

## 采用优化极限学习机的多变量混沌时间序列预测\*

高光勇<sup>1)2)</sup> 蒋国平<sup>1)†</sup>

1) (南京邮电大学控制与智能技术研究中心, 南京 210003)

2) (九江学院信息科学与技术学院, 九江 332005)

(2011年4月18日收到; 2011年7月6日收到修改稿)

基于优化极限学习机理论, 提出一种多变量混沌时间序列预测方法. 该方法利用复合混沌和混沌变尺度算法对极限学习机的模型参数进行搜索和优化, 以提高极限学习机的泛化性能; 然后利用优化后的极限学习机对 Rossler 耦合系统的多变量混沌时序进行一步和多步预测, 并且与同类算法进行了比较, 结果表明了该方法的有效性, 且算法具有较强的抗噪能力; 最后讨论了预测结果和隐层神经元数目的关系.

**关键词:** 极限学习机, 多变量时间序列, 混沌序列预测, 复合混沌优化

**PACS:** 05.45.Pq, 05.45.Tp

## 1 引言

随着混沌理论的发展, 对非线性时间序列的预测已成为国内外学者研究的前沿课题, 并已在电力、交通、金融和生物医学等诸多领域得到了广泛的应用.

目前利用 Takens 嵌入原理<sup>[1]</sup>开展的研究大都是针对单变量时间序列的预测<sup>[2-5]</sup>, 但是, 实际中的混沌系统往往受多种变量的影响, 这些多变量的时间序列之间存在相互耦合的关系, 并且比单变量时间序列包含了更多的系统信息. 因此, 针对多变量时间序列的非线性预测研究具有重要的理论和实际意义, 而多变量相空间重构理论的提出则为多变量时间序列的预测提供了一个很好的思路<sup>[6]</sup>. 目前关于多变量时间序列的预测研究相对较少, 文献<sup>[7]</sup>提出了一种基于最大 Lyapunov 指数<sup>[8]</sup>的多变量混沌时序预测方法, 通过选取多个邻近重构向量对混沌耦合系统的多变量混沌时序进行了一步和多步预测, 并利用平均绝对误差和正规化均方误差对预测值和实际值间的误差进行了评价, 试验结果表明该方法取得了较好的效果.

极限学习机 (extreme learning machine, ELM) 是最近由 Huang 等<sup>[9]</sup>依据摩尔 - 彭罗斯 (MP) 广

义逆矩阵理论<sup>[10]</sup>提出的一种机器学习新算法, 该算法仅通过一步计算即可解析求出学习网络的输出权值. 同神经网络和支持向量机相比, 极限学习机极大地提高了网络的泛化能力和学习速度. 但是, 极限学习机方法因随机给定左侧权值, 使得回归模型容易产生泛化能力与稳定性不理想等问题. 本文提出一种基于优化极限学习机的多变量混沌时间序列预测算法. 该算法定义一种复合混沌, 并利用混沌变尺度方法<sup>[11]</sup>和复合混沌对极限学习机进行优化处理, 以提高极限学习机的稳定性与泛化能力. 然后利用优化后的极限学习机对 Rossler 耦合系统的多变量混沌时序进行一步和多步预测, 并且与文献<sup>[7]</sup>的方法进行了比较, 结果表明了该算法的有效性.

## 2 多变量混沌时序的重构

对  $L$  个独立时间序列  $X_1, X_2, \dots, X_L$ , 其中  $X_l = (x_{l1}, x_{l2}, \dots, x_{lN})$ ,  $l = 1, 2, \dots, L$ , 像单变量时间序列 (即  $L = 1$ ) 一样, 进行多变量相空间重构<sup>[6]</sup>, 相空间中的相点可以表示为

$$V_n(m_1, \dots, m_i, \dots, m_L) \\ = (x_{1,n}, x_{1,n-\tau_1}, \dots, x_{1,n-(m_1-1)\tau_1};$$

\* 国家自然科学基金 (批准号: 60874091), 江苏省 '六大人才高峰' 高层次人才项目 (批准号: SJ209006), 高等学校博士点基金 (批准号: 20103223110003), 江苏省高校基础研究计划 (批准号: 08KJD510022), 江苏省自然科学基金 (批准号: BK2010526), 南京邮电大学引进人才项目 (批准号: NY209021) 和江苏省高校研究生科研创新计划 (批准号: CXZZ11\_0400) 资助的课题.

† E-mail: jianggp@njupt.edu.cn

$$\begin{aligned} &x_{2,n}, x_{2,n-\tau_2}, \dots, x_{2,n-(m_2-1)\tau_2}; \\ &\vdots \\ &x_{L,n}, x_{L,n-\tau_L}, \dots, x_{L,n-(m_L-1)\tau_L}, \end{aligned} \quad (1)$$

式中,  $n = H_0, H_0 + 1, \dots, N, H_0 = \max_{1 \leq l \leq L} ((m_l - 1)\tau_l + 1)$ ,  $\tau_l$  和  $m_l$  分别为第  $l$  个时间序列的延迟时间和嵌入维. 类似于 Takens 的延迟嵌入定理, 当总的嵌入维数  $m = m_1 + m_2 + \dots + m_L > 2d$  ( $d$  为吸引子的分形维数) 时, 存在映射  $F : R^m \rightarrow R^m$ , 使得

$$V_{n+1} = F(V_n), \quad (2)$$

或者写成等价形式

$$x_{i,n+1} = F(V_n), \quad i = 1, 2, \dots, L. \quad (3)$$

此时状态空间  $V_n \rightarrow V_{n+1}$  的演化反映了原未知动力系统的演化, 即重构后的动力学系统与原动力系统微分同胚<sup>[12]</sup>. 多变量重构后的每个相点不仅保留了与单变量重构时一致的演化关系, 同时由于参与相空间重构的耦合变量的引入, 每个相点所包含的系统信息量得到了增加, 从而使相点轨迹能够更好地逼近原动力学系统的演化规律.

