Chinese Journal of Aeronautics, 2013,**26**(3): 717–729

CSAA

Chinese Society of Aeronautics and Astronautics
& Beihang University

Chinese Journal of Aeronautics

cja@buaa.edu.cn
www.sciencedirect.com

Real-time fault detection method based on belief rule base for aircraft navigation system

Zhao Xin, Wang Shicheng *, Zhang Jinsheng, Fan Zhiliang, Min Haibo

High-Tech Institute of Xi'an, Xi'an 710025, China

Received 6 April 2012; revised 23 May 2012; accepted 18 July 2012
Available online 30 April 2013

KEYWORDS

Belief rule base;
Fault detection;
Fault tolerant control;
Integrated navigation;
Parameter recursive estimation algorithm

Abstract Real-time and accurate fault detection is essential to enhance the aircraft navigation system's reliability and safety. The existent detection methods based on analytical model draws back at simultaneously detecting gradual and sudden faults. On account of this reason, we propose an online detection solution based on non-analytical model. In this article, the navigation system fault detection model is established based on belief rule base (BRB), where the system measuring residual and its changing rate are used as the inputs of BRB model and the fault detection function as the output. To overcome the drawbacks of current parameter optimization algorithms for BRB and achieve online update, a parameter recursive estimation algorithm is presented for online BRB detection model based on expectation maximization (EM) algorithm. Furthermore, the proposed method is verified by navigation experiment. Experimental results show that the proposed method is able to effectively realize online parameter evaluation in navigation system fault detection model. The output of the detection model can track the fault state very well, and the faults can be diagnosed in real time and accurately. In addition, the detection ability, especially in the probability of false detection, is superior to offline optimization method, and thus the system reliability has great improvement.

1. Introduction

Near space aircraft plays a profound strategic role in controlling the space resource by virtue of its potentials at early warning, electronic suppression, long-distance quick attack and information collection etc. To fulfill the long range combat task, a reliable navigation technology with high precision is of vital importance. However, in actual navigation system, faults are very commonplace in any subsystem. If these faults are not timely diagnosed, the whole system will be polluted by the output data of the fault subsystem and further drastically declining or even invalidating system precision.[1] Therefore, it is a mandatory and pressing task to study the real-time detection method of navigation system to ensure its reliability and precision.

Until now, many mature fault detection methods have been proposed, which can be classified into three categories, i.e., redundant structure-based methods, analytical model-based methods and non-analytical model-based methods.[2] Here, the analytical model-based methods refer to the methods that

* Corresponding author. Tel.: +86 29 84741232.
E-mail addresses: zhaoxin20062111@163.com (X. Zhao), wshicheng@vip.163.com (S. Wang).

Peer review under responsibility of Editorial Committe of CJA.

ELSEVIER Production and hosting by Elsevier

1000-9361 © 2013 Production and hosting by Elsevier Ltd. on behalf of CSAA & BUAA. Open access under CC BY-NC-ND license.
http://dx.doi.org/10.1016/j.cja.2013.04.039

exploit relationships between system input variables and output variables in conditions where the mathematic model of a system is known. Simply, the analytical model-based methods depend on the physical model of the system, while the non-analytical model-based methods are dependent largely on the observed data without resort to the physical model. However, only a few of them are suitable to solve the online fault detection problems of the navigation systems due to its characteristics, such as its complex operating environment, the limited flight carrying source and the substantial consequences raised by fault. For example, the redundant structure-based methods are not suitable for sensors' fault detection since it brings overwhelming weight to system.[3] Among the analytical model-based methods, ordinary fault detection methods cannot be used directly for navigation system either. Presently, the chi-square test is the most popular method.[4] Chi-square test contains state test and residual test, judging fault through constructing statistical information of state and residual respectively and comparing them with relevant probability statistical distributions.[5] However, both tests have inherent shortcomings in the engineering practice. Without measuring update in the state propagator, state test method has a drastic declining of fault detection sensitivity. Furthermore, it requires learning about system's prior information accurately, otherwise a false diagnosis might occur owing to the improper initial setting. Compared with state test method, residual test method is more popular for its light computation burden, nice real-time performance as well as the flexible design compatibility. However, it is not so effective for soft fault (or gradually changing fault).[6] Actually, as a method based on analytical model, chi-square test relies heavily on system model. However, the analytical model of navigation system is very complex and often difficult to obtain precisely in engineering practice. In contrast, non-analytical model-based methods have been equally popular in such fault detection problem.

