Aero-engine Thrust Estimation Based on Ensemble of Improved Wavelet Extreme Learning Machine

Zhou Jun, Zhang Tianhong *

Jiangsu Province Key Laboratory of Aerospace Power Systems, College of Energy and Power Engineering,
Nanjing University of Aeronautics and Astronautics, Nanjing 210016, P. R. China

Abstract: Aero-engine direct thrust control can not only improve the thrust control precision but also save the operating cost by reducing the reserved margin in design and making full use of aircraft engine potential performance. However, it is a big challenge to estimate engine thrust accurately. To tackle this problem, this paper proposes an ensemble of improved wavelet extreme learning machine (EW-ELM) for aircraft engine thrust estimation. Extreme learning machine (ELM) has been proved as an emerging learning technique with high efficiency. Since the combination of ELM and wavelet theory has the both excellent properties, wavelet activation functions are used in the hidden nodes to enhance non-linearity dealing ability. Besides, as original ELM may result in ill-condition and robustness problems due to the random determination of the parameters for hidden nodes, particle swarm optimization (PSO) algorithm is adopted to select the input weights and hidden biases. Furthermore, the ensemble of the improved wavelet ELM is utilized to construct the relationship between the sensor measurements and thrust. The simulation results verify the effectiveness and efficiency of the developed method and show that aero-engine thrust estimation using EW-ELM can satisfy the requirements of direct thrust control in terms of estimation accuracy and computation time.

Key words: aero-engine; thrust estimation; wavelet extreme learning machine; particle swarm optimization; neural network ensemble

0 Introduction

Since aero-engine is one of the most critical subsystems in aircraft with the primary duty to provide thrust for aircraft stably, safely and reliably, the main purpose of aero-engine control is to regulate and manage the thrust. However, the thrust of engine cannot be measured directly in flight. Traditional control methods have no choice but to take parameters like rotational speed and pressure ratio that can be measured by sensors as feedback signals. Therefore, the thrust control can be achieved by controlling parameters that are closely related to thrust and can be easily measured[1-2]. Nevertheless, due to the wide working range and rapid changes of working conditions in the whole flight envelope, parameters like rotational speed and pressure ratio cannot represent the thrust accurately any more. In addition, with manufacture and assembly tolerances and gradual degradation of engine performance during service period, the relationship between thrust and sensor measurements becomes more complicated. All these factors lead to inaccuracy in traditional indirect thrust control. In order to guarantee the high performance and safety of aero-engine, the concept of conservative design is usually adopted to reserve enough safety margins so as to make up the uncertainties, but this is at the price of that the potential performance of aero-

engine may be not performed sufficiently[3]. To make full use of the aero-engine potential performance so as to improve thrust and increase the economic efficiency, it is quite necessary to realize direct thrust control where the thrust estimator can be seen as a virtual sensor and the estimated thrust is taken as a feedback signal[4-5]. It can be seen that the estimation of thrust is crucial for direct thrust control, but how to estimate engine thrust accurately is a big challenge. To tackle this problem, this paper proposes an ensemble of improved wavelet extreme learning machine (EW-ELM) for aircraft engine thrust estimation.

Extreme learning machine (ELM) is a novel learning technique for single-hidden layer feedforward neural network, which is proposed by Huang et al. [6-7]. Instead of iteratively adjusting learning parameters like many gradient-based methods, ELM randomly generates the input weights and hidden biases and calculates the output weights analytically. Thus it can learn much faster than traditional neural networks and support vector machine with similar or better generalization performance[8]. With the remarkable advantages, ELM has attracted lots of attentions. However, due to the random initialization of the hidden neuron parameters, limitations still exist such as ill-condition and robustness problems. Besides, the choice of the activation function type also has great influence on the network performance[9-10]. For these factors, the performance of the algorithm may be not so good as expectations when applied to real problems.

To alleviate the above weaknesses, in this work, some modifications are made on the basis of original ELM. As the combination of ELM and wavelet theory has both the excellent properties of wavelet transform and the powerful capability of ELM, wavelet activation functions are used in the hidden nodes to enhance non-linearity dealing ability. Besides, a simple and efficient optimization algorithm, particle swarm optimization (PSO) is used to select the input weights and hid-

den biases. Moreover, the optimal candidates are selected and made ensemble to improve stability. Then some experiments are carried out to train and test the thrust estimator.

1   Preliminaries

In this section, a brief review of ELM and PSO are given to provide necessary backgrounds.

