092

从表 4 数据可以看出, 使用不同投影方法的 LRGF 神经网络的训练计算时间花费大概相同, 但在训练次数较大时可以看出使用 PAP 方法所花费的时间稍短. 从理论上分析得出的投影计算复杂度知道, PAP 方法计算复杂度最小, 在隐层节点数为 4, 8, 16 时, 有大约 0.1~0.2 秒的差别. 在学习次数设定为 5000 时运算时间有大约 1 秒的差别. 其次是 GP 方法, 计算耗时最大的是 MDP 方法. 由于使用 PAP 方法时所使用的局部动态神经网络与使用 GP 和 MDP 方法的神经网络结构有所不同, 因此它们的神经网络计算复杂度不同. 整体上讲, 本文提出的 PAP 动态神经网络计算复杂度仍小于基于

GP 方法和 MDP 方法的 LRGF 神经网络.

4.2 线性动态系统的逼近

为验证基于极点配置稳定的 LRGF 神经网络对线性动态系统的学习逼近能力, 以及在逼近学习上的可靠性和有效性, 这里分别使用基于 PAP 方法和 MDP 方法的 LRGF 神经网络对线性系统进行逼近学习. 由于在文献 [14] 和文献 [20] 里已对 GP 方法和 MDP 方法进行了详细的比较和分析, 验证了在 LRGF 神经网络中使用 MDP 方法使得神经网络在学习逼近以及收敛速度性能比使用 GP 方法更优越, 因此这里只对 PAP 方法和 MDP 方法做比较仿真. 在该部分实验中, 使用第 4.1 节所采用的动态过程模型来验证本文提出的稳定神经网络模型对线性动态系统的逼近能力. 为测试神经网络在系统输出/输入带有噪声时的逼近能力, 本部分所使用的动态过程模型与第 4.1 节有所不同. 这里将系统的离散形式的方程改为

$$y(k) = A_1 y(k-1) + A_2 y(k-2) + B_1 (u(k-1) + e(k-1)) + B_2 (u(k-2) + e(k-2)) \quad (59)$$

上式中, $e(k) \in [-0.2, +0.2]$ 为系统在 k 时刻的高斯噪声. 其中 $u(k)$ 仍使用式 (58) 生成的数据. 极点配置稳定的 LRGF 神经网络的隐层神经元个数设定为 4, 神经网络学习前的初始值设定为:

- 1) 每个动态神经元的 IIR 初始状态全部设置为 0;
- 2) 每个隐层神经元内的 β_i 设置为 0;
- 3) 将隐层神经元内的实数极点 IIR 部分的极点设定为 -0.2 和 $+0.2$;
- 4) 将隐层神经元内的共轭复数极点 IIR 部分的极点设定为 $0.1 \pm 0.1i$;
- 5) 将隐层神经元的 IIR 输入权值系数设定为 $[-0.5, +0.5]$ 里的随机数;
- 6) 将神经网络的所有权值设定为 $[-0.1, +0.1]$ 里的随机数;
- 7) 设定神经网络内的所有激励函数的输入放大系数为 1, 输入偏移为 0;
- 8) 投影参数 ψ 设定为 0.9, η 设定为 0.05.

实验结果如图 5 所示. 从图 5 可以看到, 采用极点配置稳定的 LRGF 神经网络, 在线性动态系统输入/输出携带噪声情况下仍能有效地进行逼近.

图 6 和图 7 分别为采用 PAP 法和 MDP 法的 LRGF 神经网络的逼近误差平方和曲线. 由图 6 和图 7 可看出, 使用 PAP 的神经网络学习收敛速度更快. 使用 MDP 的 LRGF 神经网络在训练 50 次

后才经过误差平方和从急剧下降到缓慢下降的拐点, 而采用 PAP 的 LRGF 神经网络在 10 次学习后就 已经经过了此拐点. 从图 7 的输出逼近效果可以看出, 在输入信号带有噪声时, 原系统的输出噪声幅值 相对于神经网络输出的噪声幅值大, 所以极点配置 稳定神经网络在有效逼近原系统输出的同时, 又具 有抑制噪声的功能.

