NOVEL WEIGHTED LEAST SQUARES SUPPORT VECTOR REGRESSION FOR THRUST ESTIMATION ON PERFORMANCE DETERIORATION OF AERO-ENGINE

Su Weisheng[1], *Zhao Yongping*[2], *Sun Jianguo*[3]

(1. Aviation Motor Control System Institute, Aviation Industry Corporation of China, Wuxi, 214063, P. R. China;
2. ZNDY of Ministerial Key Laboratory, Nanjing University of Science and Technology, Nanjing, 210094, P. R. China;
3. College of Energy and Power Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing, 210016, P. R. China)

Abstract: A thrust estimator with high precision and excellent real-time performance is needed to mitigate performance deterioration for future aero-engines. A weight least squares support vector regression is proposed using a novel weighting strategy. Then a thrust estimator based on the proposed regression is designed for the performance deterioration. Compared with the existing weighting strategy, the novel one not only satisfies the requirement of precision but also enhances the real-time performance. Finally, numerical experiments demonstrate the effectiveness and feasibility of the proposed weighted least squares support vector regression for thrust estimator.

Key words: intelligent engine control; least squares; support vector machine; performance deterioration

CLC number: TP391.41; V231 **Document code**: A **Article ID**: 1005-1120(2012)01-0025-08

INTRODUCTION

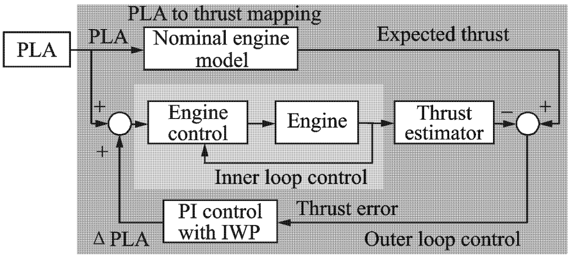
In the past few years, National Aeronautics and Space Administration (NASA) has made great efforts to develop advanced aero-engine control concepts, which has been confirmed in many articles[1-4]. Among these advanced concepts, intelligent engine control (IEC) has drawn much attention since it was proposed by Adibhatla et al[5]. Up to now, IEC has involved many contents including model-based control[6], life extending control[7], performance deterioration mitigating control (PDMC)[8-10]. PDMC shows great promise to accommodate future aero-engines with many advantages including: (1) PDMC can reduce the control dependency on human and realize autonomous operation of the propulsion system by keeping the relationship between power lever angle (PLA) and engine thrust stand still while engine performance deteriorating due to wear. (2) Compared with traditional control methodology, PDMC can achieve the direct control of engine thrust and the unmeasured variable of interest, resulting in less conservative designs which can lengthen engine life and improve operating efficiency. (3) If all the engines on a multi-engine aircraft do not have the same throttle-to-thrust relationship, a thrust imbalance will occur, causing unwanted yaw which requires pilot intervention. PDMC can avoid this.

PDMC consists of two parts: the inner loop control, namely the traditional control methodology for aero-engines, and an additional part, the outer loop control (Fig. 1). Because of outer loop control, PDMC always keeps the same throttle-to-thrust relationship without considering the engine performance deterioration. Meanwhile, it can avoid unwanted yaw caused by thrust imbalance. To implement PDMC, more emphases are put on outer loop control because the inner loop

Foundation items: Supported by the National Natural Science Foundation of China (51006052); the Nanjing University of Science and Technology Outstanding Scholar Supporting Program.

Received date: 2011-05-23; **revision received date**: 2011-08-22

E-mail: y.p.zhao@163.com



Light gray box—Inner loop control, Dark gray box—Outer loop control

Fig. 1 PDMC architecture

control involves the standard engine controller like a full authority digital engine controller (FADEC). Obviously, the outer loop control is comprised of three components: nominal engine model, proportional-integral (PI) controller with integrator windup protection (IWP), and thrust estimator. Nominal engine model only plays a role in PLA for thrust mapping to obtain the expected thrust as a reference signal. PI controller produces the incremental value of PLA (ΔPLA) via the thrust error to modify the input of PLA command implicitly. The third component, thrust estimator, is used to estimate the current thrust of engine in time. Unlike PI controller, the design of thrust estimator seems more difficult. By now, thrust estimator can be realized in two ways. One is the implementation using the conventional numerical methods, i. e. Kalman filter[11]. Since machine learning theory emerged, thrust estimators based on it like neural networks and support vector machine (SVM) have attracted great attention recently[12-14]. But the study is still on the inchoate stage. Most of the machine-learning-based thrust estimators only focused on thrust estimation without concerning performance deterioration. In this study, research on thrust estimator of performance deterioration on the basis of machine learning theory, least squares support vector regression (LSSVR), is conducted to meet the need of PDMC.

