Chapter 49
Fault Diagnosis for Aero-engine Multi-redundant Smart Sensors Based on Data Fusion

Xusheng Zhai, Shimei Yang, Gang Li and Jianming Jia

Abstract In order to diagnosis the aero-engine multi-redundant smart sensors, a method based on data fusion was proposed. In this method, an improved fuzzy C-means clustering algorithm was used to get a fusion value based on multisensing units' information, and then the residuals between the fusion value and measured values of the sensing units could be calculated. After that, the residuals could be used to monitor the health conditions of the sensors. The simulation results showed that the fusion value has a high accuracy, and the absolute error is less than 0.5 °C, and also online sensing units fault location could be completed in the form of fault vector.

Keywords Distributed control system · Redundancy technology · Data fusion · Engine

49.1 Introduction

In the aero-engine control system, a large number of sensors act as the system's "eyes and ears", because these sensors are distributed in the harsh environment with high temperature and strong vibration, they would inevitably go wrong. According to statistics, the sensors and actors failure accounted for more than 80 % of total failure in the engine control system [1]. To improve the reliability and capability of fault diagnosis and isolation, the sensor itself is becoming smarter and having more redundant degrees, and the multi-redundant smart sensors are extensively used in the aero-engine distributed control system [2–4]. Based on the traditional multi-redundant sensors, the multi-redundant smart sensors are implanted

X. Zhai (✉) · S. Yang · G. Li · J. Jia
The First Aeronautical College of Air Force, Hangkong Road 23#, Xinyang, China
e-mail: zhai120120@163.com

J. Wang (ed.), *Proceedings of the First Symposium on Aviation Maintenance and Management-Volume I*, Lecture Notes in Electrical Engineering 296,
DOI: 10.1007/978-3-642-54236-7_49, © Springer-Verlag Berlin Heidelberg 2014

microprocessor, data bus interface, and power bus interface [5, 6]. In addition to the traditional functions as sending and receiving data, the multi-redundant smart sensors can also perform data processing, redundancy management, fault diagnosis, fault isolation, and fault tolerance. In this paper, for aero-engine multi-redundant smart sensors fault conditions, a method based on data fusion was proposed, which can realize the fault diagnosis and fault location online.

49.2 The Structure of Multi-redundant Smart Sensor

As shown in Fig. 49.1, the multi-redundant smart sensor consists of multi sensing units, signal conditioning unit, microprocessor, data bus interface, and power bus interface. When the multi-redundant smart sensor works, the target's physical signal is measured by the sensing units, and then preprocessed by the signal conditioning unit before it is sent to the DSP (Digital Signal Processing). The DSP completes the processing of redundant data fusion, self-management, and the communication with the central controller. In order to improve the reliability and the measurement accuracy, redundancy technology is generally applied to the most important sensors. In Fig. 49.1, the sensing units $1 \sim n$ can be located in the same position or different geographical space, if n is 1, it indicates that the sensor has single redundant, and if n is greater than 1, it means the sensor has multi-redundancies. This paper is mainly related to the $n(n \geq 3)$ redundant sensor fault diagnosis.

49.3 Fault Diagnosis Method Based on Data Fusion

Taking a multi-redundant smart temperature sensor as the research object, it uses n redundant sensing units to measure the aero-engine low pressure turbine outlet temperature, and these n redundant sensing units can be managed by the DSP. The fault diagnosis structure based on data fusion is shown in Fig. 49.2.

The fault diagnosis principle shown in Fig. 49.2 can be described as follows: first, the samples of n sensing units would be fused to obtain a fusion value T_f, and T_f would be regarded as the only sensor output value and it is sent to the central controller, the fusion process ensured the output value with a higher reliability, and improved the measurement accuracy; second, a residual could be obtained by comparing the fusion value T_f with n sensing unit sampling values, then the residual should be tested by certain decision-making logic to get a fault vector with elements of 0 or 1; at last sensor fault diagnosis and fault location could be completed by fault vector interpretation.