本文采用与文献 [7] 相同的方法确定时间序列的延迟时间和嵌入维, 即利用互信息最小法得出延迟时间<sup>[6]</sup>, 最小嵌入维的选取采用虚邻域点法<sup>[13]</sup>.

### 3 多变量混沌时间序列预测算法

混沌运动具有遍历性、随机性、规律性的特点, 同模拟退火、遗传算法等随机搜索算法相比, 更适合用于学习模型的优化. 本文使用一种变尺度复合混沌优化方法对极限学习机进行优化以提高极限学习机的泛化能力, 然后使用优化后的极限学习机对多变量混沌时间序列进行预测.

#### 3.1 复合混沌

引入以下定义来描述提出的复合混沌 (composite chaos) 模型.

**定义** 设  $x_{n+1} = F(x_n) = f_0^{(I_n)}(x_n)$ ,  $I_n = B(\lfloor wf_1(y_n) \rfloor \bmod r) \oplus B(\lfloor ef_1(y_n) \rfloor \bmod t)$ ,  $f_0, f_1$  是两个离散混沌动力系统,  $I_n$  为对  $f_0$  混沌序列采样的间隔因子,  $B(\cdot)$  为取二进制位函数,  $\oplus$  表示异或运算,  $w, r, e$  和  $t$  为整数常量, 称  $x_{n+1} = F(x_n) = f_0^{(I_n)}(x_n)$  为  $f_0, f_1$  上的复合系统.

本文选取 logistic 混沌方程

$$x_{n+1} = f_1(x_n) = 4x_n(1 - x_n) \quad (4)$$

生成原始混沌序列, 利用混合光学双稳方程

$$y_{n+1} = f_0(y_n) = 4 \sin^2(y_n - 2.5) \quad (5)$$

产生随机间隔因子.

复合系统为

$$x_{n+1} = F(x_n) = f_0^{(I_n)}(x_n), \quad (6)$$

$$I_n = B(\lfloor 157y_n \rfloor \bmod 6) \oplus B(\lfloor 213y_n \rfloor \bmod 8). \quad (7)$$

试验中系统初值  $(x_0, y_0)$  设为  $(0.45, 0.15)$ .

一维映射的 Lyapunov 指数的计算公式<sup>[14]</sup> 为

$$\lambda = \lim_{n \rightarrow \infty} \frac{1}{n} \sum_{i=0}^{n-1} \ln \left| \frac{dF(x)}{dx} \right|_{x=x_i}. \quad (8)$$

计算得出提出的复合系统的 Lyapunov 指数  $\lambda = 1.5909$ , 正的 Lyapunov 指数意味着混沌, 所以提出的复合系统 (6) 是离散混沌系统.

接下来使用 Lyapunov 指数和近似熵对混沌序列的随机性和复杂度进行定量的分析. Lyapunov 指数表示混沌系统平均每次迭代所引起的分离指数, 体现了系统对初值微小变化的敏感性和运动轨迹的随机性. 近似熵 (approximate entropy) 是 Pincus<sup>[15]</sup> 为了解决混沌现象中求解熵的困难而提出的, 它是对非线性时间序列复杂度的一种非负的定量描述. 近似熵越大, 表明非线性的物理过程越复杂. 表 1 显示了本文提出的复合混沌与 Tent, logistic 及文献 [14] 提出的复合混沌的 Lyapunov 指数、近似熵的比较. 可以看出, 提出的复合混沌具有更强的随机性和更大的复杂度, 因而更适合用于模型参数的搜索和优化.

表 1 随机复杂度分析比较

混沌方程	Lyapunov 指数	近似熵
Tent	0.8434	0.6127
Logistic	0.6914	0.6430
文献 [14] 提出的复合混沌	0.8589	0.8564
本文提出的复合混沌	1.5909	1.6909

#### 3.2 极限学习机理论

传统前馈神经网络采用梯度下降的迭代算法去调整权重参数, 具有着明显的缺陷: 1) 学习速度缓慢, 从而计算时间代价增大; 2) 学习率难以确定且易陷入局部最小值; 3) 易出现过度训练, 引起泛化性能下降. 这些缺陷成为制约使用迭代算法的前馈神经网络的广泛应用的瓶颈. 针对这些问题, Huang 等<sup>[9]</sup> 依据摩尔 - 彭罗斯 (MP) 广义逆矩阵理论<sup>[10]</sup> 提出了极限学习机 (ELM) 算法, 该算法仅通过一步计算即可解析求出学习网络的输出权值, 同

迭代算法相比, 极限学习机极大地提高了网络的泛化能力和学习速度.

极限学习机的网络训练模型采用前向单隐层结构. 设  $m, M, n$  分别为网络输入层、隐含层和输出层的节点数,  $g(x)$  是隐层神经元的激活函数,  $b_i$  为阈值. 设有  $N$  个不同的样本  $(x_i, t_i)$ ,  $1 \leq i \leq N$ , 其中  $x_i = [x_{i1}, x_{i2}, \dots, x_{im}]^T \in R^m$ ,  $t_i = [t_{i1}, t_{i2}, \dots, t_{in}]^T \in R^n$ , 则极限学习机的网络训练模型如图 1 所示.

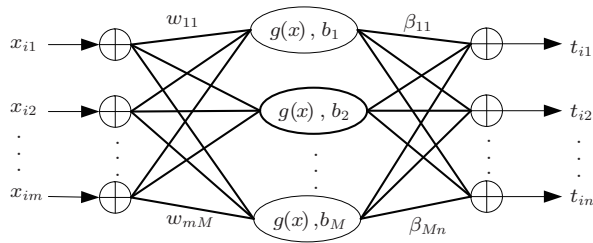


图 1 极限学习机的网络训练模型

极限学习机的网络模型可用数学表达式表示如下:

$$\sum_{i=1}^M \beta_i g(w_i \cdot x_i + b_i) = o_j, \quad j = 1, 2, \dots, N, \quad (9)$$

式中,  $w_i = [w_{1i}, w_{2i}, \dots, w_{mi}]$  表示连接网络输入层节点与第  $i$  个隐层节点的输入权值向量;  $\beta_i = [\beta_{i1}, \beta_{i2}, \dots, \beta_{in}]^T$  表示连接第  $i$  个隐层节点与网络输出层节点的输出权值向量;  $o_j = [o_{j1}, o_{j2}, \dots, o_{jn}]^T$  表示网络输出值.