SVM[15-16] has prevailed in many fields since it was proposed. Unlike artificial neural network (ANN) which easily fits in local solution, SVM owns a unique global optimization solution with unacceptable training cost for large scale problems, because the training complexity rises geometrically with the increase of the size of training samples. To lessen the training burden, LSSVR was proposed by Suykens, et al[17] to cope with quadratic programming problem in SVM by a linear equation as a surrogate. However, the lack of sparseness and robustness[18] was brought out while using the equality constraints instead of inequality ones and replacing the ε-insensitive loss function by the squared loss function. To overcome those disadvantages, the pruning strategy[19] was proposed by reducing the number of support vectors to shorten the prediction time and enhance the real-time performance. Meantime, the weighting strategy was proposed to enhance the robustness of LSSVR[18], but it did not satisfy all the settings. For example, the weighting strategy in thrust estimator reduced the estimation effectiveness because of its unsuitability. That is to say, the weighting strategy should be adopted appropriately according to the actual settings. Therefore a novel weighting strategy is proposed to improve the robustness of LSSVR while reserving good performance of thrust estimation on the basis of exhaustive analysis of the mentioned references. Finally, the experiments on a nonlinear component level model of dual-spool turbofan engine with mixing exhaust validate the effectiveness and feasibility of the proposed weighting strategy.

1 WEIGHTED LEAST SQUARES SUPPORT VECTOR REGRESSION

Before the introduction of weighted least squares support vector regression (WLSSVR), the normal LSSVR is firstly described. A training data set $\{(x_i, d_i)\}_{i=1}^{N}$ with a size N is given in the normal LSSVR, where x_i is the input variable and d_i the output variable with the value predicted from the value of x_i. The normal LSSVR is to find the normal vector $\boldsymbol{w}$ and the bias b so that for each sample (x_i, d_i), the affine function $f(x_i) = \boldsymbol{w}^{\mathrm{T}} x_i + b$ yields a small deviation between the observed valve d_i and the predicted value $f(x_i)$. The parameters can be obtained by solving the following optimization problem that the model complexity pluses the squared training errors with equality constraints

$$\min_{w,e}\left\{\frac{1}{2}\boldsymbol{w}^{\mathrm{T}}\boldsymbol{w}+\frac{C}{2}\sum_{i=1}^{N}e_i^2\right\}$$
$$d_i=\boldsymbol{w}^{\mathrm{T}}\varphi(x_i)+b+e_i \qquad (1)$$
$$i=1,\cdots,N$$

where $\boldsymbol{e}=[e_1, e_2, \cdots, e_N]^{\mathrm{T}}$, $\varphi(\cdot)$ is a nonlinear mapping which can transform the input data x_i in the input space into $\varphi(x_i)$ in the feature space, $C>0$ the regularization parameter which can control the tradeoff between the flatness of the model and closeness to the training data. Because some samples contributs more than others while obtaining $f(x)$, a weighting factor $v_i(i=1, \cdots, N)$ is introduced to form the optimization problem

$$\min_{w,e}\left\{J(\boldsymbol{w},\boldsymbol{e})=\frac{1}{2}\boldsymbol{w}^{\mathrm{T}}\boldsymbol{w}+\frac{C}{2}\sum_{i=1}^{N}v_ie_i^2\right\} \qquad (2)$$
$$d_i=\boldsymbol{w}^{\mathrm{T}}\varphi(x_i)+b+e_i \quad i=1,\cdots,N$$