1.1   Extreme learning machine

For a given data set of N arbitrary distinct samples $\Omega = \{\boldsymbol{x}_i,\ \boldsymbol{t}_i\}_1^N$, $\boldsymbol{x}_i = [x_{i1},x_{i2},\cdots,x_{in}]^{\mathrm{T}} \in \mathbf{R}^n$ is the input vector and $\boldsymbol{t}_i = [t_{i1},t_{i2},\cdots,t_{im}]^{\mathrm{T}} \in \mathbf{R}^m$ is the corresponding target vector. The ELM with L hidden nodes and activation function $g(x)$ can be expressed as

$$\sum_{i=1}^{L}\boldsymbol{\beta}_i g(\boldsymbol{w}_i \cdot \boldsymbol{x}_j + b_i) = \boldsymbol{t}_j \qquad j = 1,\cdots,N \quad (1)$$

where $\boldsymbol{w}_i = [w_{i1},w_{i2},\cdots,w_{in}]^{\mathrm{T}}$ is the weight vector connecting the i-th hidden node and the input neurons, b_i the bias of the i-th hidden node, and $\boldsymbol{\beta}_i = [\beta_{i1},\beta_{i2},\cdots,\beta_{im}]^{\mathrm{T}}$ the output weight connecting the i-th hidden node and the output nodes.

The N equations can be rewritten in form of matrix

$$\boldsymbol{H}\boldsymbol{\beta} = \boldsymbol{T} \qquad\qquad (2)$$

where

$$\boldsymbol{\beta} = \begin{bmatrix} \boldsymbol{\beta}_1^{\mathrm{T}} \\ \vdots \\ \boldsymbol{\beta}_L^{\mathrm{T}} \end{bmatrix}_{L \times m}, \qquad \boldsymbol{T} = \begin{bmatrix} \boldsymbol{t}_1^{\mathrm{T}} \\ \vdots \\ \boldsymbol{t}_N^{\mathrm{T}} \end{bmatrix}_{N \times m}$$

and $\boldsymbol{H}$ is the hidden layer output matrix

$$\boldsymbol{H} = \begin{bmatrix} g(\boldsymbol{w}_1 \cdot \boldsymbol{x}_1 + b_1) & \cdots & g(\boldsymbol{w}_L \cdot \boldsymbol{x}_1 + b_L) \\ \vdots & \cdots & \vdots \\ g(\boldsymbol{w}_1 \cdot \boldsymbol{x}_N + b_1) & \cdots & g(\boldsymbol{w}_L \cdot \boldsymbol{x}_N + b_L) \end{bmatrix}_{N \times L}$$

$$(3)$$

As analyzed in Ref. [6], the hidden node parameters $\boldsymbol{w}_i$ and b_i can be simply assigned with random values and do not need tuning, the output weight matrix is the only parameter that should be calculated, which can be determined by seeking the least-square solution to the given linear system as follows

$$\boldsymbol{\beta} = \boldsymbol{H}^{\dagger}\boldsymbol{T} \qquad\qquad (4)$$

where $\boldsymbol{H}^{\dagger}$ is the Moore-Penrose generalized in-

verse of matrix $\boldsymbol{H}$, which can be calculated through orthogonal projection $\boldsymbol{H}^{+} = (\boldsymbol{H}^{\mathrm{T}} \boldsymbol{H})^{-1} \boldsymbol{H}^{\mathrm{T}}$ when $\boldsymbol{H}^{\mathrm{T}} \boldsymbol{H}$ is nonsingular.

On the basis of ridge regression theory[11], it was suggested that a positive value should be introduced to the diagonal elements of $\boldsymbol{H}^{\mathrm{T}} \boldsymbol{H}$ to improve stability and achieve better generalization performance when determining the output weight $\boldsymbol{\beta}$[7].

Thus, Eq.(4) becomes

$$\boldsymbol{\beta} = (\boldsymbol{H}^{\mathrm{T}} \boldsymbol{H} + \lambda \boldsymbol{I})^{-1} \boldsymbol{H}^{\mathrm{T}} \boldsymbol{T} \tag{5}$$

1.2 Particle swarm optimization

PSO is a novel evolutionary algorithm which is developed by Kennedy and Eberhart[12-13]. As a population-based stochastic optimization method inspired by the social behaviors of organisms like fishes in a school or birds in a flock, PSO has found many applications in solving complex optimization problems and shown promising performance[14].

In PSO algorithm, a swarm consists of many particles and each individual, which represents a potential solution to the task at hand, keeps track of its position, velocity and best position found so far. The global best position in the searching space can be found after some iterations. In every iteration, each particle adjusts its velocity to follow the two best values: The best position $\boldsymbol{pbest}_i$ the particle has arrived and the best position $\boldsymbol{gbest}$ any particle in this population have achieved. Then a new position of each particle is obtained and assessed by the fitness function. Suppose the size of the population is N and dimension of the searching space is D. The present position of the i-th particle is represented as $\boldsymbol{X}_i = (x_i^1, x_i^2, \cdots, x_i^d)$, and the movement velocity is denoted as $\boldsymbol{V}_i = (v_i^1, v_i^2, \cdots, v_i^d)$. The particles can update their velocities and positions with the following equations[15]

$$v_i^d(t+1) = w v_i^d(t) + c_1 * rand_1^d [pbest_i^d(t) - x_i^d(t)] +$$
$$c_2 * rand_2^d [gbest^d(t) - x_i^d(t)] \tag{6}$$

$$x_i^d(t+1) = x_i^d(t) + v_i^d(t) \tag{7}$$

$$1 \leqslant i \leqslant N, 1 \leqslant d \leqslant D$$

where w is called the inertial weight, c_1 and c_2 are the learning factors and usually selected as constant 2, and $rand_1^d$ and $rand_2^d$ are random numbers in the region $[0, 1]$.