极限学习机的代价函数  $E$  可表示为

$$E(S, \beta) = \sum_{j=1}^N \|o_j - t_j\|, \quad (10)$$

式中,  $S = (w_i, b_i, i = 1, 2, \dots, M)$ , 包含了网络输入权值及隐层节点阈值. Huang 等<sup>[9]</sup> 指出极限学习机的训练目标就是寻求最优的  $S, \beta$ , 使得网络输出值与对应实际值误差最小, 即  $\min(E(S, \beta))$ .  $\min(E(S, \beta))$  可进一步写为

$$\min E(S, \beta) = \min_{w_i, b_i, \beta} \|H(w_1, \dots, w_M, b_1, \dots, b_M, x_1, \dots, x_N)\beta - T\|, \quad (11)$$

式中,  $H$  表示网络关于样本的隐层输出矩阵,  $\beta$  表示输出权值矩阵,  $T$  表示样本集的目标值矩阵.  $H, \beta, T$  分别定义如下:

$$H(w_1, \dots, w_M, b_1, \dots, b_M, x_1, \dots, x_N)$$

$$= \begin{bmatrix} g(w_1 x_1 + b_1) & \cdots & g(w_M x_1 + b_M) \\ \vdots & & \vdots \\ g(w_1 x_N + b_1) & \cdots & g(w_M x_N + b_M) \end{bmatrix}_{N \times M}, \quad (12)$$

$$\beta = \begin{bmatrix} \beta_1^T \\ \vdots \\ \beta_M^T \end{bmatrix}_{M \times n}, \quad T = \begin{bmatrix} t_1^T \\ \vdots \\ t_N^T \end{bmatrix}_{N \times n}. \quad (13)$$

极限学习机的网络训练过程可归结为一个非线性优化问题, (11) 式为目标函数. 当网络隐层节点的激活函数无限可微时, 网络的输入权值和隐层节点阈值可随机赋值, 此时矩阵  $H$  为一常数矩阵, 极限学习机的学习过程可等价于求取线性系统  $H\beta = T$  最小范数的最小二乘解  $\hat{\beta}$ , 其计算式为

$$\hat{\beta} = H^\dagger T \quad (14)$$

式中  $H^\dagger$  是矩阵  $H$  的 MP 广义逆.  $\hat{\beta}$  求解后, 极限学习机的网络训练过程也就完成了.

### 3.3 极限学习机的优化

极限学习机随机地选择输入权值和隐层节点阈值, 并且输出权值是在确定的输入权值和隐层节点阈值的基础上求得, 因而可能存在一系列并非最佳的输入权值和隐层节点阈值, 导致极限学习机的预测精度受到影响. 针对该问题, 本文提出了变尺度复合混沌优化的极限学习机预测模型 (CC-ELM), 即极限学习机网络的最佳输入权值及隐层节点阈值  $S$  通过变尺度复合混沌搜索调整得到, 输出权值  $\beta$  则通过解析 MP 广义逆求得.

在 CC-ELM 中, (11) 式可简化为

$$\min E(S) = \sum_{j=1}^N \|o_j - t_j\|. \quad (15)$$

参数  $S = (w_i, b_i, i = 1, 2, \dots, M)$  简记为  $S = (z_i, i = 1, 2, \dots, L)$ ,  $z_i$  对应于 CC-ELM 的输入权值及隐层节点阈值, 且  $a_i \leq z_i \leq b_i$ . 优化的目标就是获得最佳的  $z_i$ , 从而得到  $\min E$ . 然而, 对神经网络训练来说, 仅仅采用拟合性作为网络训练好坏的唯一标准并不合适. 一个具有相对较小拟合误差的训练网络用于测试集上的误差不一定就较小<sup>[16]</sup>. Bartlett<sup>[17]</sup> 指出, 在拟合性差不多的情况下, 具有较小输出权值的网络将会有较好的泛化性能. 因此, 我们增加输出权值范数  $\|\beta\|$  作为网络选择的辅助标准. 当优化过程中相邻两次拟合误差的差别很小时, 选择  $\|\beta\|$  较小的  $E$ .

CC-ELM 模型的具体优化步骤描述如下:

**步骤 1** 对变量初始化. 设  $k = 0, r = 0, x_i^k = x_i(0), x_i^* = x_i(0), a_i^r = a_i, b_i^r = b_i, i = 1, 2, \dots, L$ , 这里  $x_i(0)$  是复合混沌系统 (6) 的初始值, 最佳目标函数值  $E^*$  和最佳输出权值  $\beta^*$  被初始化为较大的正值.

**步骤 2**  $x_i^k$  被映射到  $z_i$  的定义域, 并记为

$$z_i^k = a_i^r + (b_i^r - a_i^r)x_i^k. \quad (16)$$

**步骤 3** 对于给定的训练样本, 由 (12),(14) 式解得  $\beta_{z_i}$ , 由 (9),(15) 式求得  $E(z_i^k), E^*$  和  $\beta^*$  按下式赋值, 式中  $\varepsilon$  是预定义的阈值 (一个较小的正数):

$$\begin{aligned} E^* &= E(z_i^k), \quad \beta^* = \beta_{z_i}, \quad x_i^* = x_i^k, \\ E^* - E(z_i^k) &> \varepsilon; \\ E^* &= E(z_i^k), \quad \beta^* = \beta_{z_i}, \quad x_i^* = x_i^k, \\ |E^* - E(z_i^k)| &< \varepsilon, \quad \|\beta_{z_i}\| < \|\beta^*\|; \\ E^* &= E^*, \quad \beta^* = \beta^*, \quad \text{其他}. \end{aligned} \quad (17)$$