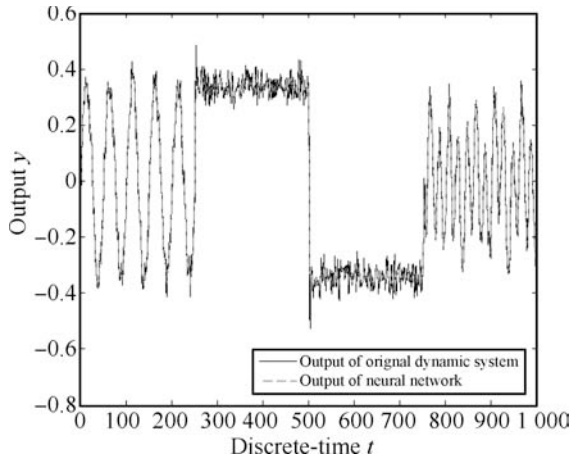


图 5 极点配置稳定的神经网络对线性动态 系统的逼近效果图

Fig. 5 The results of linear dynamic system modeled by stable LRGF neural network with pole assignment approach

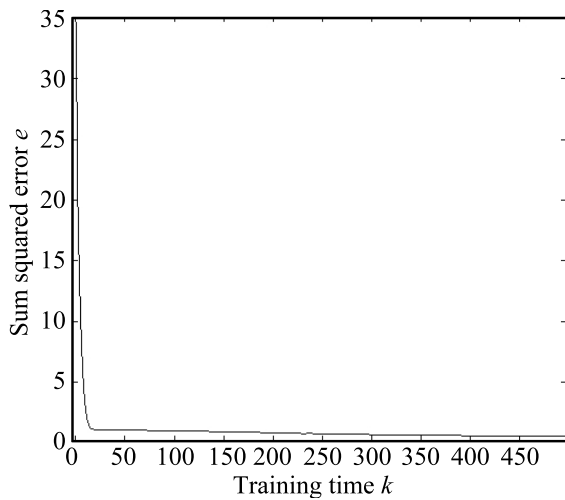


图 6 极点配置稳定的神经网络学习线性动态系统 过程误差平方和曲线图

Fig. 6 The curve of sum-squared error between the linear dynamic system and the stable LRGF neural network with pole assignment approach

隐层神经元的切换开关权值学习结果如表 5 所 示. 由表 5 可以看出, 对于第一个隐层神经元, 实数 极点部分的输出权值为 1, 共轭复数极点 IIR 部分 的输出权值为 0, 即此神经元呈实数极点的特性; 对

于第二隐层神经元, 由于最终实数极点部分的输出 权值为 0, 而共轭复数极点 IIR 部分的输出权值为 1, 所以神经元呈共轭复数极点 IIR 的动态特性; 同 理对于第三隐层神经元, 神经元呈共轭复数极点 IIR 动态特性; 对于第四隐层神经元, 神经元呈实数极点 IIR 动态特性. 因此, 使用本文提出的具有开关权值 隐层神经元的神经网络和学习方法对线性动态系统 建模是有效的.

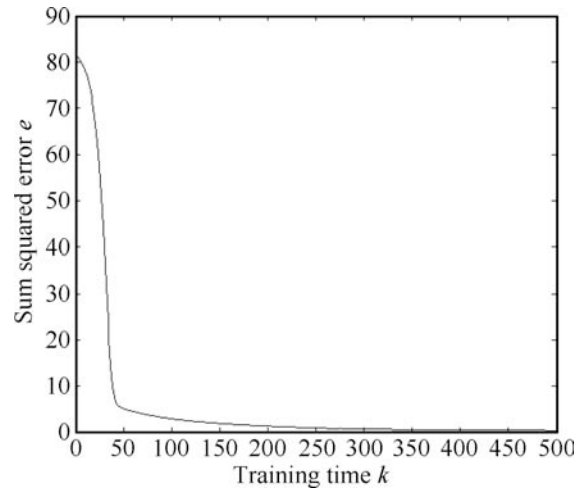


图 7 采用 MDP 投影方法的 LRGF 动态神经网络学习 线性动态系统过程的误差平方和曲线图
Fig. 7 The curve of sum-squared error between the linear dynamic system and the stable LRGF neural network with MDP approach

