

基于飞参数据的飞机操纵系统故障评估方法

吴祯涛¹, 李学仁², 杜军², 丁超¹

(1. 空军工程大学 研究生院, 西安 710038; 2. 空军工程大学 航空工程学院, 西安 710038)

摘要: 针对某型飞机的操纵系统故障评估问题, 提出了一种基于飞参数据建立的差分进化极限学习机 (DE-ELM) 算法; 该算法融合了差分进化 (DE) 和极限学习机 (ELM) 两种算法, 通过对飞参数据进行训练, 构建了飞机操纵系统的黑箱模型; 由于极限学习机 (ELM) 的输入权值以及隐含层阈值是随机产生的, 所以 ELM 的随机性较大, 稳定性不高, 故利用寻优能力较强的 DE 对 ELM 输入权值和隐含层阈值进行寻优, 从而实现 ELM 的结构优化, 提升 ELM 的稳定性和鲁棒性; 仿真结果表明, DE-ELM 算法的决定系数达到了 97.6%, 其均方误差相比于 BP 神经网络降低了约 79%, 相比于单纯的 ELM 降低了 64%; 所以说该法可以有效提高精确度, 同时具有更加良好的泛化性能。

关键词: 飞参数据; 差分进化; 极限学习机; 故障评估

Fault Evaluation Method of Aircraft Control System Based on Flight Data

Wu Zhentao¹, Li Xueren², Du Jun², Ding Chao¹

(1. Graduate College Air Force Engineering University, Xi'an 710038, China;

2. Aeronautics Engineering College Air Force Engineering University, Xi'an 710038, China)

Abstract: A differential evolution extreme learning machine (DE-ELM) algorithm based on flight data is proposed to solve the problem of fault evaluation of an aircraft's control system. The algorithm combines differential evolution (DE) and extreme learning machine (ELM), by training the flight data, a black box model of aircraft control system is constructed. Because the input weights and hidden layer thresholds of the ELM are generated randomly, the randomness of ELM is large and the stability of ELM is not high. Therefore, the input weights and hidden layer thresholds of ELM are optimized by DE, which has strong optimization ability, so that the structure of ELM can be optimized and the stability and robustness of ELM can also be improved. The simulation results show that the decisive coefficient of DE-ELM reaches 97.6%, and its mean square error is reduced by 79% compared with BP neural network and 64% compared with ELM. Therefore, this method can effectively improve the accuracy and has better generalization performance.

Keywords: flight data; differential evolution; extreme learning machine; fault evaluation

0 引言

飞机的操纵系统是飞机的核心系统之一, 飞行员通过操纵系统实现对飞机的控制, 从而完成飞行任务。一旦操纵系统出现问题, 飞行员就无法对飞机实现有效的控制, 那么就会带来重大的经济损失以及人员伤亡。所以说, 针对飞机操纵系统建立一种行之有效的故障评估方法, 是提高飞机可靠性、安全性的一种必要措施。

飞参数据反映了飞机各个部分的状态, 对飞参数据进行分析即可了解飞机的状态, 是否存在故障等^[1]。在数据分析方面有很多方法, 最早是利用 BP 神经网络实现对数据的预测, 但是该方法容易陷入局部最小并且泛化性能较差, 同时 BP 神经网络的初始权值和阈值的选取对最后的性能影

响比较大^[2], 由此产生了使用粒子群算法或遗传算法对其进行优化, 文献 [3]、文献 [4] 和文献 [5] 采用了粒子群算法进行优化, 文献 [6]、文献 [7] 和文献 [8] 采用了遗传算法。优化的目的是为了选取更优的训练参数, 但是对于神经网络本身的所具有的缺点来讲, 改进的效果并不明显。之后有人提出使用支持向量机这一方法, 该方法克服了 BP 神经网络容易陷入局部最小以及泛化能力差的缺点, 文献 [9] 就采用了最小二乘支持向量机实现了对大气数据计算机系统的故障检测, 但是该方法在解决数据样本较大的问题时所耗时间太长, 不能满足要求。