The non-analytical model-based methods are in essence data-driven methods that depend on the input/output data rather than the system model. These methods study and train system model using the history data, and then estimate the system output. The fault detection problem of navigation system is related to not only the quantitative measuring data but also subjective knowledge with fuzziness and imperfection of various models. These factors increase the difficulty and complexity to solve such problem with current detection and estimation methods. In addition, we expect to obtain particular fault state from the output results of detection on navigation system, which is to learn the concrete fault conditions. The detection problem is related to the case that the form of output estimation is distributed. But the problem cannot be solved properly by current methods.[7,8] To overcome the afore-mentioned shortcomings, it is needed to apply a flexible and reasonable approach that is able to handle the subjective information and uncertain information. The newly-developed BRB inference methodology using the evidential reasoning approach (RIMER)[9] provides a new solution to process uncertain and mixed information in decision-making concerned with human. Rather than depending on the system's precise model, RIMER can describe the fuzzy, uncertain and non-linear relations, and interpret the output very well. Recently, this method has been used to detect the leakage of oil conveying pipelines,[10,11] security analysis[12] and other fields. One of the keys of this method is to specify belief rule base (BRB) parameter effectively. Therefore, the optimization training methods of BRB parameter are studied and proposed by Yang et al.[13] But those methods are essentially offline ways, and the computation can become expensive when a very large set of data is involved. Because the off-line method in Refs. [12,13] implements parameter estimation by the traditional nonlinear programming based on the least mean squared error measure. In addition, the BRB parameter updating theory studied by Zhou et al.[10] cannot be applied directly to navigation system fault diagnosis. Therefore, it is desirable to develop an online fault detection method which can update the parameters in the established detection model in line with the newly observed data.

This paper studies navigation system fault detection method based on BRB. To alleviate the computation burden and achieve online detection, we propose an online estimation algorithm based on the expectation maximization (EM) algorithm to update the parameters of BRB-based detection model in such a way that the parameters can be updated recursively once new information becomes available. This is largely due to the recursive nature of the EM algorithm by maximizing the likelihood function. In addition, the effectiveness of the proposed method is demonstrated via a navigation experiment platform composed of strapdown inertial navigation system (SINS), global positioning system (GPS) and celestial navigation system (CNS).

2. BRB approach

Based on decision making theory,[14] Dempster–Shafer evidence theory,[15] and fuzzy sets theory,[16] BRB approach describes knowledge with the conception of belief rule and takes evidential reasoning (ER) algorithm[10,17] as reasoning machine, being able to reflect the dynamic nature of decision making problem. The basic approach of BRB relative to this paper is briefly described as follows.

2.1. Basic structure of BRB

In order to catch the system's dynamic nature, BRB consists of the belief rule sets showed as Eq. (1):[9]

$$\begin{aligned} &R_k: \text{ If } x_1 \text{ is } A_1^k \wedge x_2 \text{ is } A_2^k \wedge \cdots \wedge x_M \text{ is } A_M^k \\ &\text{Then } y = \{(F_1, \beta_{1,k}), (F_2, \beta_{2,k}), \ldots, (F_N, \beta_{N,k})\} \end{aligned} \tag{1}$$

where $\boldsymbol{x} = [x_1\ x_2\ \cdots\ x_M]$ denotes the input of the kth rule, M the input sum of rule base, $A_m^k (m = 1, 2, \ldots, M)$ the value of x_m at the kth rule, $\beta_{j,k}$ ($j = 1, 2, \ldots, N$; $k = 1, 2, \ldots, L$) the belief distributed to F_j, and " $\wedge$ " represents logical relation "and".

2.2. BRB reasoning model based on ER algorithm

This section presents the fault detection model structure under the ER frame. Please refer to Refs. [17,18] to find the details about the ER algorithm. Assume that the input and output data of the system can be represented in the form of a data pair ($\boldsymbol{x}(t)$,$y(t)$) while $\boldsymbol{x}(t)$ represents the input at the time instant t and $y(t)$ the detected output value describing system fault state. The relationship between the input and the output can be represented as

$$\hat{y}(t) = f(x_1(t), x_2(t), \ldots, x_p(t)) \tag{2}$$

where $\hat{y}(t)$ is the model estimation of $y(t), \boldsymbol{x}(t) = [\,x_1(t) \quad x_2(t) \quad \cdots \quad x_p(t)\,]$ a p-dimensional input vector, and p the dimensions of basic attributes related to the detection value. Intuitively, the more the basic attributes, the higher precision the estimated detection value is estimated with. However, on the condition of a fixed sample, the more the input dimension is, the sparser the sample is distributed in the space and the fewer the relevant data information appears. The number of the attributes depends on the real situation. The purpose of using ER algorithm is to identify the relationship between the input and the output. The key to solving the detection problem is how to approximate the function $f(\cdot)$.