表 5 隐层神经元实数极点 IIR 输出权值函数值

Table 5 The values of weight functions of IIR with real pole in hidden neurons

隐层神经元	权值函数值
第一神经元	1
第二神经元	0
第三神经元	0
第四神经元	1

4.3 非线性动态系统的逼近

在非线性系统的逼近仿真试验里, 同样采用基 于 PAP 方法和 MDP 方法的 LRGF 神经网络对动 态系统建模. 由于文献 [18] 和 [21] 里已经对 MDP 方法和 GP 方法做了详细的比较, 并验证了基于 MDP 方法的神经网络在学习逼近和收敛速度上优 于采用 GP 方法的神经网络. 因此, 这里仅对使用 PAP 方法和 MDP 方法的 LRGF 神经网络进行对 比. 本部分的非线性动态对象实验模型来自文献 [22], 其模型由以下差分方程表达:

$$y_p(k+1) = \frac{y_p(k)y_p(k-1)y_p(k-2)u(k-1)[y_p(k-2)-1] + u(k)}{1 + y_p^2(k-1) + y_p^2(k-2)} + e(k) \quad (60)$$

上式中, $e(k) \in [-0.2, +0.2]$ 为系统在 k 时刻的高斯噪声. 其中 $u(k)$ 仍为式 (58) 生成的输入数据. 极点配置稳定的 LRGF 神经网络隐层神经元个数设为 4. 初始参数按照第 4.2 节设定的数据. 实验结果如图 8~10 所示.

根据图 8 和图 10 可知, 在非线性动态系统的逼近学习过程中, 采用 PAP 的 LRGF 神经网络初始段学习误差的下降速度比采用 MDP 方法的 LRGF 神经网络快. 虽然, 采用 PAP 的 LRGF 神经网络在第 15 次学习后下降速度有所变化, 但在学习结束后仍然能够收敛到比较好的结果. 因此, 对这种非线性动态系统, 采用 PAP 的 LRGF 神经网络仍然能够有效地工作. 隐层神经元的实数极点 IIR 和共轭复数极点 IIR 的输出权值学习结果如表 6 所示. 由表 6 可知, 对于式 (60) 所示的非线性动态系统, 神经网络的所有隐层神经元呈共轭复数极点 IIR 的动态特性, 而实数极点 IIR 输出权重全为 0.

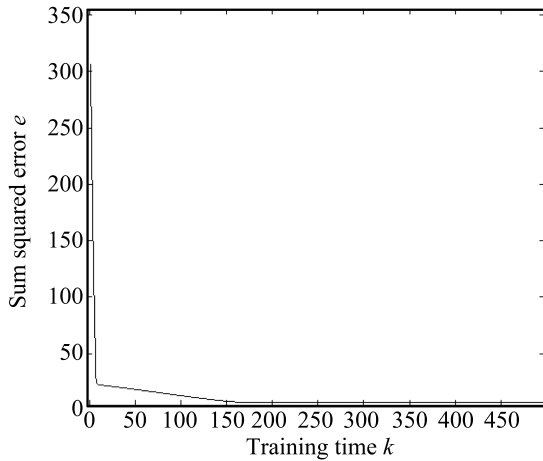


图 8 极点配置稳定的神经网络学习非线性动态系统过程误差平方和曲线图

Fig. 8 The curve of sum-squared error between the non-linear dynamic system and the stable LRGF neural network with pole assignment approach

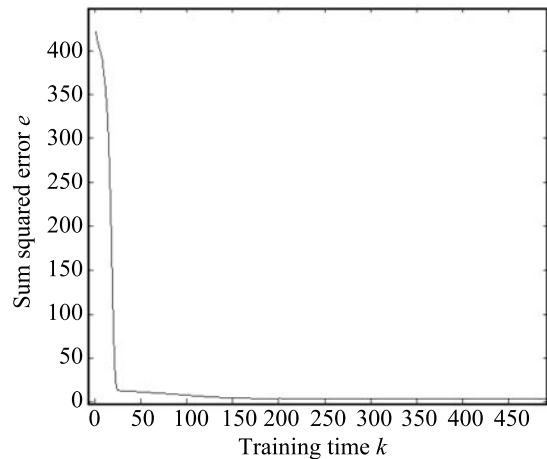


图 10 采用 MDP 投影方法的 LRGF 动态神经网络学习非线性动态系统过程的误差平方和曲线图

Fig. 10 The curve of sum-squared error between the non-linear dynamic system and the stable LRGF neural network with MDP approach