极限学习机 (Extreme Learning Machine, ELM) 是近段时间提出来的一种新方法^[10], 它是一种单隐含层前馈神经网络 (single hidden-layer feedforward neural network, SLFN), 被广泛应用于分类、回归、聚类等等问题中。由于其输入权值和隐含层阈值均是随机产生的, 所以 ELM 的稳定性较差, 预测结果波动较大, 故提出了一种差分进化 (Differential Evolution, DE) 的方法^[11]。通过 DE 先对 ELM 的输入权值以及隐含层阈值进行寻优, 再将寻优之后的输入权值和隐含层阈值赋给 ELM 进行训练, 经过 DE 的寻优之后, 优化了 ELM 的算法结构, 克服了其随机性强的

收稿日期: 2019-02-14; 修回日期: 2019-02-22。

基金项目: 国家自然科学基金 (11447174)、陕西省自然科学基金基础研究计划 (2015JQ5155) 资助项目。

作者简介: 吴祯涛 (1995-), 男, 河南新乡人, 硕士研究生, 主要从事智能数据处理方向的研究。

李学仁 (1963-), 男, 山西运城人, 教授, 主要从事智能检测与健康状态监控, 复杂系统动力学控制理论与应用等方向的研究。

缺点,提高了算法的稳定性^[12-13]。

1 差分进化极限学习机

1.1 极限学习机

极限学习机是一种典型的单隐层前馈神经网络,图 1 为典型的单隐层前馈神经网络模型。

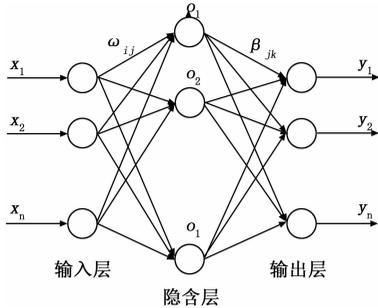


图 1 单隐层前馈神经网络模型

其中,输入层节点数为 n ,输出层节点数为 m ,隐层节点数为 l 。 ω_{ij} 为输入层第 i 个神经元与隐层第 j 个神经元间的连接权值, β_{jk} 为输入层第 j 个神经元与隐层第 k 个神经元间的连接权值。设隐层神经元的激活函数为 $g(x)$,则极限学习机模型可以用公式 (1) 表示:

$$\sum_{i=1}^l \beta_i g_i(x_j) = \sum_{i=1}^l \beta_i g(\omega_i \cdot x_j + b_i) = y_j \quad (1)$$

式中, $j = 1, 2, 3, \dots; x_j = [x_{j1}, x_{j2}, \dots, x_{jm}]$ 表示网络的输入值; $y_j = [y_{j1}, y_{j2}, \dots, y_{jm}]$ 表示网络的输出值。

将公式 (1) 改写成矩阵形式:

$$H\beta = Y \quad (2)$$

其中:

$$H(\omega_1, \omega_2, \dots, \omega_l; b_1, b_2, \dots, b_l; x_1, x_2, \dots, x_n) = \begin{pmatrix} g(\omega_1 x_1 + b_1) & g(\omega_2 x_1 + b_2) & \dots & g(\omega_l x_1 + b_l) \\ g(\omega_1 x_2 + b_1) & g(\omega_2 x_2 + b_2) & \dots & g(\omega_l x_2 + b_l) \\ \vdots & \vdots & \ddots & \vdots \\ g(\omega_1 x_n + b_1) & g(\omega_2 x_n + b_2) & \dots & g(\omega_l x_n + b_l) \end{pmatrix}_{n \times l}$$

$$Y = [Y_1^T, Y_2^T, \dots, Y_m^T]_{m \times l}^T$$

根据 Huang 所提出的定理可知,当激活函数 $g(x)$ 无限可微时,SLFN 的参数并不需要全部进行调整, ω 和 b 在训练前可以随机选择,且在训练过程中保持不变^[14]。而隐层的连接权值 β 可以通过求解以下方程组的最小二乘获得:

$$\min_{\beta} \| H\beta - T \| \quad (3)$$

其解为:

$$\hat{\beta} = H^+ T \quad (4)$$

其中: H^+ 为隐层输出矩阵 H 的 Moore-Penrose 广义逆。

综合上述,可以知道 ELM 算法主要分为以下步骤。

步骤 1: 确定隐层神经元个数,随机设定输入层与隐层间的连接权值 ω 和隐层神经元的偏置 b ;

步骤 2: 选择一个无限可微的函数作为隐层神经元的

激活函数,进而计算隐层输出矩阵 H ;