Define N distinctive evaluation grades as

$$F = \{F_1, F_2, \ldots, F_N\} \tag{3}$$

where $F_n(n = 1,2, \ldots ,N)$ represents the nth fault grade, which can be served as the evaluation grade of the system fault state and is selected according to the practice. It is worth noting that F provides a mutually exclusive complete set. Usually, the more the evaluation grades are, the more specific the studied problem appears, but the heavier the computation burden is.

In order to apply ER algorithm, the input data should be transformed into belief distribution structure. Rule-based equivalence transformation techniques can be used in this case, and more discussion on this issue can be found in Ref. [19]. As a result, each input attribute can be represented as a distribution on referential values using a belief structure. So $\boldsymbol{x}(t) = [\,x_1(t) \quad x_2(t) \quad \ldots \quad x_p(t)\,]$ can be described as

$$S(x_{t,i}) = \{(F_n, \beta_{n,i}(t,i)), \quad n = 1,2,\ldots,N; i = 1,2,\ldots,p\} \tag{4}$$

where $\beta_{n,i}(t,i)$ denotes the matched degree to the grade F_n . The above assessment implies that the attribute $x_{t,i}$ is assessed to the grade F_n with the degree $\beta_{n,i}$. $\beta_{n,i}$ can be generated using various ways, depending on the attribute.[20] The advantage of doing so is that the precise data, the random numbers, and the subjective judgments with uncertainty can be consistently modeled under the same framework.[9,13,17]

After representing each attribute using Eq. (4), the ER algorithm can be directly applied to combine all attributes and generate the final detection result $\hat{y}(t)$. Specifically, $\hat{y}(t)$ has the following belief distribution:[21]

$$O(\hat{y}(t)) = \{(F_n, \hat{\beta}_n(t)),\ n = 1,2,\ldots,N\} \tag{5}$$

where $\hat{\beta}_n(t)$ is the belief of the grade F_n, and it can be obtained by the ER analytical algorithm as follows:

$$\hat{\beta}_n(t) = \frac{\mu\left[\prod_{i=1}^{L}\left(w_i\hat{\beta}_{n,i} + 1 - w_i\sum_{j=1}^{N}\beta_{j,i}\right) - \prod_{i=1}^{L}\left(1 - w_i\sum_{j=1}^{N}\beta_{j,i}\right)\right]}{1 - \mu\left[\prod_{i=1}^{L}(1 - w_i)\right]} \tag{6}$$

$$\mu = \sum_{n=1}^{N}\prod_{i=1}^{L}\left(w_i\beta_{n,i} + 1 - w_i\sum_{j=1}^{N}\beta_{j,i}\right) - (N-1)\prod_{i=1}^{L}\left(1 - w_i\sum_{j=1}^{N}\beta_{j,i}\right)^{-1} \tag{7}$$

where w_i is the drive weight of the ith rule, and it can be obtained as follows:

$$w_i = \theta_i\prod_{m=1}^{p}(\alpha_m^i)^{\delta_m} \Big/ \sum_{l=1}^{L}\theta_l\prod_{m=1}^{p}(\alpha_m^l)^{\delta_m} \tag{8}$$

where $\theta_i \in [0,1]$ $(i = 1, 2, \ldots , L)$ is the relative weight of the ith rule, $\delta_m \in [0,1]$ $(m = 1, 2, \ldots , p)$ the relative weight of the mth input attribute in the ith rule; α_m^i denotes the matching degree between the real input and the referring value A_m^i, and it depends on premise attributes natures and data characteristic.

Eq. (5) represents the overall assessment of fault state at time t. However, in engineering practice, the system's output value $\hat{y}(t)$ is usually numerical. Therefore, it is desirable to generate numerical value equivalent to the distributed assessments Eq. (5). The concept of expected utility is introduced to define such value.[9,13,22] Suppose $u_i(i = 1, 2, \ldots , N)$ represents the utility corresponding to fault grades $F_i(i = 1, 2, \ldots , N)$, plus these natures that decision maker prefers F_j to F_i, so $u_i < u_j$. Generally, utility u_i may be estimated using prior objective or expert knowledge, or it can be obtained from optimism model, and we select the later in this paper. Eventually, the numerical output equivalent to Eq. (5) can be calculated by

$$\hat{y}(t) = \sum_{j=1}^{N}\hat{\beta}_j(t)u_j \tag{9}$$

From Eqs. (5), (6) and (9), it can be seen that the numerical output $\hat{y}(t)$ is a function of $\delta_m(m = 1,2,\ldots,p), \theta_i(i = 1,2,\ldots,L), \boldsymbol{x}, \hat{\beta}_{n,i}$ and u_j $(j = 1, 2, \ldots , N)$. Among them, attribute weight, rule weight and utility are the parameters of BRB reasoning model. If these parameters are proper, the model is correctly established. Otherwise, a huge error tends to occur and may lead to unreliability of the detected result. Therefore, the value of parameters will directly affect the output result. As such, there is a need to develop a method that can optimally learn parameters using observed data information, which will be elaborated in the following section of this paper.