**步骤 4**

$$k := k + 1, x_i^k := F(x_i^{k-1}) = f_0^{(T_i^{k-1})}(x_i^{k-1}). \quad (18)$$

**步骤 5** 重复执行步骤2—4, 直到  $E^*$  在  $T$  次循环都保持不变, 或者至多循环  $R$  次 ( $R > T$ ), 然后跳到步骤 6.

**步骤 6** 减小  $z_i$  的搜索区域, 即

$$a_i^{r+1} = z_i^* - \Gamma(b_i^r - a_i^r), \quad b_i^{r+1} = z_i^* + \Gamma(b_i^r - a_i^r), \quad (19)$$

式中  $\Gamma \in (0, 0.5), z_i^* = a_i^r + x_i^*(b_i^r - a_i^r), z_i^*$  为当前最佳解.

**步骤 7**  $x_i^*$  和  $x_i^k$  被重新赋值,

$$\begin{aligned} x_i^* &= (z_i^* - a_i^{r+1}) / (b_i^{r+1} - a_i^{r+1}), \\ x_i^k &:= (1 - \alpha)x_i^* + \alpha x_i^k, \end{aligned} \quad (20)$$

式中  $\alpha$  为一个小的正数, 例如 0.1.

**步骤 8**  $r := r + 1$ . 如果  $r > P$  或者  $E^* \leq Q$ , 则结束优化, 此时获得具有  $\min E$  的网络. 否则,  $\alpha := \alpha - 0.01$ , 然后跳到步骤 2 继续执行. 这里,  $P$  是指定的  $r$  的最大值,  $Q$  为可接受的  $E$  值.

### 3.4 预测算法

基于 CC-ELM 的多变量混沌时间序列预测算法的流程显示如下:

1) 对多变量混沌时序进行归一化处理, 并确定时序重构的嵌入维和延迟时间 ( $m_l/\tau_l$ ).

2) 按照 (1) 式进行多变量时序的相空间重构, 确定训练集和测试集.

3) 利用训练集对 CC-ELM 模型进行训练, 得到最佳参数.

4) 使用训练好的 CC-ELM 模型对测试集进行一步或多步预测, CC-ELM 的输出端同时输出多个变量的预测结果.

5) 对预测结果作反归一化处理, 计算预测误差.

### 3.5 计算复杂度分析

预测算法的计算量主要集中在多变量时序的相空间重构和 CC-ELM 模型的优化, 因而针对这两个步骤进行计算复杂度分析.

多变量时序的相空间重构有两重循环组成, 外循环次数为相空间相点数  $K$ , 内循环由  $L$  个子循环构成, 各子循环的循环次数分别为  $m_1, m_2, \dots, m_L, L$  为变量时序的个数,  $m_l$  为第  $l$  个时序的嵌入维, 总嵌入维数  $m = m_1 + m_2 + \dots + m_L$ , 因而相空间重构的计算复杂度为  $O(K(m_1 + m_2 + \dots + m_L)) = O(Km)$ . 在 CC-ELM 模型的优化步骤中, 第一步变量初始化计算复杂度为  $O(1)$ ; 第二步至第八步为外层循环, 循环次数为  $P$ ; 第二步至第五步为内层循环, 最大循环次数为  $R$ , 计算复杂度为  $O(RMN)$ , 其中,  $M, N$  分别为隐层节点数和训练样本数; 第六步至第八步计算复杂度为  $O(1)$ , 因此优化过程总的计算复杂度为  $O(1) + O(P(RMN + 1)) = O(PRMN)$ .

## 4 仿真试验

作为混沌序列预测实验用例的基准数据集包括 Lorenz, Henon, Rossler 及 MG 等混沌模型 [2-5]. 两个不同类型 Rossler 混沌方程经过耦合后得到的系统具有比 Rossler, Lorenz 等基准混沌模型具有更强的复杂性; 同时, 为了方便比较, 论文使用文献 [7] 中采用的两个耦合 Rossler 混沌系统进行预测实验. 评价体系采用平均绝对误差  $e_{AAE}$  和正规化均方误差  $e_{NMSE}$ ,

$$e_{AAE} = \frac{1}{N} \sum_{n=1}^N |\hat{x}_{l,n} - x_{l,n}|, \quad (21)$$

$$e_{NMSE} = \frac{\frac{1}{N} \sum_{n=1}^N (\hat{x}_{l,n} - x_{l,n})^2}{\sigma^2}, \quad (22)$$

式中  $\hat{x}_{l,n}$  和  $x_{l,n}$  为预测值和实际值,  $\sigma^2$  是  $x_{l,n}$  的方差.

### 4.1 实验模型

系统 1 选择两个不同的 Rossler 方程的耦合系统.

$$\begin{aligned} \dot{x}_1 &= -\omega_1 y_1 - z_1 + \varepsilon(x_2 - x_1), \\ \dot{y}_1 &= \omega_1 x_1 + 0.15y_1, \\ \dot{z}_1 &= 0.2 + z_1(x_1 - 10), \end{aligned}$$

$$\begin{aligned} \dot{x}_2 &= -\omega_2 y_2 - z_2 + \varepsilon(x_1 - x_2), \\ \dot{y}_2 &= \omega_2 x_2 + 0.15y_2, \\ \dot{z}_2 &= 0.2 + z_2(x_2 - 10), \end{aligned} \tag{23}$$

其中  $\omega_1 = 0.99, \omega_2 = 0.95$ , 系统初始环境为  $x_{1,0} = 0.1, y_{1,0} = 0.2, z_{1,0} = 0.3, x_{2,0} = 0, y_{2,0} = 0, z_{2,0} = 15$ , 耦合参数  $\varepsilon=0.05$ .