Lagrangian function is constructed to solve Eq. (2) as

$$L(\boldsymbol{w},b,\boldsymbol{e};\alpha)=J(\boldsymbol{w},\boldsymbol{e})-\sum_{i=1}^{N}\alpha_i(\boldsymbol{w}^{\mathrm{T}}\varphi(x_i)+b+e_i-d_i) \qquad (3)$$

where $\alpha=[\alpha_1,\cdots,\alpha_N]^{\mathrm{T}}$ is the Lagrangian multiplier vector. According to the Karush-Kuhn-Tucker (KKT) condition of Eq. (2), we have

$$\begin{cases}\dfrac{\partial L}{\partial \boldsymbol{w}}=0\rightarrow \boldsymbol{w}=\sum_{i=1}^{N}\alpha_i\varphi(x_i)\\ \dfrac{\partial L}{\partial b}=0\rightarrow \sum_{i=1}^{N}\alpha_i=0\\ \dfrac{\partial L}{\partial e_i}=0\rightarrow \alpha_i=Cv_ie_i\\ \dfrac{\partial L}{\partial \alpha_i}=0\rightarrow \boldsymbol{w}^{\mathrm{T}}\varphi(x_i)+b+e_i-d_i=0\end{cases} \qquad (4)$$

After eliminating $\boldsymbol{w}$ and e_i, the following equation set is obtained

$$\begin{bmatrix}0 & \boldsymbol{I}_N^{\mathrm{T}}\\ \boldsymbol{I}_N & \boldsymbol{K}+\boldsymbol{V}\end{bmatrix}\begin{bmatrix}b\\ \alpha\end{bmatrix}=\begin{bmatrix}0\\ \boldsymbol{d}\end{bmatrix} \qquad (5)$$

where $\boldsymbol{d}=[d_1, d_2, \cdots, d_N]^{\mathrm{T}}$, $\boldsymbol{I}_N=[\underbrace{1, \cdots, 1}_{N}]^{\mathrm{T}}$, $\boldsymbol{K}$ is a kernel matrix with the elements $K_{ij}=k(x_i, x_j)=\varphi(x_i)^{\mathrm{T}}\varphi(x_j)$, $\boldsymbol{V}=\mathrm{diag}\left\{\frac{1}{Cv_1}, \cdots, \frac{1}{Cv_N}\right\}$ the diagonal matrix, $k(\cdot,\cdot)$ the kernel function. Among many kernel functions the Gaussian kernel function $k(x_i,x_j)=\exp(-\|x_i-x_j\|^2/2\gamma^2)$ is commonly used with a tuning parameter γ. After solving Eq. (5), the prediction function of WLSSVR is obtained

$$f(x)=\boldsymbol{w}^{\mathrm{T}}x+b=\sum_{i=1}^{N}\alpha_ik(x_i,x)+b \qquad (6)$$

Therefore, Suykens, et al[18] gave a scheme of robust estimates to determine the weighting factor $v_i(i=1, \cdots, N)$ by formulating

$$v_i=\begin{cases}1 & \left|\dfrac{e_i}{\hat{s}}\right|\leqslant c_1\\ \dfrac{c_2-\left|\dfrac{e_i}{\hat{s}}\right|}{c_2-c_1} & c_1\leqslant\left|\dfrac{e_i}{\hat{s}}\right|\leqslant c_2\\ 10^{-4} & c_2\leqslant\left|\dfrac{e_i}{\hat{s}}\right|\end{cases} \qquad (7)$$

where $\hat{s}$ is a robust estimate of the standard deviation of the normal LSSVR error variables e_i

$$\hat{s}=\frac{IQR}{2*0.674\,5} \qquad (8)$$

where IQR is the inter-quartile range, the difference between the 75th percentile and 25th percentile. During the estimation of $\hat{s}$, how much the estimated error distribution deviates from a Gaussian distribution is taken into account. The constants c_1 and c_2 are typically chosen as $c_1=2.5$ and $c_2=3$. From the experimental results given by Suykens et al, this weighting scheme can improve the robustness of normal LSSVR due to the involvement of squared errors loss function. In addition, the weighting factor v_i is estimated from a statistical viewpoint. When the training sample x_i induces a large training error, i. e. its corresponding $c_2\leqslant|e_i/\hat{s}|$, the weighting factor is endowed with a very small number as 10^{-4}. In other words, the contribution made by sample x_i to the final target function is negligible. However, in some situations where more emphases are required to put on the training samples inducing large training errors, the weighted strategy proposed by Suykens, et al is inappropriate obviously because the modeling effectiveness may be not improved and even become worse. It is necessary to propose an appropriate weighting strategy to adapt the design of thrust estimator. In this situation, a new weighting strategy is proposed as