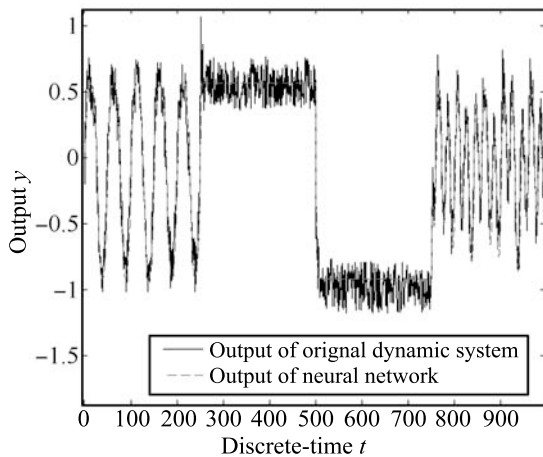


图 9 极点配置稳定的神经网络对非线性动态系统的逼近效果图

Fig. 9 The results of non-linear dynamic system modeled by stable LRGF neural network with pole assignment approach

从图 9 可以看到, 采用极点配置稳定的 LRGF 神经网络能有效逼近非线性动态系统的输出. 同样

表 6 神经元共轭复数极点 IIR 输出权值函数值

Table 6 The values of weight functions of IIR with complex-conjugate poles in hidden neurons

隐层神经元	权值函数值
第一神经元	0
第二神经元	0
第三神经元	0
第四神经元	0

4.4 对直流电机转速控制系统的逼近

本小节利用 Matlab + Simulink 对直流电机转

速控制系统进行仿真, 并使用本文提出的基于极点配置稳定的 LRGF 神经网络对控制系统逼近, 此外还分别用文献 [19–20] 中的混合型递归神经网络和全局递归神经网络对控制系统进行学习, 比较分析三种动态神经网络的学习实验结果.

实验中直流电机的技术参数为:

- 1) 终端电阻: 2.06 ohms;
- 2) 终端电感: 0.000238 H;
- 3) 反电动势常数: $60/(406 \times 2 \times \pi)$ V/rad;
- 4) 转矩常数: 0.0235 N·m/A;
- 5) 转子惯量: $1.07E-6$ kg·m²;
- 6) 机械阻尼: $12E-7$ N·m·s/rad.

电机的转子和负载之间通过弹性连接器连接, 负载的惯量为 $1/10.07E-6$ kg·m², 弹性连接器的弹性系数为 100 N/mm, 阻尼为 0.001 N·m·s/rad. 采用 PID 控制器控制电机的转速, 电机控制系统的输入信号为

$$u(k) = 3 \sin(2\pi \times 1.7t) + 3 \sin(2\pi \times 1.1t - \frac{\pi}{7}) + 3 \sin(2\pi \times 0.3t + \frac{\pi}{3}) \quad (61)$$

极点配置稳定的神经网络的隐层神经元个数设定为 4, 其初始参数取第 4.2 节中设定的数据. 对于混合型递归神经网络^[19], 同样设定神经网络的隐层神经元个数为 4, 3 个专家模块, 学习速率按照实验中的最佳效果设定, 为 $0.01/T$, T 为训练数据的样本数. 而全局递归网络^[20] 设定隐层神经元个数为 7, 根据实验的最佳效果设定学习速率为 $0.001/T$. 将样本数据归一化后分别用这三种递归神经网络训练, 得到实验结果如图 11~14.

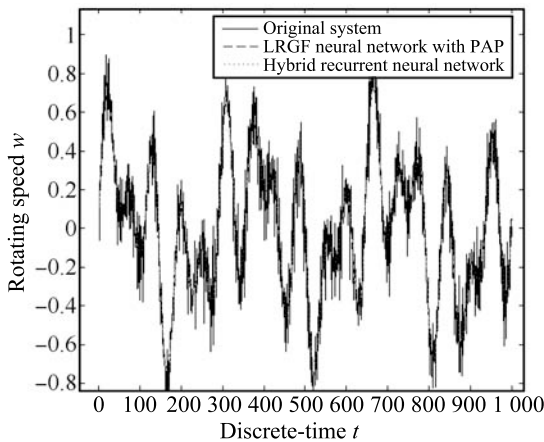


图 11 基于极点配置的 LRGF 神经网络和混合递归神经网络对直流电机控制系统的逼近效果图
Fig. 11 The results of DC motor control system modeled by hybrid recurrent neural network and stable LRGF neural network with pole assignment approach