2 EW-ELM

As discussed in Refs.[9-10], the proper choice of activation functions for the neurons in hidden layer is of great importance to achieve satisfying performance. Aiming at improving the convergence of algorithm, a dual activation function composed by an inverse hyperbolic sine and a Morlet wavelet was adopted in Ref.[16], which obtained good performance. Following that, this paper uses the same activation function in the hidden nodes combining neural and wavelet theory to improve prediction capability, that is

$$g(x) = \frac{1}{2} \left[\mathrm{arcsinh}(x) + \cos(5x) e^{(-0.5x^2)} \right] \tag{8}$$

The dilation and translation factors of wavelets need to be initialized in prior and then the PSO is used to improve the wavelet ELM. The general flowchart for EW-ELM algorithm is depicted in Fig. 1 and the detailed steps are as follows.
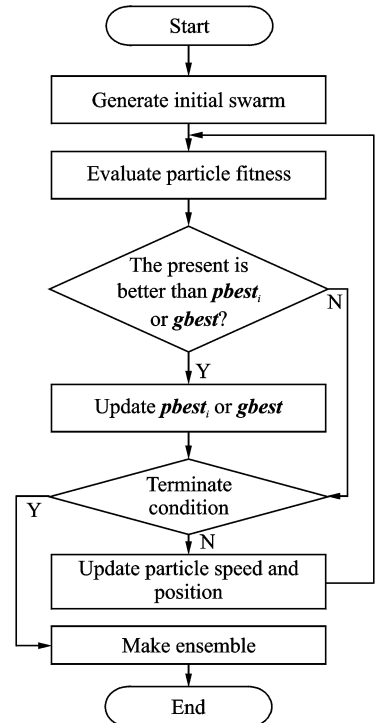

Fig. 1   General flowchart for EW-ELM algorithm

Step 1　Randomly generate the initial particles，including the priori location and velocity of particles. Each particle consists of a set of input weights and hidden biases

$$\boldsymbol{X}_i = [w_{11}, w_{12}, \cdots, w_{1n}; w_{L1}, w_{L2}, \cdots, w_{Ln}; b_1, b_2, \cdots, b_L]$$

Step 2　Compute the output weights and evaluate every particle. For each particle，the corresponding output weights can be obtained with Eq. (5)，then the better individuals are selected according to the fitness of each individual. The fitness function is defined as the root mean squared error (RMSE) on the validation set

$$f(\boldsymbol{X}_{\text{particle}}) =$$

$$\sqrt{\dfrac{\displaystyle\sum_{j=1}^{N} \left\| \displaystyle\sum_{i=1}^{L} \boldsymbol{\beta}_i g(\boldsymbol{w}_i \cdot \boldsymbol{x}_j + b_i) - \boldsymbol{t}_j \right\|_2^2}{\widetilde{N}}} \quad (9)$$

where $\widetilde{N}$ is the number of the samples in validation set and L the number of hidden neurons.

Step 3　Update $\boldsymbol{pbest}_i$ and $\boldsymbol{gbest}$. According to the fitness of all particles in the swarm，the local best position of each individual $\boldsymbol{pbest}_i$ and the global best position $\boldsymbol{gbest}$ of the swarm are computed. As investigated by Zhu[17] and Bartlett[18]，the networks with not only small training error but also small norm of weights seem to be superior in generalization performance. Therefore，both the RMSE on the validation set and the norm of output weights are taken into consideration for the determination of $\boldsymbol{pbest}_i$ and $\boldsymbol{gbest}$[19]. The details are shown as follows

$$\boldsymbol{pbest}_i = \begin{cases} \boldsymbol{X}_i & (f(\boldsymbol{pbest}_i) - f(\boldsymbol{X}_i) > \\ & \alpha f(\boldsymbol{pbest}_i)) \text{ or} \\ & \left(\begin{array}{l} |f(\boldsymbol{pbest}_i) - f(\boldsymbol{X}_i)| < \\ \alpha f(\boldsymbol{pbest}_i) \text{ and } \|\boldsymbol{\beta}_{pbest_i}\| > \|\boldsymbol{\beta}_{X_i}\| \end{array} \right) \\ \boldsymbol{pbest}_i & \text{else} \end{cases}$$