$$v_i = 10^{\left|\frac{e_i}{d_i}\right|} \quad i = 1, \cdots, N \tag{9}$$

According to Eq. (9), the larger relative training error the training sample induces, the more weighting factors are given. The presented weighting strategy is more suitable for the design of thrust estimator than Suykens, et al's because more emphases are put on the training samples producing larger relative training errors. The experimental results in section 2 will support it.

It is not enough to design thrust estimator using WLSSVR, since according to Eq. (6) every training sample is a support vector, which limits the real-time performance. The prediction time is in direct proportion to the number of support vectors. Suykens, et al proposed a pruning method as imposing sparseness to prun support values from the sorted support value spectrum. The complete reduction strategy is executed by using this method because if a training sample is pruned, it is completely discarded to retrain WLSSVR. According to Eq. (2), every training sample is a support vector in WLSSVR. Those so-called non-support vectors are imposed and every training sample makes its own contribution to WLSSVR. However, if every training sample is retained as support vector, the real-time performance of WLSSVR will be limited. As a tradeoff, during implementing the pruning method, the partial reduction[20-21] is employed to reduce the number of support vectors and enhance the real-time performance while considering the effects of the pruned training samples on the final WLSSVR. Eq. (5) is unfolded as

$$\begin{bmatrix} 0 & 1 & \cdots & 1 & \cdots & 1 \\ 1 & k_{11}+\dfrac{1}{v_1 C} & \cdots & k_{1i} & \cdots & k_{1N} \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ 1 & k_{i1} & \cdots & k_{i1}+\dfrac{1}{v_i C} & \cdots & k_{iN} \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ 1 & k_{N1} & \cdots & k_{Ni} & \cdots & k_{NN}+\dfrac{1}{v_N C} \end{bmatrix} \begin{bmatrix} b \\ \alpha_1 \\ \vdots \\ \alpha_i \\ \vdots \\ \alpha_N \end{bmatrix} = \begin{bmatrix} 0 \\ d_1 \\ \vdots \\ d_i \\ \vdots \\ d_N \end{bmatrix} \tag{10}$$

If x_i is chosen as non-support vector and pruned, only its corresponding column (the dashed box in Eq. (10) is removed without deleting the corresponding row. Hence, an over-determined linear equation set is obtained as

$$\widetilde{\boldsymbol{K}}\,\tilde{\alpha} = \widetilde{\boldsymbol{d}} \tag{11}$$

where $\widetilde{\boldsymbol{K}}$ is equal to the matrix $(\boldsymbol{K}+\boldsymbol{V})$ without the column corresponding to the sample x_i, $\tilde{\alpha} = [b, \alpha_1, \cdots, \alpha_{i-1}, \alpha_{i+1}, \cdots, \alpha_N]^{\mathrm{T}}$, $\widetilde{\boldsymbol{d}} = [0, d_1, \cdots, d_N]^{\mathrm{T}}$. Generally, Eq. (11) is solved in the least squares sense. It is well-known that the solution to Eq. (11) satisfies the following normal equation

$$\widetilde{\boldsymbol{K}}^{\mathrm{T}}\widetilde{\boldsymbol{K}}\,\tilde{\alpha} = \widetilde{\boldsymbol{K}}^{\mathrm{T}}\widetilde{\boldsymbol{d}} \tag{12}$$

where $\widetilde{\boldsymbol{K}}^{\mathrm{T}}\widetilde{\boldsymbol{K}}$ is the information matrix, whose condition number is double that of $\widetilde{\boldsymbol{K}}$. If Eq. (12) is solved directly, the obtained $\tilde{\alpha}$ becomes bad due to large rounding errors. Instead, QR decomposition with pivoting, a very stable algorithm, is used to solve Eq. (11) in the least squares sense. It is easy to realize with backslash operator in Matlab. The proposed pruned WLSSVR is summarized as follows.