3. Navigation system fault detection model based on BRB

3.1. Integrated navigation system structure design

To improve the precision and reliability of navigation system, the multi-navigation system integrated schemes for aircraft are the most popular policy at present. Among those schemes, INS/GPS integration is the maturest one since it can get steady position, velocity and attitude information. But the invalidity of GPS inevitably happens owing to some factors such as strong jamming, blackout effect appearing during near space flying phase. Therefore, another assistant navigation approach is necessary. As an assistant navigation approach, CNS has a nice autonomy and is able to provide a certain precise attitude adjusting information, being free from the afore-mentioned factors influence. As a result, INS/CNS integration is able to achieve navigation task during the invalidity phase of GPS, seizing GPS information after GPS recovering and furthermore adjusting the system. For this reason, INS/GPS/CNS integration is a feasible approach.[23] To ensure fault tolerance capability, we adopt the non-reset federal filter structure to fuse the information of the three sub-systems, and this structure has the best fault tolerance capability.[24,25] Finally, the fault tolerance filter structure is designed, as shown in Fig. 1. SINS is the main reference system. Two local filter systems are composed of GPS and CNS assisting SINS, respectively. The most optimal state estimation is obtained by fusing the two local filters information in the main filter. It is worth noting that fault detection and isolation device is designed respectively for each local filter to detect and dispose the sub-system fault in time, ensuring the reliability of navigation system.

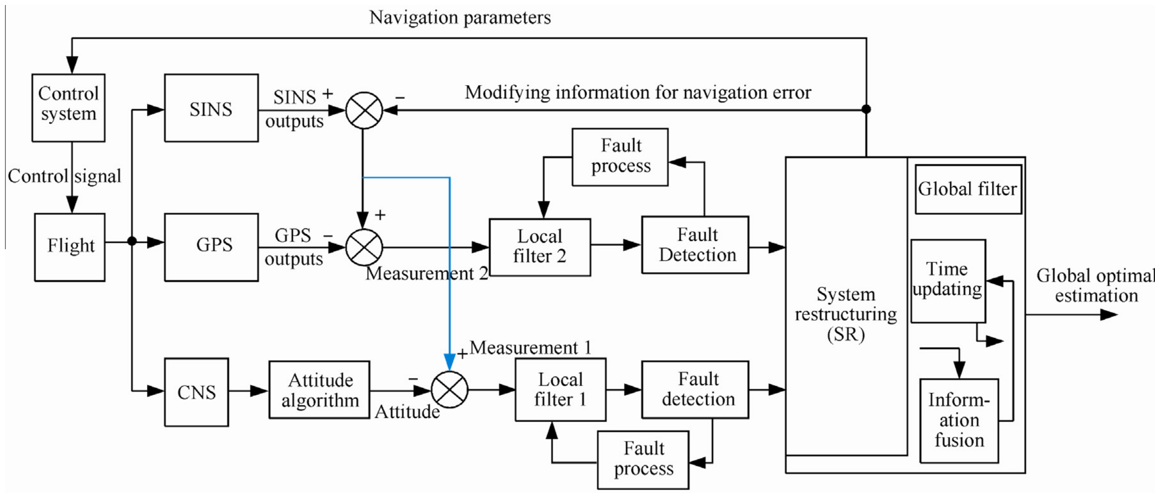

Fig. 1 SINS/GPS/CNS fault-tolerant system structure.

3.2. Fault detection model establishing

Within the framework of BRB, this section studies the modeling of the fault detection for INS/GPS/CNS integration navigation system.