$$(10)$$

$$\boldsymbol{gbest} = \begin{cases} \boldsymbol{X}_i & (f(\boldsymbol{gbest}) - f(\boldsymbol{X}_i) > \\ & \alpha f(\boldsymbol{gbest})) \text{ or} \\ & \left(\begin{array}{l} |f(\boldsymbol{gbest}) - f(\boldsymbol{X}_i)| < \\ \alpha f(\boldsymbol{gbest}) \text{ and } \boldsymbol{\beta}_{gbest} > \boldsymbol{\beta}_{X_i} \end{array} \right) \\ \boldsymbol{gbest} & \text{else} \end{cases} \quad (11)$$

where $\boldsymbol{X}_i$，$\boldsymbol{pbest}_i$ and $\boldsymbol{gbest}$ are the current position of the i-th particle，local best position of the i-th particle and the global best position of the swarm，respectively. $f(\boldsymbol{X}_i)$，$f(\boldsymbol{pbest}_i)$ and $f(\boldsymbol{gbest})$ are the corresponding fitness values. $\boldsymbol{\beta}_{X_i}$，$\boldsymbol{\beta}_{pbest_i}$ and $\boldsymbol{\beta}_{gbest}$ are the corresponding output weights attained with Eq. (5). $\alpha > 0$ is a tolerance rate which can be tuned to balance the emphasis on the training error and the norm of output weights[19].

Step 4　Update the speed and position for each particle. Each particle adjusts its speed with Eq. (6) and then the new position is obtained according to Eq. (7). Besides，some limitations should be put on the speed and position of each individual. The above optimization process is repeated until the minimum criterion is achieved or the maximum iterations are completed. Therefore，some improved wavelet ELMs with optimal parameters are attained.

Step 5　Select superior candidates and make ensemble. The idea of neural network ensembles originated in the early 1990s and then spread widely[20]. It has been accepted that the stability of a single neural network can be further improved by using an ensemble of neural networks[21]. As analyzed by Zhou et al.[22]，it tends to perform better performance to ensemble many of the candidate learners rather than all networks. Therefore，a sorting and selecting strategy is adopted based on the training error and the norm of output weights to select superior candidates. Firstly，all the individuals are sorted according to the training error on invalidation set and the first individuals are picked out. Then the individuals are sorted based on the norm of output weights further and the first $2\hat{N}$ individuals are selected to generate ensemble result. As suggested by Ref. [23]，the ensemble using a simple average outperforms optimizing weights，for the latter may lead to overfitting problem. Thus a simple average is utilized to make ensemble in this paper.

3　Aero-engine Thrust Estimation

In this section，the EW-ELM algorithm presented previously is adopted to develop thrust

estimator for a military two-shaft turbofan engine with afterburner. Fig. 2 shows the major components of the aircraft engine, including inlet, fan, compressor, high-pressure turbine (HPT), low-pressure turbine (LPT), afterburner and nozzle. As it is very energy-costing to use real aero-engine for experiments and there may be many risks of accident occurring at test, a simulation model is employed to generate data samples for the training and testing of the thrust estimator. The simulation model used here is a component-level model (CLM) with high confidence which assembles the major components of aero-engine and cal-
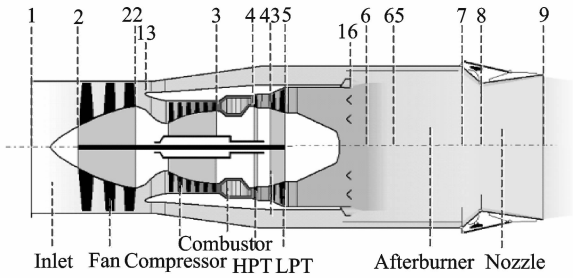

Fig. 2   Simplified diagram for a military turbofan engine

culates engine state variables based on the working conditions and the control inputs from engine controller with the aim to simulate engine characteristics.

As the thrust of aero-engine varies greatly in the full flight envelope, it is difficult to estimate thrust accurately with a sole estimator. A possible solution to this problem is to group the samples in the full envelope to a few clusters in which the thrusts are similar and deviate slightly, and an estimator is designed for each of the clusters. In the thrust estimation phase, firstly the cluster that the sample is belong to is determined by comparing the distance between the sample and all the cluster centers, and then the corresponding thrust estimator is chosen to perform thrust estimation. In this paer, the K-means clustering method is adopted to partition all the samples to ten clusters. The schematic of aero-engine full flight envelop thrust estimator is depicted in Fig. 3.
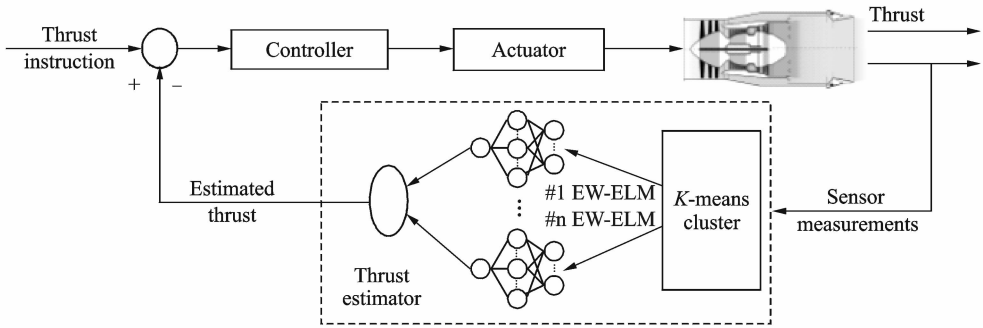