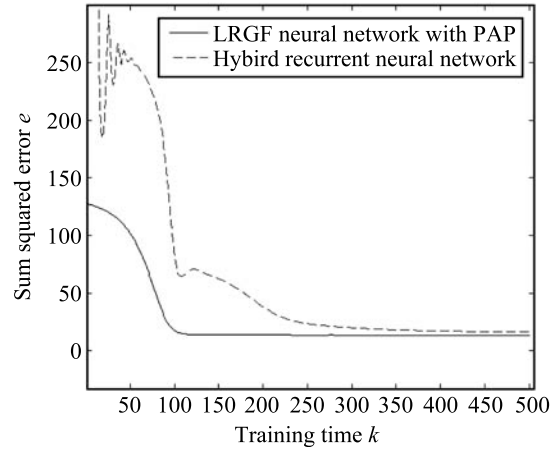


图 12 基于极点配置稳定的 LRGF 神经网络和混合递归神经网络学习过程误差平方和曲线图
Fig. 12 The curves of sum-squared error of the hybrid recurrent neural network and the stable LRGF neural network with pole assignment approach

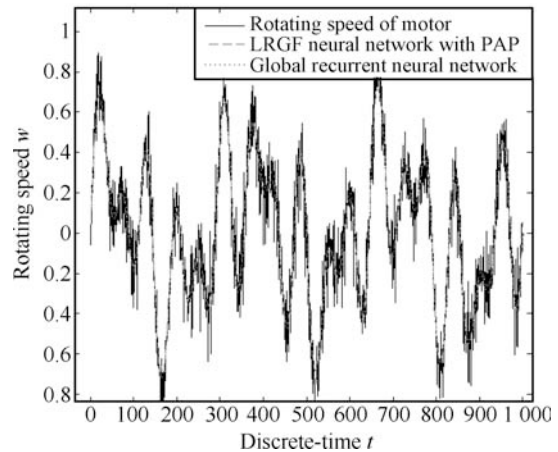


图 13 基于极点配置稳定的 LRGF 神经网络和全局递归神经网络对直流电机控制系统的逼近效果图
Fig. 13 The results of DC motor control system modeled by global recurrent neural network and stable LRGF neural network with pole assignment approach

实验中的三种神经网络的学习算法都是基于梯度下降原理. 从图 11 可以看到, 无论是使用混合递归神经网络^[19] 还是使用极点配置稳定的 LRGF 神经网络都能有效地逼近电机转速控制系统, 但在逼近程度上, 采用极点配置稳定的 LRGF 神经网络更好. 从图 12 可知, 在混合递归神经网络训练的初始阶段, 平方和误差曲线有些抖动, 接着误差急剧减小, 但总的来说, 其误差平方和几乎在整个学习过程中都大于基于极点配置稳定的 LRGF 神经网络. 从极点配置稳定的 LRGF 神经网络的误差曲线可以看到, 由于采用新的神经元结构和极点配置投影技术,

在对神经元极点直接配置的同时保证了神经网络在整个学习过程中的稳定性, 因此使得神经网络的学习误差曲线平滑, 收敛速度更快.

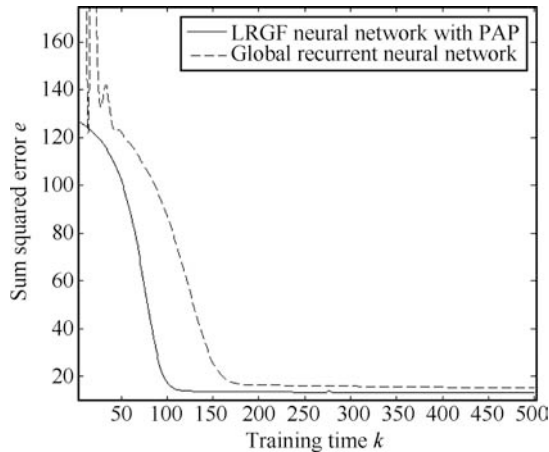


图 14 基于极点配置稳定的 LRGF 神经网络和全局递归神经网络学习过程误差平方和曲线图

Fig. 14 The curves of sum-squared error of the global recurrent neural network and the stable LRGF neural network with pole assignment approach

全局递归神经网络^[22]对电机转速控制系统的逼近效果如图 13. 使用文献 [20] 中的全局递归神经网络能有效地逼近电机转速控制系统, 但在逼近程度上, 极点配置稳定的 LRGF 神经网络较好. 在学习过程中, 同混合递归神经网络类似, 初始阶段的误差平方和曲线有较大波动 (如图 14). 采用极点配置稳定的 LRGF 神经网络较全局递归神经网络能更快地收敛.