Fuzzy C-means clustering method [7] was applied for data fusion in this paper. First, fuzzy C-means clustering methods would be used to separate n sampling sequences into an appropriate number of categories, the number of categories should be confirmed according to the number of sensing units and failure modes, and then select the center vector of the category containing the maximum sampling

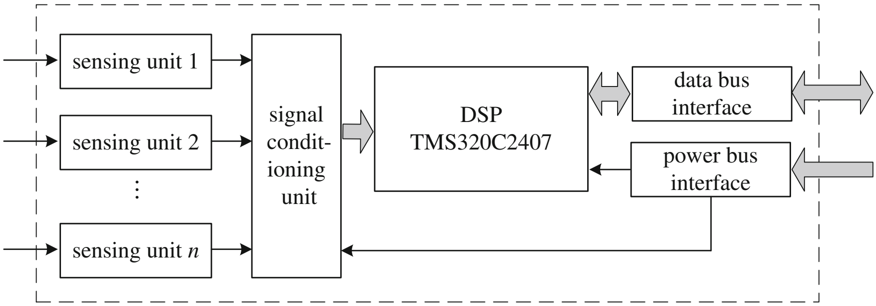


Fig. 49.1 The multi-redundant smart sensor structure

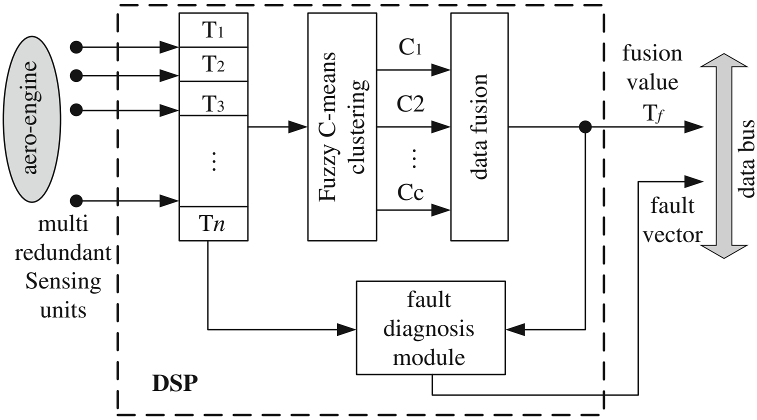


Fig. 49.2 Multi-redundant smart temperature sensor fault diagnosis structure based on data fusion

sequences as the fusion value by the principle of majority. Because the category center vector is nearest to all vectors by Euclidean distance in the same category, it is reasonable to regard the center value as the fusion value.

When the fusion value T_f was obtained, we can get a residual vector $[\varepsilon_1, \varepsilon_2, \varepsilon_3, \ldots, \varepsilon_n]$ by subtracting each sensing unit sample $[T_1, T_2, T_3, \ldots, T_n]$ from T_f. Then a fault vector $F = [f_1, f_2, f_3, \ldots, f_n]$ could be used to complete fault isolation, and the fault vector assignment rules are as follows:

$$\begin{cases} f_i = 0 \ if \ |\varepsilon_i| \le \bar{\varepsilon} \\ f_i = 1 \ if \ |\varepsilon_i| > \bar{\varepsilon} \end{cases} \tag{49.1}$$

In this paper, the fault threshold $\bar{\varepsilon}$ is determined by the mean and standard deviation of the normal sensor residuals, if the mean of the normal residuals was μ, the standard deviation of the sensor normal residuals was σ, then the fault threshold $\bar{\varepsilon}$ would be: $\bar{\varepsilon} = \mu + n\sigma$.

Then we can calculate the fault vector $F = [f_1, f_2, f_3, \ldots, f_n]$ by using formula (49.1), and according to the fault vector, which of the sensing units of the multi-redundant smart sensor is failed could be determined.

49.4 An Improved Fuzzy C-Means Algorithm

The basic thought of C-means classification algorithm is dividing the objects into clusters ensuring the similarity between objects is greatest in the same cluster and is smallest in different clusters. The Fuzzy C-Means algorithm is developed from ordinary C-means algorithm, and the difference between them is that the fuzzy C-means algorithm applied soft fuzzy partition method.