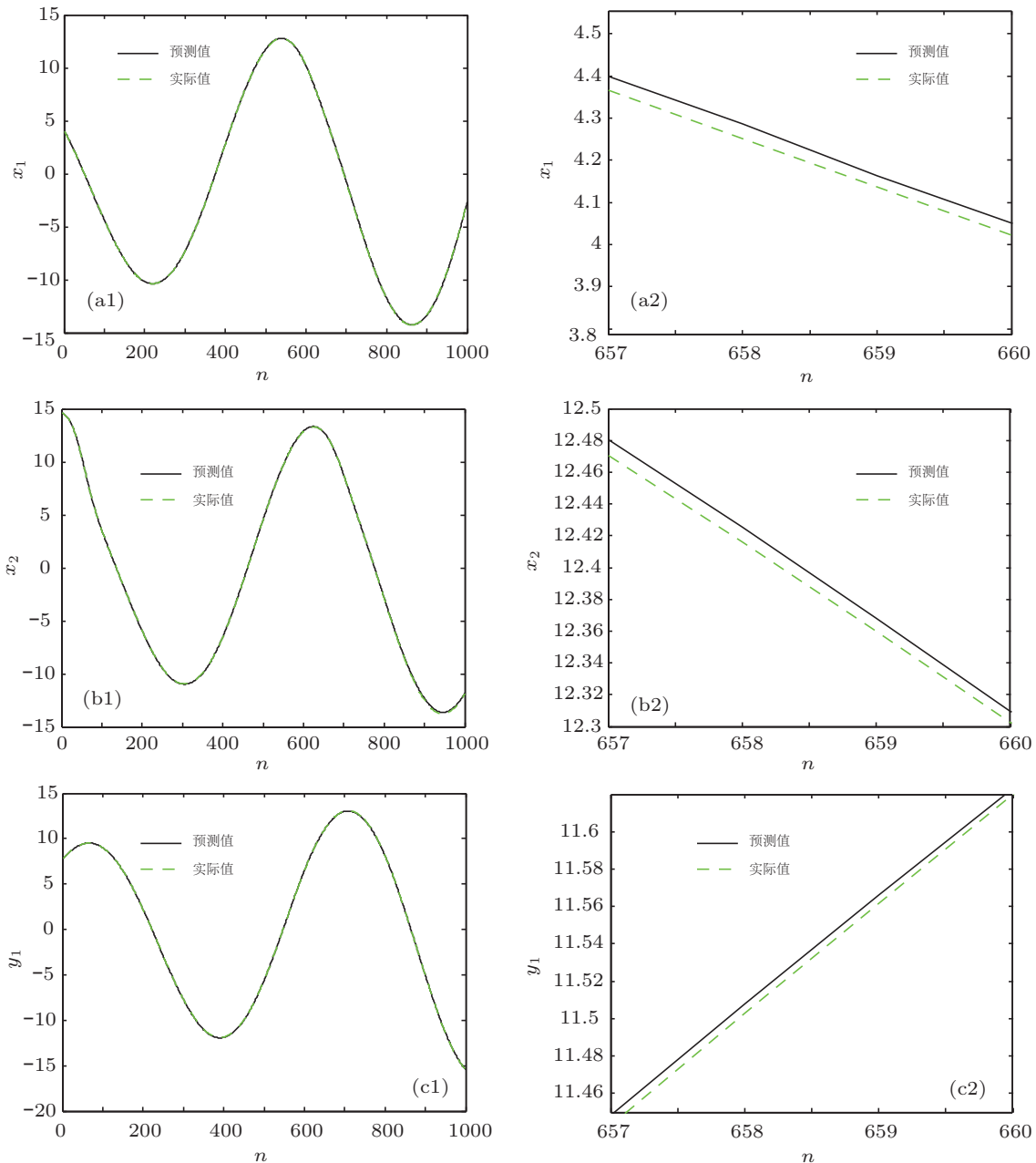


图 2 系统 1 的一步预测值与实际值的对比图 (a1) 预测变量  $x_1$ ; (a2) (a1) 的局部放大; (b1) 预测变量  $x_2$ ; (b2) (b1) 的局部放大; (c1) 预测变量  $y_1$ ; (c2) (c1) 的局部放大

### 系统 2

$$\begin{aligned} \dot{x}_1 &= -\omega_1 y_1 - z_1 + \varepsilon(x_2 - x_1), \\ \dot{y}_1 &= \omega_1 x_1 + 0.15y_1, \end{aligned}$$

$$\begin{aligned} \dot{z}_1 &= 0.2 + z_1(x_1 - 10), \\ \dot{x}_2 &= \omega_2 + 0.25x_2 + z_2 + \varepsilon(x_1 - x_2), \\ \dot{y}_2 &= 3 + y_2\omega_2, \end{aligned}$$

$$\begin{aligned} \dot{z}_2 &= -0.5y_2 + 0.05z_2, \\ \dot{\omega}_2 &= -x_2 - y_2, \end{aligned} \quad (24)$$

其中  $\omega_1 = 0.925$ , 系统初始环境为  $x_{1,0} = 0.1$ ,  $y_{1,0} = 0.2$ ,  $z_{1,0} = 0.3$ ,  $x_{2,0} = 0$ ,  $y_{2,0} = 0$ ,  $z_{2,0} = 15$ ,  $\omega_{2,0} = -20$ , 耦合参数  $\varepsilon = 0.008$ .

在预测实验中, 先由两个系统产生步长为 17000 的各变量时间序列, 然后去掉前面 10000 步长的暂态点. 系统求解方法采用四阶龙格-库塔积分法, 积分步长为  $\Delta h = 0.01$ , 预测时由前 5000 步的时间序列重构向量为参考向量, 对后 1000 步的变量值进行预测.