Fig. 3   Schematic of aero-engine full flight envelop thrust estimator

In total, there are as many as 45 output parameters in the CLM. However, not all the parameters can be taken as the inputs of the thrust estimator owing to two cases. One is that not so many sensors can be equipped to measure all the parameters because the weight of an aero-engine will increase and more room is needed to furnish them. Another is that too many inputs may result in a lot of redundancies which tend to increase the computation complexity and have negative impact on the generalization performance. Therefore, it is quite necessary to select proper features as in-

puts. During this process, two principles are taken into consideration: strong measurability of the parameter which will guarantee that the selected parameters are easy to be measured so as to reduce the cost of parameter measurement, and expert experience which will make the selection more reasonable. Finally, six features are selected as the inputs to construct thrust estimator which are the altitude (H), the Mach number (Ma), the bypass exit total pressure (P_{t16}), the main fuel flow (W_f), the afterburner fuel flow (W_{fAB}), and the engine temperature ratio which is

defined as the ratio of afterburner inlet total temperature T_{t6} to compressor inlet total temperature T_{t2}. Before developing the thrust estimator, the inputs are normalized into [0, 1]. The relative deviation (RD) between the estimated thrust and the actual value is taken as the evaluation criteria which is defined as

$$RD = \left| \frac{\text{Estimated value} - \text{Actual value}}{\text{Actual value}} \right| \quad (12)$$

The paper performs all the experiments in MATLAB 2012 environment on a personal computer with i7 processor, 4 GB memory, and Windows 7 operation system. For EW-ELM, the maximum optimization epoch is set to be 30, the population size is selected as 20, and 50 hidden nodes are used. Taking the engine working under steady-state as an example, some simulation data samples in different flight environment and working conditions are collected. The experimental results for one of the thrust estimators, the #1 estimator, are presented in Fig. 4. From Fig. 4, it can be observed that the accuracy of the thrust estimator is satisfying, as the relative deviations on both the training and testing sets are within 0.25%. The detailed results of thrust estimation for aero-engine in full envelop are tabulated in Table 1, in which, MAX represents the maximum RD, MEDIAN means the median RD, and MEAN and STD denote the mean value and the standard deviation of RDs, respectively. As shown in Table 1, the thrust estimator based on support vector regression (SVR) obtains prediction accuracy comparable to that of estimator based on ELM, but it takes much more time to perform prediction. In contrast, the performance of estimator based on ELM seems satisfying, as it has good real time performance and the prediction accuracy is passable. However, it is not quite stable due to the random determination of the learn-

ing parameters and the results are the average of 30 trials. In general, the estimator based on EW-ELM is optimal, which outperforms SVR in prediction time and achieves better robustness compared to ELM. The maximum RD of that thrust estimator is 0.169%, which is also superior to the thrust estimator based on adaptive genetic neural network with maximum RD of 0.281% in Ref. [5]. As a novel learning algorithm, ELM can be performed in high efficiency with good generalization performance and stability after being further optimized by PSO and made ensemble. Thus the thrust estimation using EW-ELM can meet the requirements of direct thrust control in terms of thrust estimation accuracy and prediction time even when the engine works in a wide range flight condition.
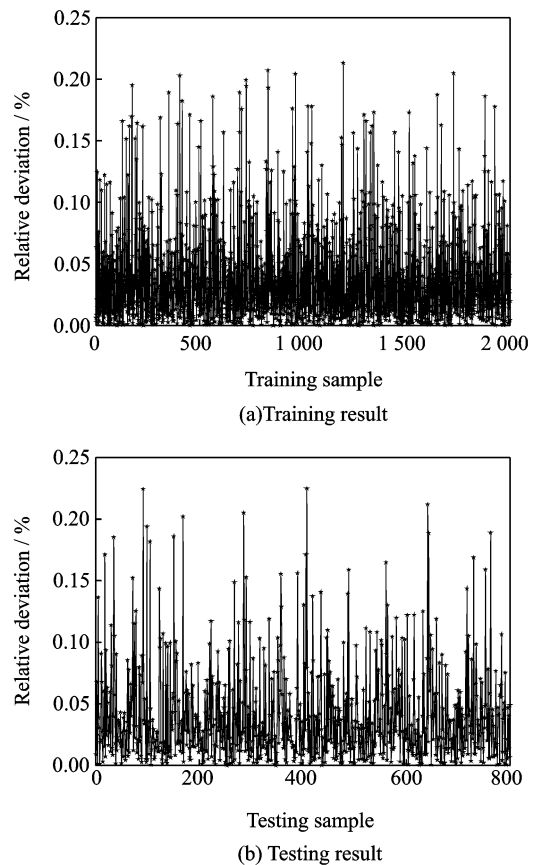