Algorithm 1: the proposed pruned weighted LSSVR (Z-WLSSVR)

Step 1: Initialize the kernel parameter γ, the regularization parameter C, the weighting factor $v_i^{\mathrm{old}} = 1 (i = 1, \cdots, N)$.

Step 2: Construct Eq. (5) with v_i^{old} and solve it.

Step 3: Calculate the predefined performance index. If the performance index degrades, stop, else sort the support values $|\alpha_i|$. And choose a small amount of training samples with smallest support values and remove their corresponding columns from Eq. (10).

Step 4: Compute the weighting factors v_i^{new} of the remaining training samples according to Eq. (9), where $e_i = \dfrac{\alpha_i}{C v_i^{\mathrm{old}}}$.

Step 5: Set $v_i^{\mathrm{old}} = v_i^{\mathrm{old}} \cdot v_i^{\mathrm{new}}$, update the weighting factors in Eq. (11) with v_i^{old}, solve it by using the QR decomposition, and then go to Step 3.

2 EXPERIMENTS

The proposed Z-WLSSVR with the advantages of effectiveness and feasibility is applied to design thrust estimator for PDMC. All the experiments in the study are carried out by Matlab 2007a on a personal computer with Intel (R) Core™ i7 CPU 950 processor and Windows XP operation system. The research object is a nonlinear component level model (CLM) of dual-spool turbofan engine with mixing exhaust (Fig. 2). For convenient comparison and measuring, a performance index RD, relative deviation, is defined as

$$RD = \frac{d_i - f(x_i)}{d_i} \tag{13}$$

where d_i is the measured value, $f(x_i)$ the predicted value.

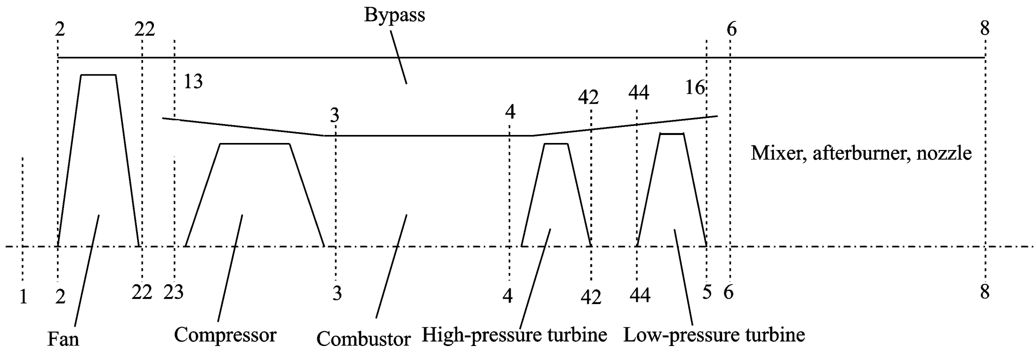


Fig. 2 Illustration of dual-spool turbofan engine with mixing exhaust

Z-WLSSVR is firstly used to design thrust estimator for PDMC. Then Suykens, et al's pruned LSSVR (S-LSSVR) is also utilized to demonstrate the irrationality of Suykens, et al's weighting strategy in this setting. For a fair comparison , all algorithms are initialized with the same model parameters: the kernel parameter γ and the regularization parameter C, determined by cross validation technique[22] with normal LSSVR. In the experiments, the performance deterioration of four components including fan, compressor, high-pressure turbine, and low-pressure turbine is considered with the efficiency degradation ranging from 0%—5%. The PLA scales in the closed interval [25°, 110°]. When PLA is set beyond 75°, the afterburner starts to work. As for the input variables, according to Ref. [13], seven measurements are selected including the altitude (H), Mach number (Ma), the total pressure of the outlet of bypass (P_{16}), the total temperature of section 8 (T_8), the temperature ratio of aero-engine (ETR), the main fuel flow (WFB), and afterburner fuel flow (WFA). These input variables are normalized within the range of [0, 1] before input into thrust estimator because of their different measurements. The experimental results on four flight conditions using the proposed Z-WLSSVR are drawn in Fig. 3. Duo to the space limitation, simulations with other algorithms, including normal LSSVR[17], S-LSSVR[19], S-WLSSVR[18], and Z-WLSSVR, are not mentioned here but the detailed results are listed in Table 1.