### 4.2 一步预测实验

首先分别对系统 1 的变量  $x_1$  作单变量时间序列重构, 对变量  $x_1, x_2$  进行两变量时间序列重构, 对变量  $x_1, x_2, y_1$  进行三变量时间序列重构, 对变量  $x_1, x_2, y_1, y_2$  进行四变量时间序列重构. 然后采用提出的预测算法进行一步预测实验, 预测结果如表 2 所示. 从表 2 可以看出, 采用本文方法的各变量一步预测的平均绝对误差  $e_{AAE}$  和正规化均方误差  $e_{NMSE}$  均较小, 并且预测结果优于文献 [7] 的方法. 图 2 显示了重构变量取  $x_1, x_2, y_1$  时的一步预测值与实际值的对比图.

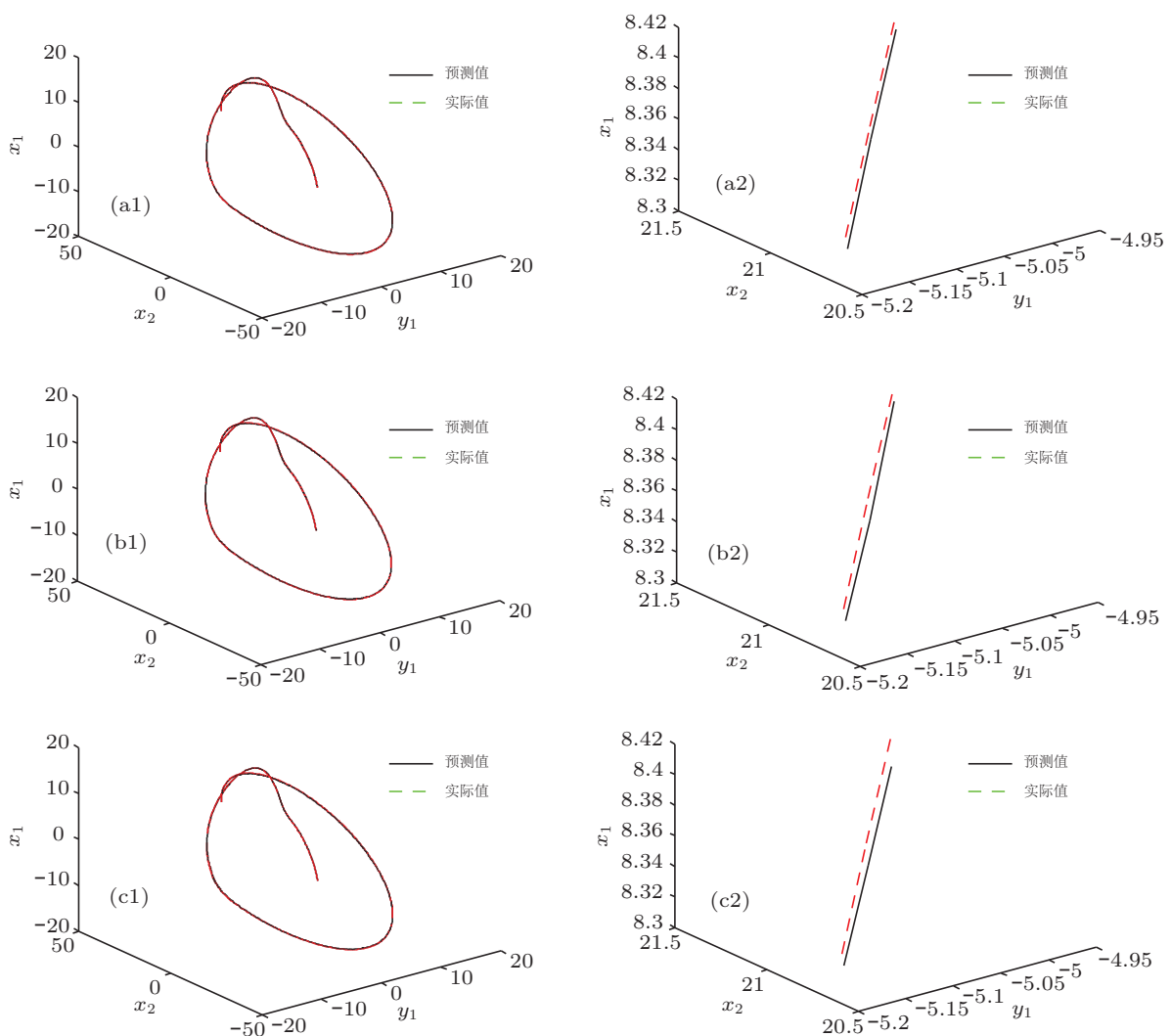


图 3 系统 2 的多步预测结果对比图 (a1) 1 步预测; (a2) (a1) 的局部放大; (b1) 2 步预测; (b2) (b1) 的局部放大; (c1) 3 步预测; (c2) (c1) 的局部放大

表 2 系统 1 的一步预测结果 (隐层节点数  $J = 20$ )

重构变量	重构参数 ( $m_l/\tau_l$ )	$e_{\text{AAE}}$		$e_{\text{NMSE}}$	
		本文算法	文献 [7]	本文算法	文献 [7]
$x_1$	6/17	0.0009	0.0198	$0.1899 \times 10^{-7}$	$0.5354 \times 10^{-5}$
$x_1$	4/17	0.0055	0.0488	$0.6031 \times 10^{-6}$	$0.1241 \times 10^{-3}$
$x_2$	3/14	0.0064	0.0706	$0.8767 \times 10^{-6}$	$0.2675 \times 10^{-3}$
$x_1$	1/17	0.0384	0.0497	$0.3022 \times 10^{-4}$	$0.0716 \times 10^{-3}$
$x_2$	4/14	0.0394	0.0762	$0.4315 \times 10^{-4}$	$0.1951 \times 10^{-3}$
$y_1$	2/11	0.0293	0.0384	$0.1717 \times 10^{-4}$	$0.0564 \times 10^{-3}$
$x_1$	1/17	0.0433	0.0662	$0.4596 \times 10^{-4}$	$0.1575 \times 10^{-3}$
$x_2$	4/14	0.0357	0.0709	$0.2497 \times 10^{-4}$	$0.1712 \times 10^{-3}$
$y_1$	2/11	0.0338	0.0562	$0.2958 \times 10^{-4}$	$0.1215 \times 10^{-3}$
$y_2$	1/16	0.0241	0.0600	$0.1506 \times 10^{-4}$	$0.1008 \times 10^{-3}$