(a)Training result

(b) Testing result

Fig. 4   Experimental results of thrust estimation for #1 estimator

Table 1   Statistical results of performance for full flight envelop thrust estimator

Algorithm	MAX	MEDIAN	MEAN	STD	Test time/ms
SVR	2.52×10^{-3}	1.56×10^{-4}	2.46×10^{-4}	3.13×10^{-4}	2.49
ELM	2.73×10^{-3}	1.74×10^{-4}	2.64×10^{-4}	3.09×10^{-4}	0.13
EW-ELM	1.69×10^{-3}	1.39×10^{-4}	2.10×10^{-4}	2.63×10^{-4}	0.56

Due to the wear, tear, corrosion and fouling over the engine service period, aero-engine health condition will degrade in the lifetime, which tends to cause the engine thrust to deviate from the initial totally healthy state. Hence, in order to achieve direct thrust control, it is very important to guarantee the accuracy of thrust estimation during the engine degrading and aging process. As it has been proved that ELM has good generalization performance after further optimization, and the thrust estimator based on EW-ELM will accommodate the engine performance degradation if the deterioration samples are added to the training set in the training phase of thrust estimator. For evaluating the effectiveness of the method, some experiments are carried out at the engine design point ($H=0$ km, $Ma=0$, PLA$=70°$) and under the cruse condition ($H=12$ km, $Ma=0.8$, PLA$=40°$), where PLA is power level angle.

There are many factors which lead to engine health degradation, and the efficiency and flow rate deviation of the compressor and the turbine are usually adopted for engine health assessment[24]. As the efficiency changes along with the flow capability when the performance of gas path component degrades[25], in this paper only the efficiency deteriorations are considered and the flow rates of the components are determined according to couple factor between the efficiency and flow rate. For the generation of training samples, the efficiency degradations of compressor and turbine are partitioned by an equal interval of 1% in the range of 0—5% and a total of $6^4 = 1\ 296$ samples are collected for each flight condition. By this way the instances of performance degradation in one component or more than one component and even in all the primary gas path components are covered. As for the testing data set, a total of $3^4 = 81$ samples with degradation of 1.3%, 2.5% and 3.8% in the efficiency are generated to test the generalization performance of the method. The results are shown in Fig. 5 and the details are listed in Table 2. It can be seen that both the estimation accuracy and real-time performance are satisfying, which shows the thrust estimation method proposed has good degradation accommodation ability.
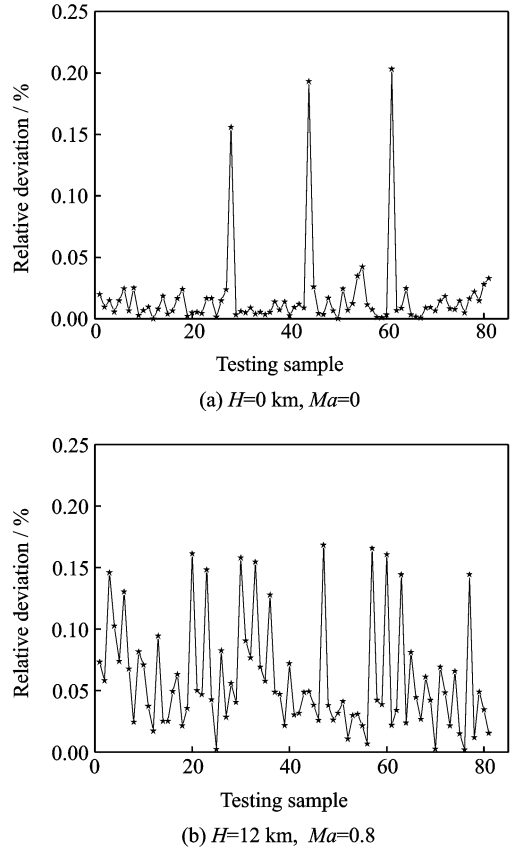

(a) H=0 km, Ma=0

(b) H=12 km, Ma=0.8

Fig. 5　Experimental results of the thrust estimation for aero-engine with health degradation

Table 2　Statistical results of thrust estimation for aero-engine with health degradation

Working condition	Data set	MAX	MEDIAN	MEAN	STD	Test time/ms
$H = 0$ km, $Ma = 0$	Training	$1.71×10^{-3}$	$6.59×10^{-5}$	$1.08×10^{-4}$	$1.84×10^{-4}$	0.48
	Testing	$2.03×10^{-3}$	$8.82×10^{-5}$	$1.76×10^{-4}$	$3.42×10^{-4}$	0.48
$H = 12$ km, $Ma = 0.8$	Training	$3.14×10^{-3}$	$5.49×10^{-4}$	$6.94×10^{-4}$	$5.53×10^{-4}$	0.48
	Testing	$1.68×10^{-3}$	$4.69×10^{-4}$	$5.89×10^{-4}$	$4.45×10^{-4}$	0.48