For filter algorithm of navigation system, the fault detection is implemented by monitoring residual $\boldsymbol{r}$ and relevant values to $\boldsymbol{r}$ in general, as in the cases of Ref. [26] that detects system gradual fault by gathering residual features and Ref. [27] that adjusts filter gain through residual and effectively kicks off system fault. The standard chi-square test method constructs analytical fault detection function (FDF) also through residual. In fact, when fault happens in the assistant navigation system, not only residual $\boldsymbol{r}$ has a gradual or sudden changing tendency, but $\dot{\boldsymbol{r}}$ also has a certain changing tendency.[28] Therefore, the measuring residual and its changing rate are used as the input attribute parameters of the fault detection model in this paper, namely $\boldsymbol{x}(t) = [\boldsymbol{r}(t) \quad \dot{\boldsymbol{r}}(t)]$. And the fault detection value of residual test method based on analytical model is used as the model output. Considering the detection ability of analytical method is susceptible to system model, the FDF in non-analytical form is defined as

$$y(t) = f(\boldsymbol{x}(t)) = f(\boldsymbol{r}(t), \dot{\boldsymbol{r}}(t)) \tag{10}$$

One crucial problem needing to be solved is how to obtain the training data $y(t)$ at the initial implementation of the algorithm. Since $y(t)$ corresponds to the FDF value of the residual test method, this paper uses the relatively accurate output of residual test method as the training data. In the calculating process, the relatively accurate residual and its variance information need to be known in advance, and this is based on the premise that the prediction state variables are accurate. Different schemes corresponding to different cases should be adopted to satisfy the premise in the experiment. Three schemes can be used to obtain the training data $y(t)$ in different contexts as follows:

Scheme 1
In the simulation, it can use a period of the standard state variables obtained by the pre-established flying trajectory to calculate the accurate residual and its variance, and then the results are adopted to get the training data $y(t)$.

Scheme 2
In the actual flying test, it can use the history flying trajectory data to calculate the accurate residual and its variance, and then to get the training data $y(t)$.

Scheme 3
In the actual flying test, in view of the short-term high-precision characteristic of SINS, its short-term high precision state information can be used to calculate the accurate residual and its variance, and furthermore to get the training data $y(t)$.

In practice, the selection of the above schemes is context-dependent. Since the algorithms are tested by the simulation, this paper adopts Scheme 1.

The fault detection model based on BRB can then be established as the specific procedures going like below

Step 1 Set up input evaluation framework of attributes. Since both $\boldsymbol{r}$ and $\dot{\boldsymbol{r}}$ are numerical data, it is necessary to transform them to belief distribution structure using BRB reasoning approach. With the traits of measurements in fault and normal states, the language grade intuitively compartmentalized in the same way is introduced to represent numerical characteristics of attributes: S—small, M—moderate and B—big. In this way, the input attribute evaluation framework shall be defined as A = (S, M, B).

Step 2 Achieve equivalent transformation of belief distribution. Quantified referential points corresponding to evaluation framework are selected in accordance with data distribution characteristics of the input attributes $\boldsymbol{r}$ and $\dot{\boldsymbol{r}}$. All input attributes may then be assessed with reference to this framework using the rule-based information transformation technique.[29]

Step 3 Define discernment framework of the detected output. To evaluate the local system fault state, the navigation system state is divided into two categories: normal and fault. Of course, we can select more state and different divisions. In order to simplify problem, this paper only takes into consideration the above states. Hence, define the following discernment framework of navigation system state:

$$O(\hat{y}) = \{(F_1, \hat{\beta}_1), (F_2, \hat{\beta}_2)\} = \{(\text{Normal}, \hat{\beta}_1), (\text{Fault}, \hat{\beta}_1)\} \quad (11)$$

Step 4 Establish the initial BRB of the detection model. In fact, the attributes $\boldsymbol{r}$ and $\dot{\boldsymbol{r}}$ are zero mean Gauss white noise in normal state, but such characteristic does not exist under fault state. The bigger $\boldsymbol{r}$ and $\dot{\boldsymbol{r}}$ are, the more distinct the fault appears and the higher the degree of belief distributed to fault state is. Therefore, the initial BRB for fault detection of navigation system can be established as Table 1. The belief rules in Table 1 reflect the causality between the input attributes and the fault state of navigation system. Take the example of the Rule 9, which means if $\boldsymbol{r}$ and $\dot{\boldsymbol{r}}$ are big at current time instant, then the belief given to the fault state of navigation system is 1, namely complete fault; while the rule weight is 1, namely believing the rule completely. Other rules can be explained similarly.

Step 5 Update model parameters and calculate detection function. As explained before, obtaining the detection function value through model is on the premise that the model parameters, including θ_i ($i = 1, 2, \ldots, L$), δ_m($m = 1, 2, \ldots, p$) and u_j ($j = 1, 2, \ldots, N$), are known already. Step 4 endows parameters initial values, and the starting value of the model, whereas the actual values shall be determined by an optimism algorithm. This paper proposes an online updating algorithm to determine parameters and the specific steps are elaborated in the following section. After knowing the parameters, the detection function value turns available by Eqs. (5)–(9).

3.3. Criterion of fault discrimination

During fault detection, detection function value fluctuates in a range normally, and there is a reference benchmark $\bar{y}$ when the state is normal. Therefore, the fault occurrence can be defined by detecting the distance between detection function value $\hat{y}(t)$ and the reference benchmark $\bar{y}$.