### 4.3 多步预测实验

对系统 2 的变量  $x_1, x_2, \omega_2, y_1$  进行四变量时间序列重构后, 采用提出的预测算法进行步长  $s = 1, 2, 3$  步预测. 表 3 显示了预测结果以及

与文献 [7] 的比较. 可以看出, 即使在步长为 3 的情况下, 仍获得较好的预测结果, 并且优于文献 [7] 的方法. 图 3 直观地展现了步长  $s = 1, 2, 3$  时变量  $x_1, x_2, y_1$  的预测值与实际值的对比效果.

表 3 系统 2 的多步预测结果 (隐层节点数  $J = 50$ )

重构变量	重构参数 ( $m_l/\tau_l$ )	$s$	$e_{\text{AAE}}$		$e_{\text{NMSE}}$	
			本文算法	文献 [7]	本文算法	文献 [7]
$x_1$	3/15	1	0.0196	0.0775	$0.1090 \times 10^{-4}$	$0.4749 \times 10^{-3}$
$x_2$	5/12		0.0465	0.1522	$0.1070 \times 10^{-4}$	$0.8856 \times 10^{-3}$
$\omega_2$	1/15		0.0398	0.0432	$0.4892 \times 10^{-4}$	$0.0506 \times 10^{-3}$
$y_1$	1/15		0.0214	0.0610	$0.1122 \times 10^{-4}$	$0.2344 \times 10^{-3}$
$x_1$	3/15	2	0.0298	0.1546	$0.0256 \times 10^{-3}$	0.0019
$x_2$	5/12		0.0658	0.3029	$0.0240 \times 10^{-3}$	0.0035
$\omega_2$	1/15		0.0957	0.0849	$0.2145 \times 10^{-3}$	0.0002
$y_1$	1/15		0.0327	0.1211	$0.0294 \times 10^{-3}$	0.0009
$x_1$	3/15	3	0.0406	0.2312	$0.0466 \times 10^{-3}$	0.0042
$x_2$	5/12		0.0822	0.4520	$0.0405 \times 10^{-3}$	0.0078
$\omega_2$	1/15		0.1161	0.1257	$0.3833 \times 10^{-3}$	0.0004
$y_1$	1/15		0.0445	0.1806	$0.0576 \times 10^{-3}$	0.0020

表 4 系统 2 时间序列加入白噪声后的预测结果 (隐层节点数  $J = 30$ )

重构变量	重构参数 ( $m_l/\tau_l$ )	噪声分布	$e_{\text{AAE}}$		$e_{\text{NMSE}}$	
			本文算法	文献 [7]	本文算法	文献 [7]
$x_1$	5/15	无	0.0066	0.0786	$0.0855 \times 10^{-5}$	$0.1851 \times 10^{-3}$
$x_2$	3/12	无	0.0373	0.0841	$0.4133 \times 10^{-5}$	$0.2185 \times 10^{-3}$
$x_1$	5/15	$N(0, 1 \times 10^{-3})$	0.0408	0.1001	$0.3536 \times 10^{-4}$	$0.2797 \times 10^{-3}$
$x_2$	3/12	$N(0, 1 \times 10^{-3})$	0.0960	0.1111	$0.2992 \times 10^{-4}$	$0.3484 \times 10^{-3}$
$x_1$	5/15	$N(0, 1 \times 10^{-2})$	0.1098	0.2080	$0.2553 \times 10^{-3}$	0.0012
$x_2$	3/12	$N(0, 1 \times 10^{-2})$	0.1138	0.1154	$0.0461 \times 10^{-3}$	0.0013
$x_1$	5/15	$N(0, 1 \times 10^{-1})$	0.3334	0.5998	0.0024	0.0097
$x_2$	3/12	$N(0, 1 \times 10^{-1})$	0.3736	0.6573	0.0004	0.0115

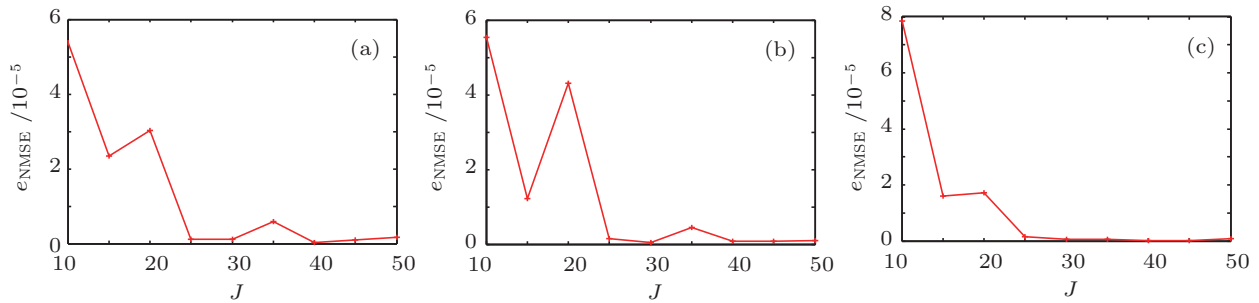


图4 系统1  $x_1, x_2, y_1$  预测值的正规化均方误差  $e_{\text{NMSE}}$  与  $J$  值的关系 (a) 预测变量  $x_1$ ; (b) 预测变量  $x_2$ ; (c) 预测变量  $y_1$

#### 4.4 噪声影响实验

为了分析噪声对本文预测方法的影响, 在加噪的情况下选取系统2的  $x_1(n), x_2(n)$  两个变量进行预测, 表4显示了预测结果以及与文献[7]的比较. 可以看出, 预测的  $e_{\text{AAE}}$  和  $e_{\text{NMSE}}$  随着加入噪声的方差的增加而增大, 但是仍获得较好的预测结果, 并且优于文献[7]的方法, 说明本文预测方法具有较强的抗噪能力.