步骤 3: 计算输出层权值 $\hat{\beta}; \hat{\beta} = H^+ T$ 。

1.2 差分进化

差分进化 (Differential Evolution, DE) 是 1995 年由 Storn 等人提出的,是一种全局搜索的基于群体进化的算法。该算法的鲁棒性和全局收敛能力较强,并且具有记忆个体最优解和种群内信息共享的特点。它的基本思想是在种群中随机选择 3 个个体向量,其中一个作为基准向量,之后将两个个体向量的差分量作为基准向量的扰动量,从而得到变异向量,然后变异向量与基准向量进行杂交,得到试验向量,最后基准向量与试验向量进行对比,留下较优者,保存在下一代群体之中。这样就可以逐步的改善群体质量,寻求最优解^[15]。具体步骤如下:

步骤 1: 初始化种群。设个体维数为 D ,种群规模为 NP ,进化代数数为 t , $X(t)$ 为第 t 代种群。首先产生第 0 代种群:

$$X(0) = [x_1(0), x_2(0), \dots, x_{NP}(0)] \quad (5)$$

其中: $x_i(t) = (x_{i,1}(t), x_{i,2}(t), \dots, x_{i,D}(t))$,表示第 t 代种群中的第 i 个个体。

个体取值可以由下式表示:

$$x_{i,j}(0) = L_j + rand_{i,j}[0,1](U_j - L_j) \quad (6)$$

其中: $1 \leq i \leq NP, 1 \leq j \leq D, [L_j, U_j]$ 为第 j 维上的取值范围, $rand_{i,j}[0,1]$ 为 0 和 1 之间的随机数。

步骤 2: 适应度函数。先使用 ELM 算法计算出输出权值矩阵,之后设定均方根误差为差分进化算法的目标函数,从个体中选取样本进行验证,得到样本的适应度。

步骤 3: 变异。假设对第 i 个基准向量 $x_i(t)$ 进行变异操作,变异向量可以由如下的形式来表示:

$$v_i(t) = x_{r1}(t) + F(x_{r2}(t) - x_{r3}(t)) \quad (7)$$

其中: $x_{r1}(t), x_{r2}(t)$ 和 $x_{r3}(t)$ 为从种群中随机选择的 3 个向量, $r1, r2, r3 \in [1, NP]$,且 $r1 \neq r2 \neq r3$ 。 F 为缩放因子,取值范围为 $[0, 1]$ 。3 个向量选择完毕之后将两个向量进行差分,然后与第 3 个向量相加,得到变异向量 $v_i(t)$ 。

步骤 4: 交叉。引入交叉运算的目的是为了提高种群的多样性,交叉之后得到的个体可以由下式表示:

$$u_{i,j}(t) = \begin{cases} v_{i,j}(t), rand_j[0,1] < CR \text{ or } j = j_{rand} \\ x_{i,j}(t), otherwise \end{cases} \quad (8)$$

其中: $u_{i,j}(t)$ 是新个体, $v_{i,j}(t)$ 是变异产生的新个体, $x_{i,j}(t)$ 是初始个体, CR 是交叉概率,取值范围为 $[0, 1]$, j_{rand} 是 $[1, D]$ 之间产生的随机整数。

步骤 5: 选择。将交叉之后的个体 $u_{i,j}(t)$ 与初始个体 $x_{i,j}(t)$ 进行比较,比较的指标就是它们的适应度函数,即它们的均方根误差,比较之后取较小的作为下一代个体。整个过程可以由下式来表示:

$$x_i(t+1) = \begin{cases} u_{i,j}(t), f(u_{i,j}(t)) \leq f(x_{i,j}(t)) \\ x_{i,j}(t), f(u_{i,j}(t)) > f(x_{i,j}(t)) \end{cases} \quad (9)$$

其中: f 为适应度函数,即样本的均方根误差。经过差分进化进行优化之后的极限学习机算法流程可

以由图 2 表示:

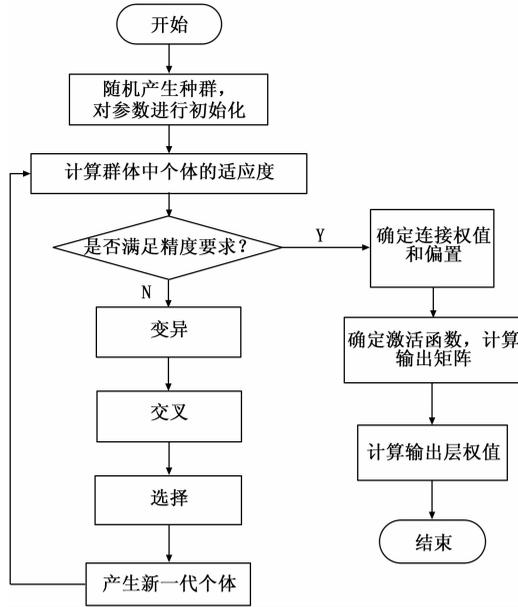


图 2 DE-ELM 算法流程图

2 操纵系统的故障评估

2.1 数据选择

飞行员通过控制飞机操纵系统实现对飞机的操控, 操纵系统的正常与否不仅影响飞机的飞行姿态、飞行品质等等, 更重要的是直接影响到飞机的安全以及机上人员的安全。尤其是对于军用飞机来讲, 操纵系统一旦出现故障, 就难以完成战斗任务, 难以实现对于敌对目标的精确打击, 从而导致战争的失利。

飞机是由舵面来保持其飞行姿态的稳定, 操纵系统则是对舵面进行直接的控制, 从而能够使飞机能够稳定的飞行。而舵面可以分为以下两种: 主要舵面和辅助舵面。其中主要舵面有: 左右副翼、升降舵、方向舵。其功能就是实现飞机绕 x、y、z 三轴运动。辅助舵面有: 前缘缝翼、前缘襟翼、后缘襟翼、飞行扰流板、地面扰流板和水平安定面。其主要功能是改善飞机的操纵和爬升性能。本文主要对飞机主要舵面进行故障的评估, 所以选取了副翼、升降舵、方向舵作为输出量。

本文中采用的数据是某型飞机从起飞至降落整个过程中产生的飞参数据, 输入量为驾驶杆横向位移、驾驶杆纵向位移和脚蹬位移, 输出量为左副翼角位移、右副翼角位移、升降舵角位移、方向舵角位移。将前 1000 个数据作为训练集, 随后的 100 个数据作为测试集。部分飞参数据如表 1 所示:

2.2 评价指标

为了评估该方法, 使用均方误差和决定系数两个指标来进行评价。均方误差 E 以及决定系数 R^2 计算公式如下:

$$E = \frac{1}{l} \sum_{i=1}^l (\hat{y}_i - y_i)^2, i = 1, 2, \dots, n \quad (10)$$

表 1 部分飞参数据

左副翼角位移/°	右副翼角位移/°	升降舵角位移/°	方向舵角位移/°	驾驶杆横向位移/mm	驾驶杆纵向位移/mm	脚蹬位移/°
-2.18	2.09	2.63	2.53	3.15	2.56	2.25
-2.51	2.54	2.49	2.53	3.35	2.66	2.25
-2.62	2.74	2.49	2.53	3.49	2.65	2.25
-2.6	2.7	2.42	2.53	3.44	2.69	2.25
-2.61	2.69	2.43	2.53	3.45	2.69	2.25
-2.57	2.64	2.42	2.54	3.39	2.69	2.24

$$R^2 = \frac{(l \sum_{i=1}^l \hat{y}_i y_i - \sum_{i=1}^l \hat{y}_i \sum_{i=1}^l y_i)^2}{(l \sum_{i=1}^l \hat{y}_i^2 - (\sum_{i=1}^l \hat{y}_i)^2)(l \sum_{i=1}^l y_i^2 - (\sum_{i=1}^l y_i)^2)} \quad (11)$$

其中: l 为测试集样本个数。 \hat{y}_i ($i = 1, 2, \dots, n$) 为第 i 个样品的预测值; y_i ($i = 1, 2, \dots, n$) 为第 i 个样品的真实值; n 为样本的数目。均方误差 E 越小, 决定系数 R^2 ($0 < R^2 < 1$) 越接近于 1, 模型的拟合程度越好。

2.3 实验过程

选取 ELM 算法中的隐含层个数为 30, 选取 DE 算法中的种群规模 $NP=30$, 交叉概率 $CR=0.8$, 缩放因子 $F=1$, 最大迭代次数为 50 次。通常情况下, NP 一般取 20-50, NP 越大其种群的多样性就越强^[16], 但是寻求最优解的时间也会相应的增加, 为了提高故障评估的实时性, 所以说选择 30 为种群规模。利用 DE-ELM 对左副翼角位移 (FUL)、右副翼角位移 (FUR)、升降舵角位移 (FUPL)、方向舵角位移 (FUPR) 的拟合效果如图 3~6 所示, 拟合误差如图 7~10 所示。