Definition 1. As for the parameter $y(t)$ describing the fault feature of navigation system, $\hat{y}(t)$ denotes its estimation value calculated by BRB model at the time instant t, and $\bar{y}$ its reference benchmark under normal state obtained from prior knowledge. When Eq. (12) is valid, the complete fault of the system can be confirmed:

$$d_t = |\hat{y}(t) - \bar{y}| \geqslant \text{Thr} \quad (12)$$

where Thr is the threshold given previously and d_t the distance.

Table 1 Initial BRB for fault detection of navigation system.

Rule serial number	$\boldsymbol{r}$	$\dot{\boldsymbol{r}}$	$[\beta_{1,k}\quad \beta_{2,k}]$	Rule weight
1	S	S	[0.95 0.05]	1.00
2	S	M	[0.78 0.22]	0.98
3	S	B	[0.66 0.34]	1.00
4	M	S	[0.55 0.45]	0.99
5	M	M	[0.50 0.50]	1.00
6	M	B	[0.36 0.64]	0.85
7	B	S	[0.24 0.76]	1.00
8	B	M	[0.10 0.90]	0.90
9	B	B	[0 1.00]	1.00

4. Parameter updating recursive algorithm of fault detection model

If the parameters of the detection model are not given a priori or only known partially or imprecisely, the model accuracy will be affected dramatically. To solve this problem, Liu et al.[12,13] established BRB nonlinear optimization model based on minimum average variance method. But this method is an offline type in essence, and its tracking ability will get degraded if the system state changes. Regarding the navigation system reality, it is necessary to develop a method that can optimally estimate BRB parameter and track the system change timely and quickly. Therefore, the paper proposes a BRB parameter updating recursive algorithm.

Assume that the input/output data pairs ($\boldsymbol{x}(n)$,$y(n)$) of the detection model are known (for description convenience, t appearing before is replaced by n here), the history input/output data pairs ($\boldsymbol{x}(1)$,$y(1)$), ($\boldsymbol{x}(2)$,$y(2)$), ..., ($\boldsymbol{x}(n)$,$y(n)$) must be accounted because this approach applies mathematical statistics method to estimate parameters. $y(n)$ is belief distribution form or numerical and the latter is preferred considering the study background of this paper. Firstly, a theorem gives the parameters updating recursive equation directly and then followed by its detailed demonstration and derivation.

Theorem 1. *Assume that the probability density function (PDF) of $y(n)$ is $f(y(n)|\,\boldsymbol{x}(n),\boldsymbol{Q})$, where $\boldsymbol{Q}$ refers to unknown model parameter vector to be evaluated. And assume that the BRB inputs $\boldsymbol{x}(1)$,$\boldsymbol{x}(2)$,..., $\boldsymbol{x}(n)$ are relatively independent and so are the outputs $y(1)$,$y(2)$,...,$y(n)$, then the below recursive equation is valid:*

$$\boldsymbol{Q}(n+1) = \prod_H \left\{ \boldsymbol{Q}(n) + \frac{1}{n}[\boldsymbol{\Xi}(\boldsymbol{Q}(n))]^{-1}\boldsymbol{\Gamma}(\boldsymbol{Q}(n)) \right\} \quad (13)$$

where

$$\boldsymbol{\Gamma}(\boldsymbol{Q}(n)) = \nabla_Q \lg f(y(n)|\boldsymbol{x}(n), \boldsymbol{Q}(n)) \quad (14)$$

$$\boldsymbol{\Xi}(\boldsymbol{Q}(n)) = E\{-\nabla_Q \nabla_Q^{\mathrm{T}} \lg f(y(n)|\boldsymbol{x}(n), \boldsymbol{Q})|\boldsymbol{x}(n), \boldsymbol{Q}(n)\} \quad (15)$$

where ∇_Q is column gradient operator related to $\boldsymbol{Q}$, $\prod_H\{\cdot\}$ the projection onto the constraint condition H, $E\{\cdot|\cdot\}$ conditional expectation and $\boldsymbol{\Xi}(\boldsymbol{Q}(n))$ is the augmented information matrix calculated at $\boldsymbol{Q}(n)$. Besides, the theorem gives the general expression to solve model parameters. What is included in $\boldsymbol{Q}$ is up to the actual studied subject. As to this paper, $\boldsymbol{Q}$ consists of three parts, $\boldsymbol{V} = [\theta_i\ \delta_m]^T$, $\boldsymbol{u} = [u_j]^T$ and other PDF parameter $\boldsymbol{\sigma}$ of $y(n)$.