5 总结

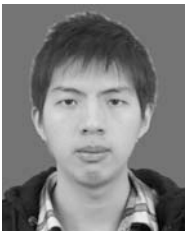
本文提出了一种新的采用极点配置稳定的 LRGF 神经网络. 通过极点配置的方法对神经网络隐层神经元内 IIR 的极点进行配置, 实现了对线性系统和非线性系统的学习逼近功能. 由仿真实验分析可以看到, 本文提出的神经网络在稳定投影方法的计算复杂度上小于 GP 方法和 MDP 方法, 而且在学习收敛速度上优于采用 MDP 投影方法的 LRGF 神经网络和其他一些递归神经网络^[21-22]. 此外, 采用 PAP 方法的 LRGF 神经网络在拥有学习收敛速度快的同时, 学习逼近性能较好, 且还具有稳定区域投影简单的优点. 而在 MDP 方法中, 需要带有多个约束条件的目标函数的最小值, 同时还需解方程组, 若神经元 IIR 超过 2 阶则这种方法在理论上求解过程太复杂. 本文提出的极点配置投影方法由于其投影操作简单, 无需求解高阶方程, 因

此为拥有高阶动态神经元的 LRGF 神经网络的参数稳定投影计算提供了一种新的解决方案. 对本文提出的极点配置投影技术在拥有高阶动态神经元的 LRGF 神经网络里的应用将是下一步研究工作.

References

- Boely N, Botez R M. New Approach for the identification and validation of a nonlinear F/A-18 model by use of neural networks. *IEEE Transactions on Neural Networks*, 2010, **21**(11): 1759-1765
- Han Bing, Han Min. Nonlinear time delay systems identification based on dynamic BP algorithm. *Journal of Dalian University of Technology*, 2010, **50**(5): 777-781 (韩冰, 韩敏. 基于动态 BP 算法的非线性滞后系统辨识. 大连理工大学学报, 2010, **50**(5): 777-781)
- Nascimento C F, Oliveria A A, Goedtel A, Silva I N, Serni P J A. Neural network-based approach for identification of the harmonic content of a nonlinear load in a single-phase system. *IEEE Latin America Transactions*, 2010, **8**(1): 65-73
- Zhao H Q, Zeng X P, He Z Y. Low-complexity nonlinear adaptive filter based on a pipelined bilinear recurrent neural network. *IEEE Transactions on Neural Networks*, 2011, **22**(9): 1494-1507
- Barbounis T G, Theocharis J B. A locally recurrent fuzzy neural network with application to the wind speed prediction using spatial correlation. *Neurocomputing*, 2006, **70**(7-9): 1525-1542
- Park D C. Prediction of MPEG video traffic over ATM networks using dynamic bilinear recurrent neural network. *Applied Mathematics and Computation*, 2008, **205**(2): 648-657
- Ferrari S. Multiobjective algebraic synthesis of neural control systems by implicit model following. *IEEE Transactions on Neural Networks*, 2009, **20**(3): 406-419
- Park J H, Kim S H, Moon C J. Adaptive neural control for strict-feedback nonlinear systems without backstepping. *IEEE Transactions on Neural Networks*, 2009, **20**(7): 1204-1209
- Hou Z G, Gupta M M, Nikiforuk P N, Tan M, Cheng L. A recurrent neural network for hierarchical control of interconnected dynamic systems. *IEEE Transactions on Neural Networks*, 2007, **18**(2): 466-481
- Talebi H A, Khorasani K, Tafazoli S. A recurrent neural-network-based sensor and actuator fault detection and isolation for nonlinear systems with application to the satellite's attitude control subsystem. *IEEE Transactions on Neural Networks*, 2009, **20**(1): 45-60
- Cho H C, Knowles J, Fadali M S, Lee K S. Fault detection and isolation of induction motors using recurrent neural networks and dynamic Bayesian modeling. *IEEE Transactions on Control Systems Technology*, 2010, **18**(2): 430-437
- Mousapha A I, Selmic R R. Wireless sensor network modeling using modified recurrent neural networks: application to fault detection. *IEEE Transactions on Instrumentation and Measurement*, 2008, **57**(5): 981-988
- Tsoi A C, Back A D. Locally recurrent globally feedforward networks: a critical review of architectures. *IEEE Transactions on Neural Networks*, 1994, **5**(2): 229-239