#### 4.5 隐层节点数 $J$ 选择实验

隐含层神经元的数目  $J$  对极限学习机的训练和预测结果有一定的影响.  $J$  过少将不能很好地拟合出实际的混沌序列轨迹, 但  $J$  也不是越大越好, 随着  $J$  的增大, 算法的时间成本将增加, 而且  $J$  过大时还会出现过拟合现象. 图4显示了  $J$  值与系

统1的  $x_1, x_2, y_1$  三个重构变量预测值的正规化均方误差  $e_{\text{NMSE}}$  的关系. 可以看出, 当  $J = 40$  时,  $x_1, x_2, y_1$  的预测值均获得较小的  $e_{\text{NMSE}}$ , 随着  $J$  的增大,  $e_{\text{NMSE}}$  呈上升趋势.

## 5 结论

本文提出了一种多变量混沌时间序列预测方法, 该方法具有以下特点: 1) 相比与其他类型混沌系统, 定义的复合混沌具有更强的随机复杂度, 更适合用于模型参数的搜索和优化; 2) 采用复合混沌的变尺度优化算法对极限学习机进行优化, 优化后的极限学习机模型(CC-ELM) 具有更好的泛化性能; 3) 利用CC-ELM对耦合混沌系统的多变量同时进行了一步和多步预测, 试验结果表明与同类方法相比, 该方法具有更高的预测精度.

- [1] Takens F 1981 *In Lecture notes in mathematics*, Vol.898 *Dynamical systems and turbulence*(Berlin:Springer)p366
- [2] Yan H, Wei P, Xiao X C 2009 *Chin. Phys. B* **18** 3287
- [3] Samanta B 2011 *Expert Syst. with Appl.* **38** 11406
- [4] Zhang C T, Ma Q L, Peng H 2010 *Acta Phys. Sin.* **59** 7623 (in Chinese) [张春涛, 马千里, 彭宏 2010 物理学报 **59** 7623]
- [5] Fang F, Wang H Y, Yang Z M 2011 *Appl. Mech. Mater.* **47** 3180
- [6] Cao L Y, Mees A, Judd K 1998 *Phys. D* **121** 75
- [7] Zhang Y, Guang W 2009 *Acta Phys. Sin.* **58** 0756 (in Chinese) [张勇, 关伟 2009 物理学报 **58** 0756]
- [8] Lu S, Wang H Y 2006 *Acta Phys.Sin.* **55** 572 (in Chinese) [卢山, 王海燕 2006 物理学报 **55** 572]
- [9] Huang G B, Zhu Q Y, Siew C K 2006 *Neuro Computing* **70** 489
- [10] Serre D 2002 *Matrices:Theory and Applications* (New York: Springer) p145
- [11] Zhang T, Wang H W, Wang Z C 1999 *Control and Decision* **14** 285 (in Chinese) [张彤, 王宏伟, 王子才 1999 控制与决策 **14** 285]
- [12] Sauer T, Yorke J A, Casdagli M 1991 *J. Stat. Phys.* **65** 579
- [13] Wang H Y, Sheng Z H, Zhang J 2003 *J. South. Univ. (Natural Science Edition)* **33** 115 (in Chinese) [王海燕, 盛昭瀚, 张进 2003 东南大学学报(自然科学版) **33** 115]
- [14] Tong X J, Cui M G 2009 *Science in China F* **39** 588 (in Chinese) [佟晓筠, 崔明根 2009 中国科学(F辑) **39** 588]
- [15] Pincus S 1995 *Chaos* **5** 110
- [16] Zhu Q Y, Qin A K, Suganthan P N, Huang G B 2005 *Pattern Recognition* **38** 1759
- [17] Bartlett P L 1998 *IEEE Trans. Inform. Theory* **44** 525



# Prediction of multivariable chaotic time series using optimized extreme learning machine\*

Gao Guang-Yong<sup>1)2)</sup> Jiang Guo-Ping<sup>1)†</sup>

1) (*Center for Control & Intelligence Technology, Nanjing University of Posts and Telecommunications, Nanjing 210003, China*)

2) (*School of Information Science & Technology Jiujiang University, Jiujiang 332005, China*)

(Received 18 April 2011; revised manuscript received 6 July 2011)

## Abstract

A prediction algorithm of multivariable chaotic time series is proposed based on optimized extreme learning machine (ELM). In this algorithm, a presented composite chaos system and mutative scale chaos method are utilized first to search and optimize the parameters of ELM for improving the generalization performance. Then the optimized ELM is used to predict the multivariable chaotic time series of Rossler coupled system for single step and multi-step, and the scheme is compared with the congeneric method, which shows the validity and stronger ability against noise of the developed algorithm. Finally, the relation between prediction result and number of hidden neurons is discussed.

**Keywords:** extreme learning machine, multivariable chaotic time series, prediction of chaotic time series, composite chaos optimization

**PACS:** 05.45.Pq, 05.45.Tp

---

\* Project supported by the National Natural Science Foundation of China (Grant No. 60874091), the Six Projects Sponsoring Talent Summits of Jiangsu Province, China (Grant No. SJ209006), the Research Fund for the Doctoral Program of Higher Education of China (Grant No. 20103223110003), the Natural Science Basic Research Project for Universities of Jiangsu Province, China (Grant No. 08KJD510022), the Natural Science Foundation of Jiangsu Province, China (Grant No. BK2010526), the Project for Introduced Talent in Nanjing University of Posts and Telecommunications, China (Grant No. NY209021), and the Scientific Research Innovation Program for the Graduat Students in Jiangsu Province, China (Grant No. CXZZ11\_0400).

† E-mail: jianggp@njupt.edu.cn