Proof. According to the independent assumption, after knowing the history input/output data pairs, we can obtain directly

$$f(y(1), y(2), \ldots, y(n)|\boldsymbol{x}(1), \boldsymbol{x}(2), \ldots, \boldsymbol{x}(n), \boldsymbol{Q}) = \prod_{\tau=1}^{n} f(y(\tau)|\boldsymbol{x}(\tau), \boldsymbol{Q}) \quad (16)$$

From Eq. (16), the expectation of log-likelihood function at time instant n is defined as

$$L_{n+1}(\boldsymbol{Q}) = E\left\{ \lg \prod_{\tau=1}^{n} f(y(\tau)|\boldsymbol{x}(\tau), \boldsymbol{Q})|\boldsymbol{x}(1), \boldsymbol{x}(2), \ldots, \boldsymbol{x}(n), \boldsymbol{Q}(n) \right\}$$
$$= E\left\{ \sum_{\tau=1}^{n} \lg f(y(\tau)|\boldsymbol{x}(\tau), \boldsymbol{Q})|\boldsymbol{x}(1), \boldsymbol{x}(2), \ldots, \boldsymbol{x}(n), \boldsymbol{Q}(n) \right\} \quad (17)$$

Then Eq. (17) can further be written as

$$L_{n+1}(\boldsymbol{Q}) = E\left\{\sum_{\tau=1}^{n-1} \lg f(y(\tau)|\boldsymbol{x}(\tau),\boldsymbol{Q})|\boldsymbol{x}(1),\boldsymbol{x}(2),\ldots,\boldsymbol{x}(n-1),\right.$$
$$\boldsymbol{Q}(n-1)\} + E\{\lg f(y(n)|\boldsymbol{x}(n),\boldsymbol{Q})|\boldsymbol{x}(n),\boldsymbol{Q}(n)\} = L_n(\boldsymbol{Q})$$
$$\left. + E\{\lg f(y(n)|\boldsymbol{x}(n),\boldsymbol{Q})|\boldsymbol{x}(n),\boldsymbol{Q}(n)\right\} \tag{18}$$

To obtain the expression of $L_{n+1}(\boldsymbol{Q})$, we consider the Talyor expansion of the first term on the right-hand side of Eq. (18) and get

$$L_n(\boldsymbol{Q}) = L_n(\boldsymbol{Q}(n)) + [\nabla_Q L_n(\boldsymbol{Q}(n))](\boldsymbol{Q}-\boldsymbol{Q}(n)) + \frac{1}{2}(\boldsymbol{Q} - \boldsymbol{Q}(n))^{\mathrm{T}}\left[\nabla_Q\nabla_Q^{\mathrm{T}} L_n(\boldsymbol{Q}(n))\right](\boldsymbol{Q}-\boldsymbol{Q}(n)) \tag{19}$$

By defining $L_n(\boldsymbol{Q}), \nabla_Q\nabla_Q^{\mathrm{T}} L_n(\boldsymbol{Q}(n))$ can be given by Refs. [30,31]

$$\nabla_Q\nabla_Q^{\mathrm{T}} L_n(\boldsymbol{Q}(n)) \approx -(n-1)\boldsymbol{\Xi}(\boldsymbol{Q}(n)) \tag{20}$$

where $\boldsymbol{\Xi}(\boldsymbol{Q}(n)) = E\left\{-\nabla_Q\nabla_Q^{\mathrm{T}} \lg f(y(n) \mid \boldsymbol{x}(n), Q) \mid \boldsymbol{x}(n), \boldsymbol{Q}(n)\right\}$.

Because $\boldsymbol{Q}(n)$ in Eq. (19) is the maximum point of $L_n(\boldsymbol{Q})$, there is

$$\nabla_Q L_n(\boldsymbol{Q}(n)) = \boldsymbol{0} \tag{21}$$

Substituting Eqs. (15) and (21) into Eq. (19), we can get

$$L_n(\boldsymbol{Q}) \approx L_n(\boldsymbol{Q}(n)) - \frac{1}{2}(\boldsymbol{Q}-\boldsymbol{Q}(n))^{\mathrm{T}}[(n-1)\boldsymbol{\Xi}(\boldsymbol{Q}(n))](\boldsymbol{Q}-\boldsymbol{Q}(n)) \tag{22}$$