The estimator designed above is very suitable for the thrust estimation under steady state. As for the thrust estimation during dynamic process, it can be achieved by introducing some previous thrust to the input of EW-ELM. Since aero-engine can be considered as a second-order object approximately, the estimated thrust of the previous two steps, $f(t-1)$ and $f(t-2)$ are added to the input. So as to assess the performance of the thrust estimator, some experiments are performed at the flight height of $H=0$ km and the Mach number of $Ma=0$. Since the feedback thrust is taken as the input parameter, it should be firstly normalized to $[1, 2]$. In the training phase, the throttle lever is pushed form $25°$ to $70°$, and then pulled back from $70°$ to $25°$, thus as many as 1 196 train samples are collected. In the testing phase, the throttle lever is randomly pushed and pulled in the closed interval $[25°, 70°]$, and the test results are presented in Fig. 6. The details of the results are given in Table 3. It can be learned that the dynamic thrust estimator can meet the requirement in term of prediction time but the accuracy is not so good as steady-state thrust estimator, as on the testing data set the maximum RD is 1.98% and the percentage of the samples with RD>1% is about 1.90%. As show in Fig. 6, the samples with relatively large prediction error usually gather in the region where the thrust is relatively small. This is due to that the same absolute prediction error will cause lager relative deviation when the actual thrust is small. Nevertheless, for the thrust estimation during dynamic process, the prediction accuracy is acceptable.
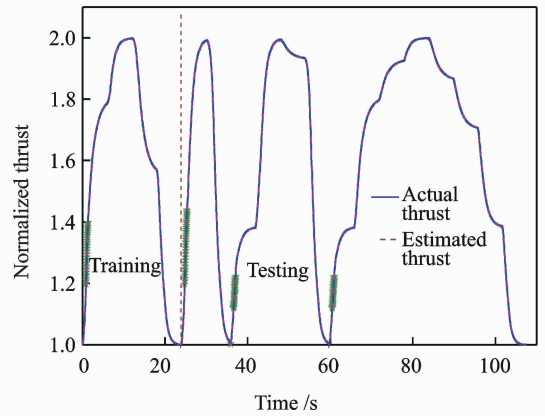

Fig. 6   Experimental results of thrust estimation during dynamic process(RD>1%)

Table 3   Statistical results of thrust estimation during dynamic process

Data set	MAX	MEDIAN	MEAN	STD	Samples with RD>1%	Samples with RD>1.5%	Test time/ms
Training	$1.84×10^{-3}$	$2.03×10^{-5}$	$5.33×10^{-5}$	$1.12×10^{-4}$	0	0	0.51
Testing	$1.98×10^{-2}$	$1.31×10^{-4}$	$6.80×10^{-4}$	$2.21×10^{-3}$	102(1.90%)	45(0.84%)	0.51

4   Conclusions

In this paper, a new learning algorithm named ensemble of improved wavelet extreme learning machine was proposed and used to build aero-engine thrust estimator for direct thrust control. In EW-ELM, PSO was adopted to optimize the parameters for the hidden neurons so as to reduce the negative impact of un-optimal parameters. As the ensemble of neural networks can further improve the stability, the optimal candidates in the population were selected to make ensemble. Not only the training error but also the norm of output weights which is tightly related to gen-eralization performance were considered for the determination of optimal learning parameters and the selection of optimal candidates. Furthermore, a thrust estimator based on EW-ELM was constructed and evaluated using the data generated by an engine component-level model. The results show that the proposed method can generate better generalization performance compared to SVR at faster prediction speed and outperforms original ELM in terms of stability and robustness. It also states that the thrust estimation using EW-ELM can satisfy the requirements of direct thrust control in terms of prediction accuracy. As difference may exist in the results of numerical simula-

tions and the actual effect，future work will focus on the further evaluation of the thrust estimator based on EW-ELM through processor-in-the-loop simulation and hardware-in-the-loop simulation in an operating environment more similar to real world.

Acknowledgements

References：

[1] ZHAO Y P. Support vector regression and their applications to parameter estimation for intellignet aero-engines［D］. Nanjing：Nanjing University of Aeronautics and Astronautics，2009.（in Chinese）

[2] ZHAO Y P，SUN J G. Aeroengine thrust estimation using least squares support vector regression machine［J］. Journal of Aerospace Power，2009，24（6）：1420-1425.（in Chinese）

[3] GASTINEAU Z，HAPPAWANA G，NWOKAH O D I. Robust model-based control for jet engines［C］// AIAA/ASME/SAE/ASEE Joint Propulsion Conference ＆ Exhibit，34th. Cleveland，OH：［s. n.］，1998.

[4] MAGGIORE M，ORDÓÑEZ R，PASSINO K M，et al. Estimator design in jet engine applications［J］. Engineering Applications of Artificial Intelligence，2003，16(7)：579-593.