According to Fig. 3, the maximum of the absolute value of RD is not more than 3.5‰, which satisfies the requirement of PDMC. When normal LSSVR is used to design thrust estimator, the prediction time is the longest, usually more than 2 ms, which does not meet the requirement of designing aero-engine controller. Hence, S-LSSVR is employed to cut down the prediction time with the enhancement of the real-time performance through reducing the number of support vectors. S-WLSSVR is also utilized to model thrust estimator to further increase real-time performance. Due to the inappropriate weighting strategy, the real-time performance is not improved under the last three flight conditions. The prediction time is shortened obviously by Z-WLSSVR compared with S-LSSVR, which saves more time for engine controller. Because of the success of constructing thrust estimator for PDMC, the effectiveness and

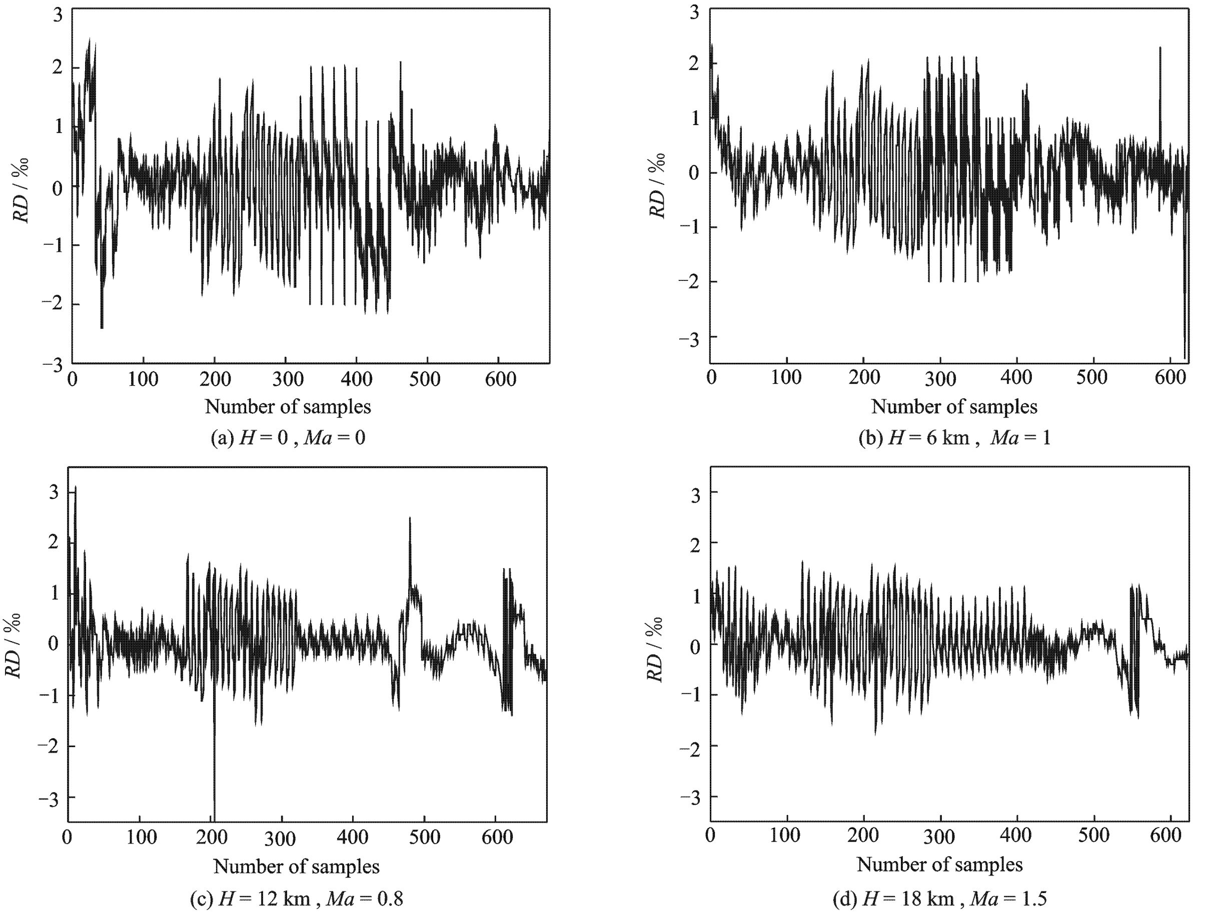


Fig. 3 Experimental results

Table 1 Experimental results

Flight condition	Algorithms	MAX/10^{-3}	MIN/10^{-3}	MEAN/10^{-5}	STD/10^{-4}	Prediction time/ms	#SV
H=0 km Ma=0	Normal LSSVR	2.495 1	−2.177 5	−1.636 8	7.621 3	7.03	3 483
	S-LSSVR	2.600 8	−2.379 7	−5.531 2	8.135 8	1.69	860
	S-WLSSVR	3.279 1	−2.463 1	−5.691 4	9.080 2	1.67	840
	Z-WLSSVR	2.448 4	−2.448 4	0.416 7	8.276 8	1.09	550
H=6 km Ma=1	Normal LSSVR	1.997 2	−3.447 1	−0.015 4	7.261 2	6.70	3 321
	S-LSSVR	1.980 6	−3.499 1	0.414 4	7.390 7	1.53	760
	S-WLSSVR	3.130 9	−3.077 8	20.941 0	10.484 0	1.88	930
	Z-WLSSVR	2.303 2	−3.354 3	3.186 1	8.185 1	0.43	210
H=12 km Ma=0.8	Normal LSSVR	3.054 3	−3.482 7	3.738 4	6.024 2	7.03	3 483
	S-LSSVR	3.455 5	−3.488 0	5.293 0	6.175 4	0.70	400
	S-WLSSVR	3.439 8	−3.455 3	5.858 1	6.327 0	2.93	1 470
	Z-WLSSVR	3.147 4	−3.498 2	6.237 1	6.191 0	0.33	160
H=18 km Ma=1.5	Normal LSSVR	1.440 2	−1.569 7	3.864 5	5.095 1	6.70	3 321
	S-LSSVR	2.414 1	−1.823 8	4.740 9	6.345 2	0.68	350
	S-WLSSVR	3.147 9	−2.612 8	−4.522 2	8.286 0	2.55	1 260