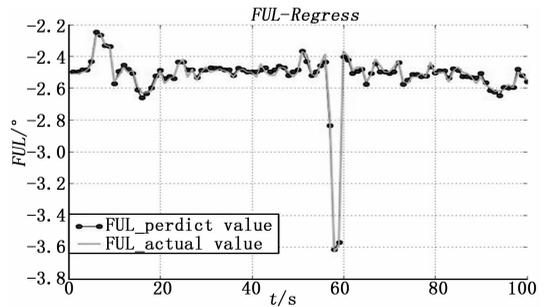


图 3 左副翼角位移拟合

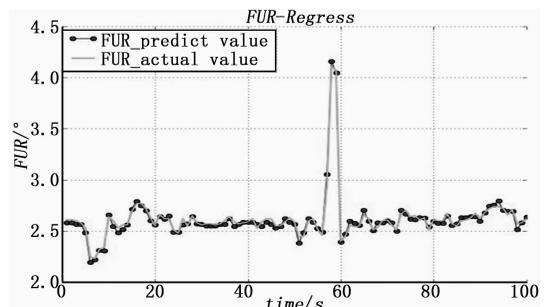


图 4 右副翼角位移拟合

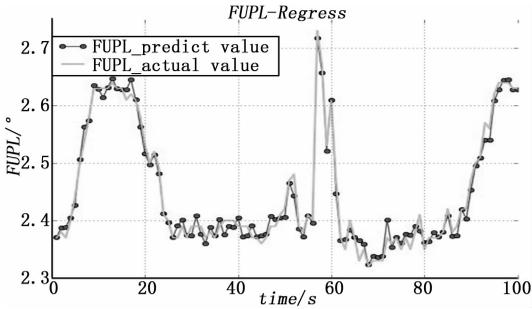


图 5 升降舵角位移拟合

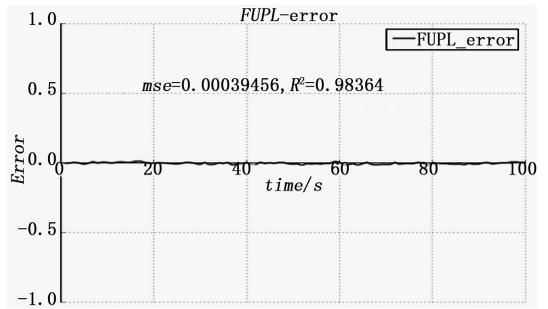


图 10 方向舵角位移误差

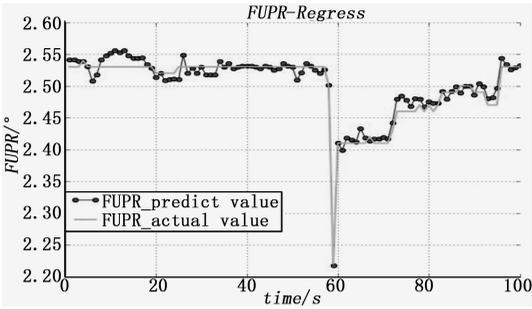


图 6 方向舵角位移拟合

图 7 左副翼角位移误差

图 8 右副翼角位移误差

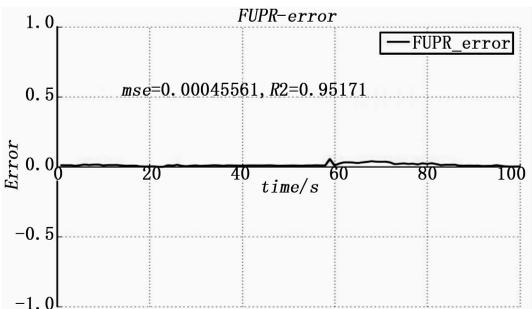


图 9 升降舵角位移误差

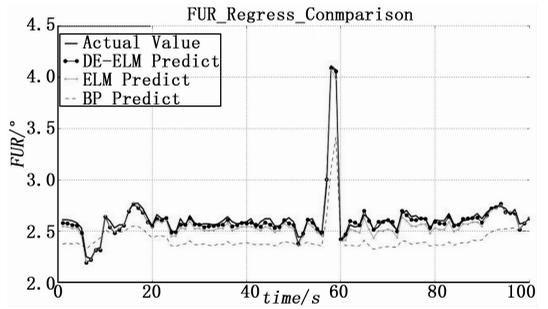


图 11 3种方法对右副翼角位移的拟合

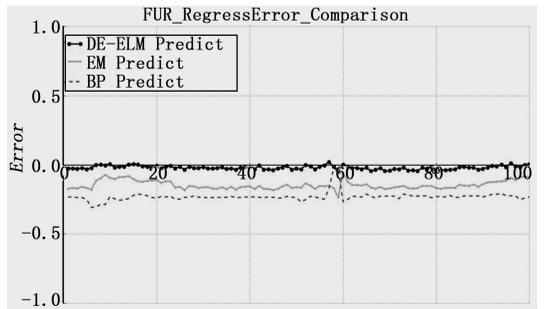


图 12 3种方法对右副翼角位移拟合的偏差

由图中可以看出 DE-ELM 的预测数据非常接近原始数据，其相对误差均在 0 附近，均方误差均小于 5×10^{-4} ，决定系数均大于 0.95，可以认为 DE-ELM 对于飞参数据的拟合有着非常好的效果。为了更加直观地体现该方法的优点，下面将针对同一组数据，分别用 BP 神经网络、ELM 算法和 DE-ELM 算法对其进行训练，对比最终的拟合结果。选择右副翼角位移为输出量，操纵杆的纵向位移和横向位移以及脚踏位移为输入量。设置 BP 神经网络隐含层神经元个数为 5，迭代次数 1000，目标误差 1×10^{-4} ，学习速率 0.01；设置 ELM 隐含层神经元个数为 30；DE-ELM 相关参数设置与之前相同。3 种不同方法对同一数据进行拟合的效果如图 11 所示，与真实数据的偏差如图 12 所示。