In this method, n vectors of $x_j(1, 2\ldots, n)$ are divided into c clusters of $G_i(i = 1, 2, \ldots, c)$, and the center of each cluster could be calculated by making the value function of non-similarity index minimum. Select Euclidean distance as the non-similarity index to describe the distance between vector x_k and the cluster center c_i in cluster i, then the value function can be defined as:

$$J = \sum_{i=1}^{c} J_i = \sum_{i=1}^{c} \left(\sum_{k, x_k \in Gi} ||x_k - c_i||^2 \right) \tag{49.2}$$

Each cluster can be characterized by a matrix U with a dimension of $c \times n$, the elements $u_{ij} \in [0, 1]$ of the matrix U represents the membership degree of the j-th vector x_j for cluster i. The sum of all the membership degree of a data set is equal to 1 according to the normalization principle:

$$\sum_{i=1}^{c} u_{ij} = 1, \forall j = 1, \ldots, n \tag{49.3}$$

Then, formula (49.2) could be changed as follows:

$$J(U, c_1, \ldots, c_c) = \sum_{i=1}^{c} J_i = \sum_{i=1}^{c} \sum_{j}^{n} u_{ij}^m d_{ij}^2 \tag{49.4}$$

where $d_{ij} = \left\| x_j - c_i \right\|$ is the Euclidean distance between the j-th vector and the i-th cluster's center vector, and $m \in [1, \infty)$ is weighting exponent.

The new objective function is constructed as follows:

$$\begin{aligned} \overline{J}(U, c_1, \ldots, c_c, \lambda_1, \ldots, \lambda_n) &= J(U, c_1, \ldots, c_c) + \sum_{j=1}^{n} \lambda_j \left(\sum_{i=1}^{c} u_{ij} - 1 \right) \\ &= \sum_{i=1}^{c} \sum_{j}^{n} u_{ij}^m d_{ij}^2 + \sum_{j=1}^{n} \lambda_j \left(\sum_{i=1}^{c} u_{ij} - 1 \right) \end{aligned} \tag{49.5}$$

where λ_j, $j = 1, 2 \ldots, n$ is the Lagrange multiplier. Then take the derivative calculation of the input parameters to get the update function of membership degree and cluster center vector:

$$c_i = \frac{\sum_{j=1}^{n} u_{ij}^m x_j}{\sum_{j=1}^{n} u_{ij}^m} \tag{49.6}$$

$$u_{ij} = \frac{1}{\sum_{k=1}^{c} \left(\frac{d_{ij}}{d_{kj}}\right)^{2/(m-1)}} \tag{49.7}$$

However, this algorithm above has a drawback that it does not convergence to the optimal solution every time, the performance of the algorithm depends on the initial membership matrix, so if we can obtain an appropriate initial membership matrix by using certain optimization methods, the problems of the algorithm would be overcome.

PSO (Particle Swarm Optimization, PSO) algorithm is a relatively simple optimization algorithm [8], this paper used PSO algorithm to optimize the initial membership matrix. Basic PSO algorithm can be expressed as follows: it is supposed that there is a particle swarm composed of m particles in the space of D dimension, the spatial position of the i-th particle is defined as $X(i) = (x_{i1}, x_{i2}, \ldots, x_{id})$, $i = 1, 2, \ldots, m$, and its best position that experienced is P_i, the corresponding fitness is F_i, and the flight speed is V_i; the best position of all particles that experienced is named global optimum position and is defined as P_g, the corresponding fitness is F_g. Then the i-th particle of the particle swarm at $n+1$ generation could be iterated according to the following equation:

$$v_{id}^{n+1} = \lambda(w \times v_{id}^n + c_1 \times \mathrm{rand}(1) \times (P_{id}^n - x_{id}^n) + c_2 \times \mathrm{rand}(2) \times (P_{gd}^n - x_{id}^n)) \tag{49.8}$$

$$x_{id}^{n+1} = x_{id}^n + v_{id}^{n+1} \tag{49.9}$$

where λ is shrinkage factor, w is inertia weight, c_1 and c_2 are acceleration factors, $\mathrm{rand}(1)$ and $\mathrm{rand}(2)$ are random numbers in the interval of [0, 1].

Then we can suppose the sample space is: $X = \{x_1, x_2, \ldots, x_N\}$, each particle represents a membership matrix $U = [u_{ij}]_{c \times n}$, and the PSO fitness function can be designed as follows:

$$F(x_i) = \frac{1}{1 + J(U, C)} = \frac{1}{1 + \sum_{i=1}^{c} \sum_{j}^{n} u_{ij}^m d_{ij}^2} \tag{49.10}$$

As can be seen from the Eq. (49.10), the classification result is proportional to the fitness function value. If the classification result improved, $J(U, C)$ will be smaller than before, and the fitness function value will increase.