	Z-WLSSVR	1.574 2	−1.664 3	3.262 5	5.918 6	0.40	200

MAX—Maximum of RD, MIN—Minimum of RD, STD—Standard deviation of RD, MEAN—Mean of RD, #SV—Number of support vectors, Prediction time—Prediction time of thrust estimator feasibility of the proposed Z-WLSSVR are confirmed. Meantime, the appropriate weighting strategy should be applied according to the actual settings.

3 CONCLUSION

Recently, the intelligent engine control has attracted much attention. As a representative, performance deterioration mitigating control is an advanced control concept for future aero-engines. The design of thrust estimator becomes the key issue for PDMC with high accuracy and excellent real-time performance under the condition of performance degradation. The normal LSSVR is utilized to improve performance deterioration and S-LSSVR is used to enhance the real-time performance. Considering the actual setting, a novel weighting strategy is proposed to improve the S-WLSSVR's drawback of reducing the real-time performance. By the aid of the proposed Z-WLSSVR, the accuracy of thrust estimator satisfies the requirement of PDMC and its real-time performance is enhanced further. Finally, the numerical experiments validate the proposed Z-WLSSVR and its application.

References:

[1] Litt J S, Simon D L, Garg S, et al. A survey of intelligent control and health management technologies for aircraft propulsion systems[R]. NASA/TM-2005-213622, 2005.

[2] Garg S. NASA Glenn research in controls and diagnostics for intelligent aerospace propulsion systems [R]. NASA/TM-2005-214036, 2005.

[3] Litt J S, Frederick D K, Guo T. The case for intelligent propulsion control for fast engine response [R]. AIAA 2009-1876, 2009.

[4] Simon D L, Garg S. A systematic approach for model-based aircraft engine performance estimation [R]. NASA/TM-2010-216077, 2010.

[5] Adibhatla S, Brown H, Gastineau Z. Intelligent engine control (IEC)[R]. AIAA 92-3484, 1992.