由图可以很直观地看出来，DE-ELM 的偏差非常接近于 0，ELM 的偏差约在 -0.1 左右，而 BP 神经网络的偏差约在 -0.25 左右。为了更加全面的来评价这 3 种方法，表 2 利用均方误差 E 和决定系数 R^2 两个指标来对 3 种算法进行评价。由于本次试验中采取的数据有 4 个输出，为了更加全面的体现对数据的拟合效果，所以表中求得的均方误差和决定系数是所有 4 个输出的平均值，而不是选取其中一

个输出作为评价指标。

表 2 BP 神经网络、ELM 和 DE-ELM 的对比

	均方误差 E	决定系数 R ²
BP	0.00213617	0.91659
ELM	0.00123896	0.95096
DE-ELM	0.00044537	0.97605

上述结果表明, DE-ELM 相对于 ELM 和 BP 神经网络来讲, 均方误差分别降低了 64% 和 79%, 同时决定系数也更加接近于 1, 所以说采用 DE-ELM 方法, 可以进一步提高该模型的精度。

以上是针对正常数据进行拟合, 下面将进行对故障数据的检测, 观测该算法是否能够正确检测出故障位置。下面选取了一组左副翼角位移存在故障的数据, 来测试 DE-ELM 算法的对故障数据的检测性能。仿真结果如图 13 所示。

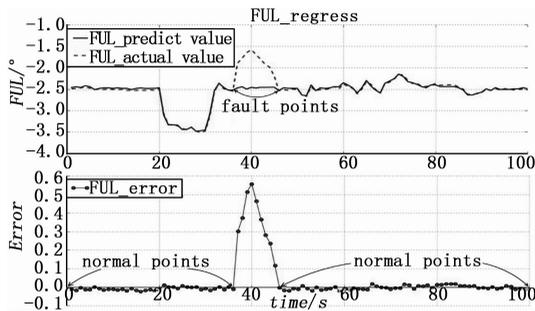


图 13 3 种方法对左副翼角位移拟合的偏差

图 13 中, 上图为使用 DE-ELM 对故障数据进行拟合的情况, 下图为预测数据与真实数据的相对误差。由图中可以看出, 在第 40s 附近, 有一部分数据与预测数据之间存在很大的误差。通过询问相关人员, 此时飞机正是由于操纵系统出现故障导致左副翼出现了不正常的偏转, 所以说, 使用 DE-ELM 方法可以很好的检测出来故障位置, 进而确定出现故障的时间, 而后就可以进行分析到底是由于什么样的原因导致了在该时间段内操纵系统出现了故障, 最终实现故障的排除。