In summary, the steps of the improved fuzzy C-means classification algorithm can be expressed as follows:

Step 1: given the number of clusters c, tolerance error ε, set the PSO algorithm parameters like swarm size m, shrinkage factor λ, inertia weight w and acceleration factors c_1 and c_2;
Step 2: initialize the particles: $U_1, U_2, \ldots, U_m$;
Step 3: calculate the cluster center and the fitness function according to formula (49.6) and (49.10), and then produce the particle of next generation according to Eqs. (49.8)–(49.9);
Step 4: if the iteration times achieve the maximum time that set before, stop the iteration, and the global optimum position P_g in this case is the optimal membership degree, otherwise go to step 3;
Step 5: update the membership and cluster centers according to Eqs. (49.6) – (49.7), and calculate the value function, if the changes of the value function is less than ε, stop the iteration, otherwise go to step 5 again.

49.5 Fault Diagnosis Simulation

49.5.1 Fault Samples

In order to guarantee the accuracy and reliability of the measurement, a certain turbofan engine installed eight conventional thermocouple sensors to measure the low pressure turbine outlet temperature. It is assumed that conventional thermocouple sensors were replaced by a new smart temperature sensor which had eight sensing units to measure the low pressure turbine outlet temperature, then the eight measured values would be fused by the sensor's DSP to get a fusion value, after that the fusion value would be sent to the central controller and complete fault diagnosis and fault location.

In this paper, eight groups of temperature data were generated by mathematical processing based on a group of real low pressure turbine outlet temperature, and these eight groups of data were regarded as the samples collected by the eight sensing units of the smart temperature sensor. It is assumed that the fourth unit suffered constant bias fault, the sixth unit suffered gain changes faults, and all the anther sensing units were fault-free, the mathematical processing was as follows:

Fig. 49.3 The eight groups of samples that generated ($T_1 \sim T_8$ referred to the eight sensing units)

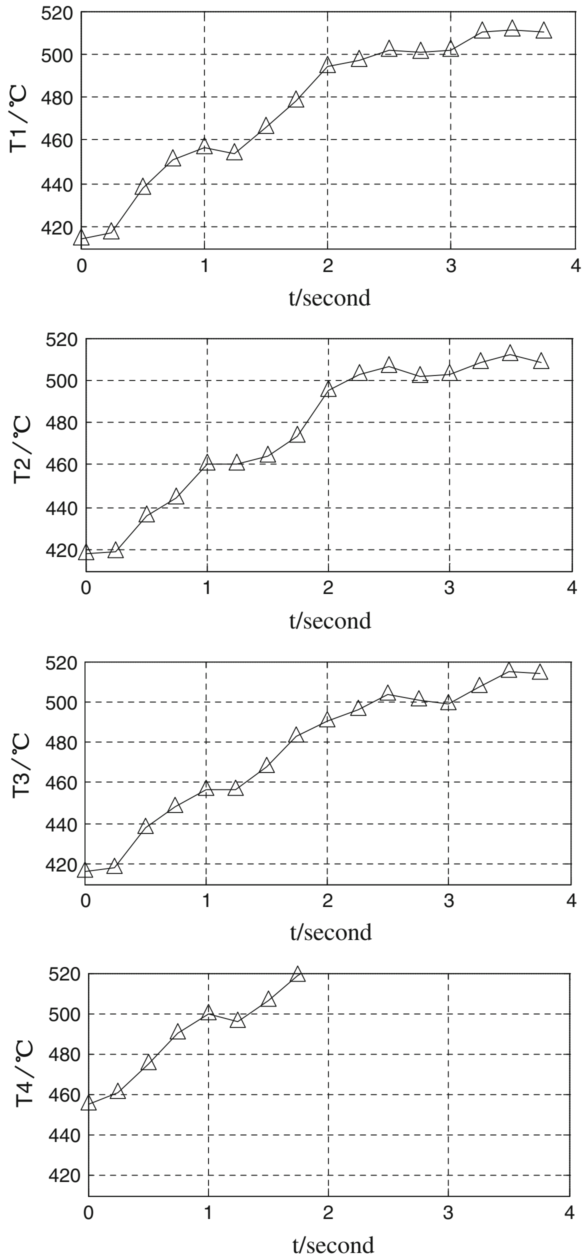
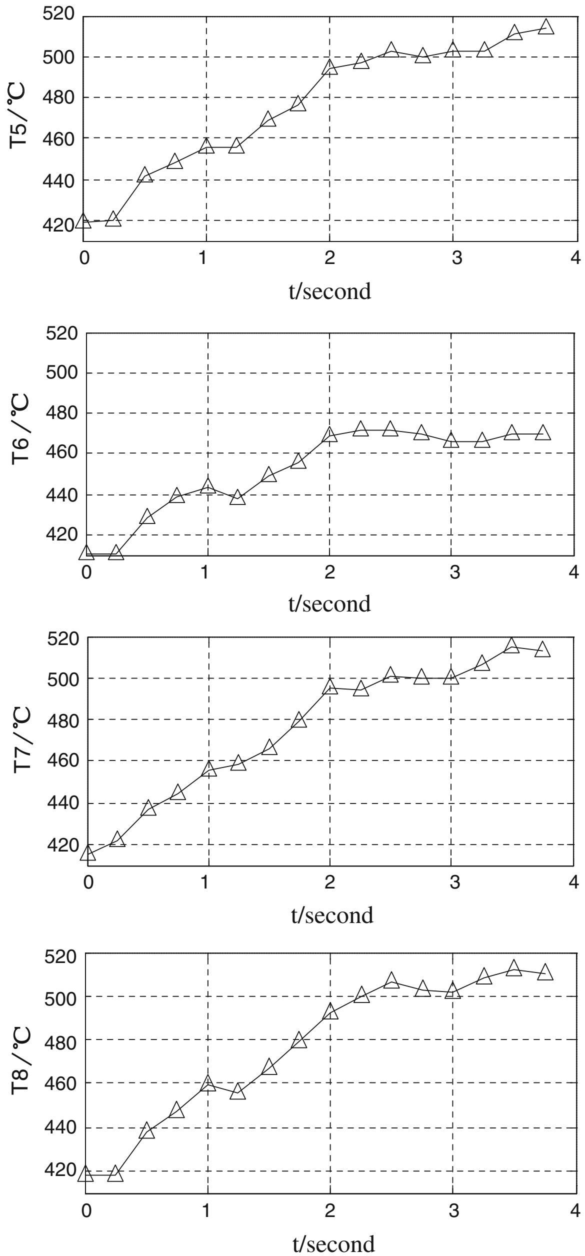