- 14 Zeng H B, Xiao S P, Liu B. New stability criteria for recurrent neural networks with a time-varying delay. *International Journal of Automation and Computing*, 2011, **8**(1): 128–133
- 15 Shao H Y. Delay-dependent stability for recurrent neural networks with time-varying delays. *IEEE Transactions on Neural Networks*, 2008, **19**(9): 1647–1651
- 16 Wang L S, Zhang R, Zhang Z B, Xu Z B, Peng J G. Some characterizations of global exponential stability of a generic class of continuous-time recurrent neural networks. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, 2009, **39**(3): 763–772
- 17 Jin L, Nikiforuk P N, Gupta M M. Absolute stability conditions for discrete-time recurrent neural networks. *IEEE Transactions on Neural Networks*, 1994, **5**(6): 954–963
- 18 Patan K. Stability analysis and the stabilization of a class of discrete-time dynamic neural networks. *IEEE Transactions on Neural Networks*, 2007, **18**(3): 660–673
- 19 Namikawa J, Tani J. A model for learning to segment temporal sequences, utilizing a mixture of RNN experts together with adaptive variance. *Neural Networks*, 2008, **21**(10): 1466–1475
- 20 Rankovic V M, Nikolic I Z. Identification of nonlinear models with feed forward neural network and digital recurrent network. *FME Transactions*, 2008, **36**(2): 87–92
- 21 Juang C F, Lin C C. A recurrent self-organizing neural fuzzy inference network. *IEEE Transactions on Neural Networks*, 1999, **10**(4): 828–845
- 22 Patan K. *Artificial Neural Networks for the Modelling and Fault Diagnosis of Technical Processes*. Berlin: Springer, 2008. 78–112



孙 健 2009 年在重庆大学自动化专业获得工学学士学位。重庆大学自动化学院控制理论与控制工程专业博士研究生。主要研究方向为故障诊断, 多智能体, 复杂网络, 并行计算, 神经网络, 神经科学。本文通信作者。

E-mail: cqsunjian@gmail.com

(**SUN Jian** Received his bachelor

degree in automation from Chongqing University in 2009. Currently, he is a Ph.D. candidate in control theory and control engineering at Chongqing University. His research interest covers fault diagnosis, multi-agent, complex network, parallel computing, neural network, and neuroscience. Corresponding author of this paper.)



柴 毅 1982 年在国防科学技术大学电子技术系获工学学士学位。1994 年在重庆大学电子信息工程学院自动化系, 工业自动化专业获工学硕士学位。2001 年在重庆大学自动化学院控制理论与控制工程专业获工学博士学位。主要研究方向为信息处理, 融合与控制, 智能系统理论及其应用, 工业工程控制理论与技术,

计算机网络与系统控制。E-mail: cqchaiyi@gmail.com

(**CHAI Yi** Received his bachelor and master degrees from National University of Defense Technology and Chongqing University in 1982 and 1994, respectively. And he received his Ph. D. degree from Chongqing University in 2001. He is now a professor in control theory and control engineering. His research interest covers information fusion theory and control, fault diagnosis and prediction.)



李华锋 2007 在淮北师范大学数学与应用数学专业获得理学学士学位, 2009 年在重庆大学应用数学专业获得理学硕士学位。重庆大学自动化学院控制理论与控制工程专业博士研究生。主要研究方向为信息融合理论与应用, 图像处理, 神经网络。E-mail: lhfaieww@126.com

(**LI Hua-Feng** Received his bache-

lor and master degrees from Huaibei Normal University and Chongqing University in 2007 and 2009, respectively, all in mathematics and applied mathematics. Currently, he is a Ph. D. candidate in control theory and control engineering at Chongqing University. His research interest covers image processing and information fusion, neural network.)



朱智勤 2010 年在重庆大学电子信息工程专业获得工学学士学位。重庆大学自动化学院控制理论与控制工程专业博士研究生。主要研究方向为故障诊断, 神经网络。E-mail: 445303326@qq.com

(**ZHU Zhi-Qin** Received his bache-

lor degree in electronic and information engineering from Chongqing University in 2010. Currently, he is a Ph. D. candidate in control theory and control engineering at Chongqing University. His research interest covers fault diagnosis and neural network.)