3 结论

本文利用具有良好全局寻优能力的 DE 算法, 结合结构简单、计算速度快的 ELM 算法, 综合之后得到了一种泛化性能好、稳定性强的 DE-ELM 算法。经过实验可以看出, 该法与传统的 BP 神经网络和单纯的 ELM 算法相比, 明显优势是其精确度较高, 除此之外, 由于 DE 算法的加入, 降低了由于 ELM 其本身由于输入权值和隐含层阈值的随机性而带来的不确定性, 增强了 ELM 算法的稳定性, 对于该型飞机的操纵系统模型建立有着良好的效果, 所以说, 通过该法基于飞参数据进行的建立故障评估方法来说有着十分良好的应用前景。

参考文献:

[1] 胡朝江, 陈 列, 杨全法. 飞机飞参系统及应用 [M]. 北京:

国防工业出版社, 2012.

- [2] Li J, Cheng J H, Shi J Y, et al. Brief introduction of back propagation (BP) neural network algorithm and its improvement [J]. *Advances in Computer Science and Information Engineering*, 2012, 169: 553-558.
- [3] Ismail A, Jeng D S, Zhang L L. An optimised product-unit neural network with a novel PSO-BP hybrid training algorithm: Applications to load-deformation analysis of axially loaded piles [J]. *Engineering Applications of Artificial Intelligence*, 2013, 26 (10): 2305-2314.
- [4] Wang H S, Wang Y N, Wang Y C. Cost estimation of plastic injection molding parts through integration of PSO and BP neural network [J]. *Expert Systems with Applications*, 2013, 40 (2): 418-428.
- [5] Wang P, Huang Z Y, Zhang M Y, et al. Mechanical Property Prediction of Strip Model Based on PSO-BP Neural Network [J]. *Journal of Iron and Steel Research, International*, 2016, 15 (3): 87-91.
- [6] Wang S X, Zhang N, Wu L, et al. Wind speed forecasting based on the hybrid ensemble empirical mode decomposition and GA-BP neural network method [J]. *Renewable Energy*, 2016, 94: 629-636.
- [7] Wang W, Li M, Reda Hassanien Emam Hassanien, et al. Optimization of thermal performance of the parabolic trough solar collector systems based on GA-BP neural network model [J]. *International Journal of Green Energy*, 2017, 14 (10): 819-830.
- [8] EL-Bakry M Y, EL-Dahshan E A, Radi A, et al. Optimization of thermal performance of the parabolic trough solar collector systems based on GA-BP neural network model [J]. *International Journal of Green Energy*, 2017, 14 (10): 819-830.
- [9] 邱亚洲. 基于飞参数据的大气数据计算机系统故障诊断 [J]. *计算机与数字工程*, 2017, 3 (45): 449-452.
- [10] Huang G B, Zhu Q Y, Siew C K. Extreme learning machine: theory and applications [J]. *Neurocomputing*, 2006, 70 (1-3): 489-501.
- [11] 丁青锋, 尹晓宇. 差分进化算法综述 [J]. *智能系统学报*, 2017, 12 (4): 431-442.
- [12] 周江曼, 黄清秀, 彭敏放, 等. 基于差分进化优化 ELM 的模拟电路故障诊断 [J]. *计算机工程与应用*, 2014, 50 (8): 221-225.
- [13] Zhai J H, Xu H Y, Wang X Z. Dynamic ensemble extreme learning machine based on sample entropy [J]. *Soft Computing*, 2012, 16 (9): 1493-1502.
- [14] Huang G B, Zhou H M, Ding X J, et al. Extreme learning machine for regression and multiclass classification [J]. *IEEE transactions on systems, man, and cybernetics-part b: cybernetics*, 2012, 42 (2): 513-529.
- [15] Qin A K, Huang V L, Suganthan P N. Differential evolution algorithm with strategy adaptation for global numerical optimization [J]. *IEEE transactions on evolutionary computation*, 2009, 13 (2): 398-417.
- [16] Rainer S, Kenneth P. Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces [J]. *Journal of Global Optimization*, 1997, 11 (4): 341-359.