Fig. 49.3 (continued)



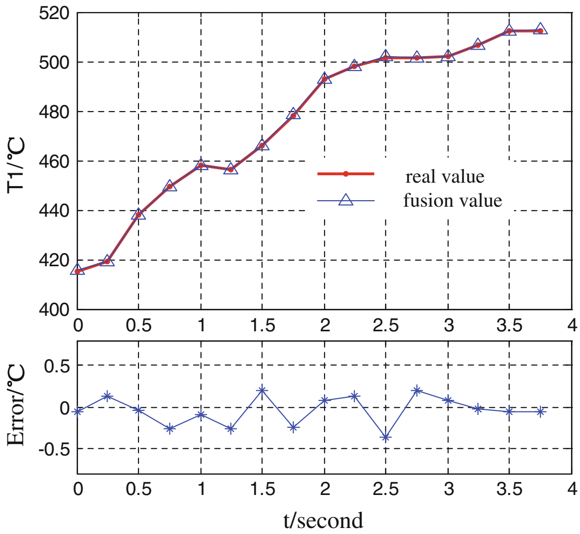


Fig. 49.4 The fusion value and the error between the fusion value and the real value

$$
\begin{aligned}
T_1 &= x + n_1 \\
T_2 &= x + n_2 \\
T_3 &= x + n_3 \\
T_4 &= x + 40 + n_4 \\
T_5 &= x + n_5 \\
T_6 &= x - 10.8 \times t + n_6 \\
T_7 &= x + n_7 \\
T_8 &= x + n_8
\end{aligned} \tag{49.11}
$$

where T_i is the collected data of different sensing unit; x is the real low pressure turbine outlet temperature data; n_i is random numbers with 0-mean and 2-variance; t was time, $i = 1, 2, \ldots, 8$. The eight groups of samples that generated were shown as in Fig. 49.3.

49.5.2 Fault Diagnosis

The parameters were set as follows: the number of clusters was: $c = 3$, the tolerance error was: $\varepsilon = 0.0001$, the PSO swarm size was: $m = 50$, the shrinkage factor was: $\lambda = 0.85$, the inertia weight was: $w = 0.7$, and the acceleration factors were: $c_1 = 1.8$, $c_2 = 2.2$.