[5] YAO Y L，SUN J G. Application of adaptive genetic neural network algorithm in design of thrust estimator［J］. Journal of Aerospace Power，2007，22(10)：1748-1753.（in Chinese）

[6] HUANG G B，ZHU Q Y，SIEW C K. Extreme learning machine：Theory and applications［J］. Neurocomputing，2006，70(1)：489-501.

[7] HUANG G B，WANG D H，Lan Y. Extreme learning machines：A survey［J］. International Journal of Machine Learning and Cybernetics，2011，2(2)：107-122.

[8] HUANG G B，ZHOU H，DING X，et al. Extreme learning machine for regression and multiclass classification［J］. Systems，Man，and Cybernetics，Part B：Cybernetics，IEEE Transactions on，2012，42(2)：513-529.

[9] DAQI G，GENXING Y. Influences of variable scales and activation functions on the performances of multilayer feedforward neural networks［J］. Pattern Recognition，2003，36(4)：869-878.

[10] GOMES G S S，LUDERMIR T B，LIMA L M M R. Comparison of new activation functions in neural network for forecasting financial time series［J］. Neural Computing and Applications，2011，20(3)：417-439.

[11] HOERL A E，KENNARD R W. Ridge regression：applications to nonorthogonal problems［J］. Technometrics，1970，12(1)：69-82.

[12] EBERHART R C，KENNEDY J. A new optimizer using particle swarm theory［C］//Proceedings of the Sixth International Symposium on Micro Machine and Human Science. ［S. l. ］：IEEE，2002：39-43.

[13] KENNEDY J. Encyclopedia of machine learning ［M］.［S. l. ］：Springer US，2011：760-766.

[14] GU W B，TANG D B，ZHENG K. Solving job-shop scheduling problem based on improved adaptive particle swarm optimization algorithm［J］. Transactions of Nanjing University of Aeronautics ＆ Astronautics，2014，31(5)：559-567.

[15] SHI Y，EBERHART R. A modified particle swarm optimizer ［C］//Evolutionary Computation Proceedings，1998. Anchorage，USA：IEEE，1998：69-73.

[16] JAVED K，GOURIVEAU R，ZERHOUNI N. SW-ELM：A summation wavelet extreme learning machine algorithm with a priori parameter initialization ［J］. Neurocomputing，2014，123：299-307.

[17] ZHU Q Y，QIN A K，SUGANTHAN P N，et al. Evolutionary extreme learning machine ［J］. Pattern Recognition，2005，38(10)：1759-1763.

[18] BARTLETT P L. The sample complexity of pattern classification with neural networks：The size of the weights is more important than the size of the network ［J］. Information Theory，IEEE Transactions on，1998，44(2)：525-536.

[19] HAN F，YAO H F，LING Q H. An improved evolutionary extreme learning machine based on particle swarm optimization ［J］. Neurocomputing，2013，116：87-93.

[20] HANSEN L K，SALAMON P. Neural network ensembles ［J］. IEEE Transactions on Pattern Analysis ＆ Machine Intelligence，1990，12(10)：993-1001.

[21] LAN Y，SOH Y C，HUANG G B. Ensemble of online sequential extreme learning machine ［J］. Neurocomputing，2009，72(13)：3391-3395.

［22］ ZHOU Z H，WU J，TANG W. Ensembling neural networks：Many could be better than all［J］. Artificial Intelligence，2002，137(1)：239-263.

［23］ OPITZ D W，SHAVLIK J W. Actively searching for an effective neural network ensemble［J］. Connection Science，1996，8(3/4)：337-354.

［24］ ZHAO Y S，HU J，TU B F，et al. Simulation of component deterioration effect on performance of high bypass ratio turbofan engine ［J］. Journal of Nanjing University of Aeronautics & Astronautics，2013，45(4)：447-452. (in Chinese)

［25］ HAO Y. Research on civil aviation engine fault diagnosis and life predication based on intelligent technologies ［D］. Nanjing：Nanjing University of Aeronautics and Astronautics，2006. (in Chinese)

Mr. **Zhou Jun** received the B. S. Degree in Flight Vehicle Propulsion Engineering from Nanjing University of Aeronautics and Astronautics in 2011. He is currently a Ph. D. candidate in aerospace propulsion theory and engineering at Nanjing University of Aeronautics and Astronautics. His research is focused on direct thrust control of aero-engine and aero-engine prognositcs and health management.

Prof. **Zhang Tianhong** received his M. S. degree in aero-engine and Ph. D. degree in aerospace propulsion theory and engineering from Nanjing University of Aeronautics and Astronautics in 1994 and 2001，respectively. He is currently a professor at College of Energy and Power Engineering in Nanjing University of Aeronautics and Astronautics. His research interests include the development of aero-engine control system，aero-engine measurement techniques and aero-engine health management techniques.

（Production Editor：Sun Jing）