[6] Gastineau Z D, Happawana G, Nwokah O D I. Robust model-based control for jet engines[R]. AIAA-98-3752, 1998.

[7] Guo T. A roadmap for aircraft engine life extending control[C]//Proceedings of the 2001 American Control Conference. Piscataway: IEEE, 2001: 3702-3705.

[8] Litt J S, Turso J A, Shah N. A demonstration of a retrofit architecture for intelligent control and diagnostics of a turbofan engine[R]. AIAA-2005-6905, 2005.

[9] Litt J S, Sowers T S. Evaluation of an outer loop retrofit architecture for intelligent turbofan engine thrust control[R]. AIAA 2006-5106, 2006.

[10] Garg S. Introduction to advanced engine control concepts[R]. 20070010763, 2007.

[11] Litt J S. An optimal orthogonal decomposition method for Kalman filter-based turbofan engine thrust estimaton [R]. NASA/TM-2005-213864, 2005.

[12] Zhao Y, Sun J. Boosting sparse least squares support vector regression (BSLSSVR) and its application to thrust estimation[J]. Transactions of Nanjing University of Aeronautics and Astronautics, 2008, 25(4): 254-261.

[13] Zhao Y, Sun J. Aeroengine thrust estimation using least squares support vector regression machine[J]. Journal of Aerospace Power, 2009, 24(6): 1420-1425. (in Chinese)

[14] Zhao Y, Sun J, Wang Q, et al. Thrust estimator design based on K-means clustering and reduced least squares support vector regression[J]. Journal of Aerospace Power, 2010, 25(5): 1177-1183. (in Chinese)

[15] Vapnik V N. The nature of statistical theory[M]. New York: Springer-Verlag, 1995.

[16] Cortes C, Vapnik V N. Support vector networks [J]. Machine Learning, 1995, 20(3): 273-297.

[17] Suykens J A K, Vandewalle J. Least squares support vector machine classifiers[J]. Neural Processing Letter, 1999, 9(3): 293-300.

[18] Suykens J A K, Brabanter J D, Lukas L, et al. Weighted least squares support vector machines: robustness and sparse approximation[J]. Neurocomputing, 2002, 48(1/4): 85-105.

[19] Suykens J A K, Lukas L, Vandewalle J. Sparse approximation using least square vector machines[C]// IEEE 2000 International Symposium on Circuits and Systems. Piscataway: IEEE, 2000: 757-760.

[20] Valyon J, Horváth G. A robust LS-SVM regression [J]. Transactions on Engineering, Computing and Technology, 2005, 7(8): 148-153.

[21] Zhao Y, Sun J. A non-flat function robust regression algorithm using multi-kernel LS-SVM[J]. In-

formation and Control, 2008, 37(2): 160-165. (in Chinese)

[22] An S, Liu W, Venkatesh S. Fast cross-validation algorithms for least squares support vector machine and kernel ridge regression[J]. Pattern Recognition, 2007, 40(8): 2154-2162.

一种新的加权最小二乘支持向量回归机及其对性能退化航空发动机推力估计的应用

苏伟生[1] 赵永平[2] 孙健国[3]

(1. 中国航空工业集团公司航空推力控制系统研究所,无锡,214063,中国;
2. 南京理工大学智能弹药技术国防重点实验室,南京,210094,中国;
3. 南京航空航天大学能源与动力学院,南京,210016,中国)

摘要:要实现航空发动机的性能退化缓解控制,需要设计一个高精度并具有良好实时性的推力估计器。针对这种需要,本文提出了一种新的加权最小二乘支持向量回归机,并在此基础上设计了性能退化推力估计器。和现有的加权策略相比较,基于此新的加权策略的推力估计器不仅能满足性能退化缓解控制对精度的要求也能满足实时性的要求。最后,仿真实验证明此加权最小二乘支持向量回归机以及基于此回归机设计的推力估计器的有效性和可行性。

关键词:智能发动机控制;最小二乘;支持向量机;性能退化
中图分类号:TP391.41; V231

(Executive editor: Zhang Bei)

基金项目:国家自然科学基金(51006052)资助项目;南京理工大学卓越计划紫金之星资助项目。