In the simulation, the fuzzy C-means algorithm optimized by PSO divided the eight groups of samples into three clusters: $(T_1, T_2, T_3, T_5, T_7, T_8)$, (T_4), (T_6), of

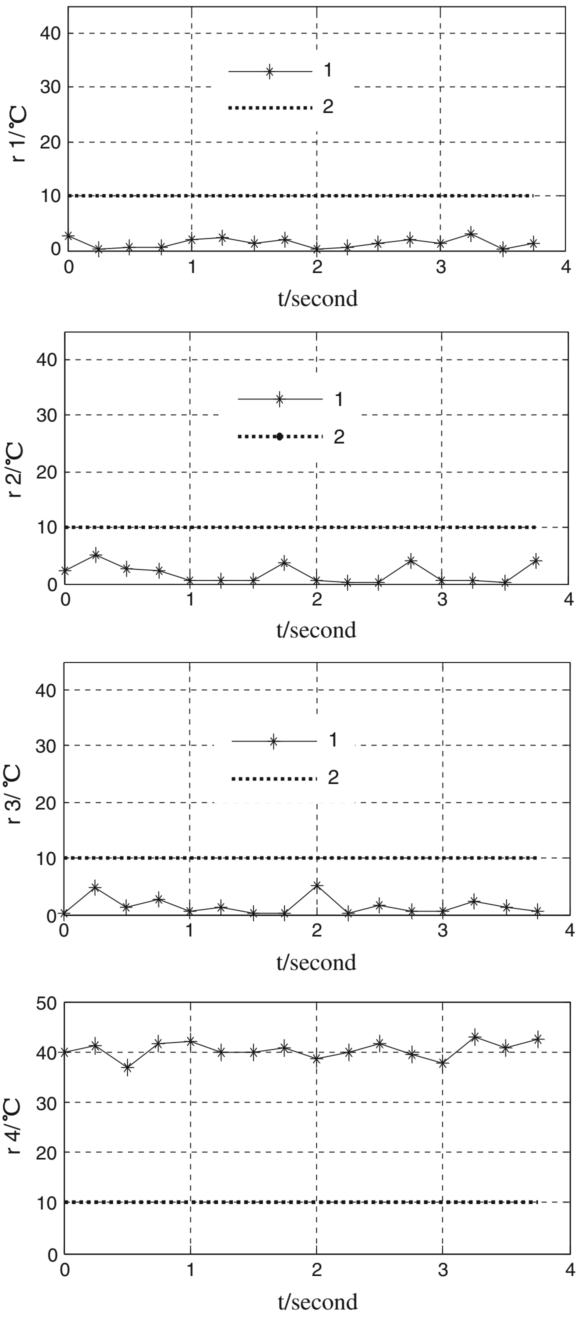
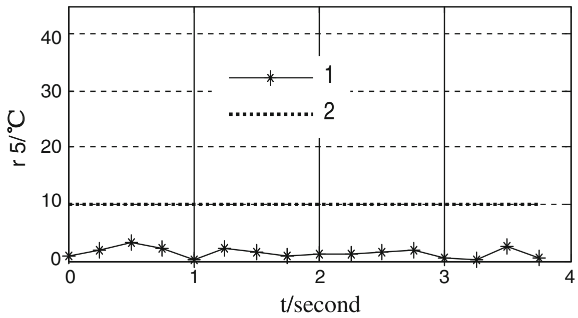
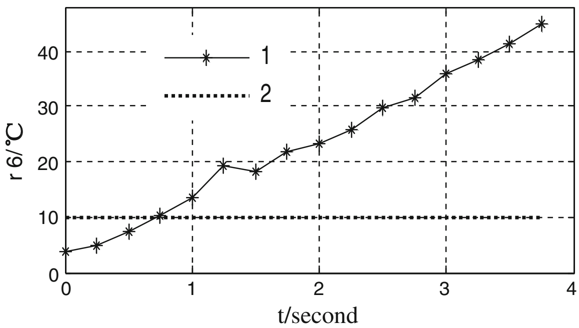
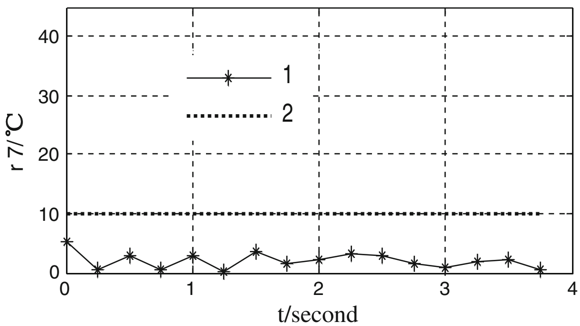


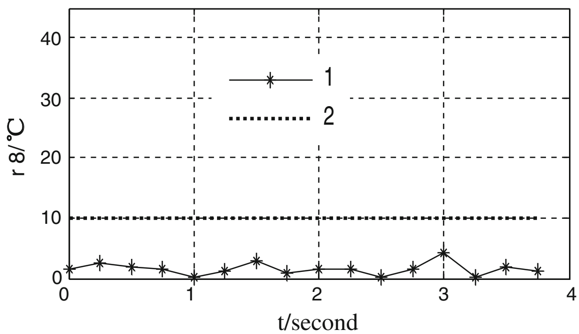
Fig. 49.5 The residual (absolute value) of each sensing unit (1-line: residual; 2-line: threshold)

Fig. 49.5 (continued)









which the first cluster had six groups of samples, so first cluster's center T_f was regarded as the fusion value of the sensing units according to the principle of majority. As shown in Fig. 49.4, the fusion value T_f showed a good approximation to the real value, and the absolute error was less than 0.5 °C.

Then the residuals between the measured values of sensing units and the fusion value were calculated and shown in Fig. 49.5, in which 1-line indicated the absolute values of the residuals and 2-line indicated the thresholds.

However, the threshold should be determined according to the actual data of the low pressure turbine outlet temperature. In this paper, the fault threshold value $\bar{\varepsilon}$ was calculated with the formula: $\bar{\varepsilon} = \mu + n\sigma$, where μ was the mean of the normal sensor residuals, and σ was the standard deviation of the normal sensor residuals. The mean and the standard deviation of the residuals were known because the sensor data was artificially generated. In addition, the low pressure turbine outlet temperature field was not uniform in actual station, so it was necessary to broaden the threshold range in an appropriate level. The threshold was determined as $\bar{\varepsilon} = 10$ °C by considering all the above factors.

After the threshold was determined, a fault vector could be obtained by checking whether the residuals exceeded the threshold. In this case, we could see that the fourth residual and the sixth residual exceeded the threshold, thereby a fault vector of $[0 \quad 0 \quad 0 \quad 1 \quad 0 \quad 1 \quad 0 \quad 0]$ was generated and sent to the central controller, the central controller could directly determine that the fourth and sixth sensing units failed by through of checking the fault vector.

49.6 Conclusions

The structure of an aero-engine multi-redundant smart sensor based on DSP was introduced first, and for aero-engine multi-redundant smart sensors fault conditions, a method based on data fusion which can realize the fault diagnosis and fault location for the multi-redundant smart sensors online was proposed. The method used an improved fuzzy C-means clustering algorithm to get a fusion value, and then the residuals between the fusion value and measured values of these sensing units could be calculated. After that, the residuals could be used to monitor the health conditions of the sensors. The simulation results showed that the fusion value had a high accuracy, and also online fault location could be completed in the form of fault vector.

References

1. He B, Yu D, Shi X (2011) Application of simulation model of sensors to analysis fault of control system of turbojet engine. J Propuls Tech 22:364–367 (in Chinese)

2. Culley DE, Thomas R, Saus J (2007) Concepts for distributed engine control, In: 43rd AIAA joint propulsion conference and exhibit, AIAA-2007-5709
3. Alireza R, Behbahani (2007) Adaptive distributed intelligent control architecture for future propulsion systems. AFRL-PR-WP-TP-2007-228
4. Behbahani A, Culley D, Smith B et al (2008) Status, vision, and challenges of an intelligent distributed engine control architecture. AFRL-RZ-WP-TP-2008-2042
5. Kun Q (2006) The research on areo-engine distributed control and status monitoring system and its control algorithm. Xi'an, Air Force Engineer University (in Chinese)
6. Kailong Cai (2008) The research and experiment on aviation propulsion systems distributed control system. Air Force Engineer University, Xi'an (in Chinese)
7. Jaradat MAK, Langari R (2009) A hybrid intelligent system for fault detection and sensor fusion. Appl Soft Comput pp 415–422
8. Kennedy J, Eberhart RC (1995) Particle swarm optimization. In: IEEE international conference on neural networks. Perth IEEE Service Center, Piscataway, pp 1942–1948