Then taking Taylor expansion, there is

$$\lg f(y(n)|\boldsymbol{x}(n),\boldsymbol{Q}) \approx \lg f(y(n)|\boldsymbol{x}(n),\boldsymbol{Q}(n)) + (\nabla_Q \lg f(y(n)|\boldsymbol{x}(n), \times \boldsymbol{Q}(n)))(\boldsymbol{Q}-\boldsymbol{Q}(n)) + \frac{1}{2}(\boldsymbol{Q}-\boldsymbol{Q}(n))^{\mathrm{T}} \left(\nabla_Q\nabla_Q^{\mathrm{T}} \lg f(y(n)|\boldsymbol{x}(n),\boldsymbol{Q}(n))\right) \times (\boldsymbol{Q}-\boldsymbol{Q}(n)) \tag{23}$$

Define

$$\boldsymbol{\Gamma}(\boldsymbol{Q}(n)) = \nabla_Q \lg f(y(n)|\boldsymbol{x}(n),\boldsymbol{Q}(n))$$

where $\nabla_Q \lg f(y(n)|\boldsymbol{x}(n),\boldsymbol{Q}(n))$ represents the gradient vector at $\boldsymbol{Q}(n)$.

So the conditional expectation of Eq. (23) can be written as

$$E\{\lg f(y(n)|\boldsymbol{x}(n),\boldsymbol{Q})|\boldsymbol{x}(n),\boldsymbol{Q}(n)\} = E\{\lg f(y(n)|\boldsymbol{x}(n),\boldsymbol{Q}(n))|\boldsymbol{x}(n), \boldsymbol{Q}(n)\} + \boldsymbol{\Gamma}(\boldsymbol{Q}(n))(\boldsymbol{Q}-\boldsymbol{Q}(n)) + \frac{1}{2}(\boldsymbol{Q}-\boldsymbol{Q}(n))^{\mathrm{T}} \times E\left\{\nabla_Q\nabla_Q^{\mathrm{T}} \lg f(y(n)|\boldsymbol{x}(n), \boldsymbol{Q}(n))|\boldsymbol{x}(n),\boldsymbol{Q}(n)\right\}(\boldsymbol{Q}-\boldsymbol{Q}(n)) \tag{24}$$

Eqs. (18), (22) and (24) lead to the following expression:

$$L_{n+1}(\boldsymbol{Q}) = L_n(\boldsymbol{Q}(n)) + E\{\lg f(y(n)|\boldsymbol{x}(n),\boldsymbol{Q}(n))|\boldsymbol{x}(n),\boldsymbol{Q}(n)\} + \boldsymbol{\Gamma}(\boldsymbol{Q}(n))(\boldsymbol{Q}-\boldsymbol{Q}(n)) - \frac{n}{2}(\boldsymbol{Q}-\boldsymbol{Q}(n))^{\mathrm{T}} \times [\boldsymbol{\Xi}(\boldsymbol{Q}(n))](\boldsymbol{Q}-\boldsymbol{Q}(n)) \tag{25}$$

Similarly, since $\boldsymbol{Q}(n+1)$ is the maximum point of $L_{n+1}(\boldsymbol{Q})$ in Eq. (25) and the first and second terms of Eq. (25) are the constants, we obtain

$$\nabla_Q L_{n+1}(\boldsymbol{Q}(n+1)) = \boldsymbol{0} \tag{26}$$

So the maximizing parameter $\boldsymbol{Q}(n+1)$ is given by

$$\boldsymbol{Q}(n+1) = \boldsymbol{Q}(n) + \frac{1}{n}[\boldsymbol{\Xi}(\boldsymbol{Q}(n))]^{-1}\boldsymbol{\Gamma}(\boldsymbol{Q}(n)) \tag{27}$$

As to actual system, every parameter of $\boldsymbol{Q}$ must meet a certain constraint condition. The constraint conditions of parameters in this paper are

Constraint 1
Since the rule weights are normalized, there are

$$0 \leqslant \theta_i \leqslant 1 \quad (i = 1,2,\ldots,L) \tag{28}$$

Constraint 2
Since the relative weights of attributes are normalized, there are

$$0 \leqslant \delta_m \leqslant 1 \quad (m = 1,2,\ldots,p) \tag{29}$$

Constraint 3
Utility value shall meet the following conditions:

$$0 \leqslant u_j \quad (j = 1,2,\ldots,N) \tag{30}$$

And if F_j is preferred to F_i, then

$$u_i < u_j \tag{31}$$

Under the condition of the above constraints, Eq. (27) is revised as follows:

$$\boldsymbol{Q}(n+1) = \prod_H \left\{\boldsymbol{Q}(n) + \frac{1}{n}[\boldsymbol{\Xi}(\boldsymbol{Q}(n))]^{-1}\boldsymbol{\Gamma}(\boldsymbol{Q}(n))\right\}$$

where $\prod_H\{\cdot\}$ is the projection onto the constraint condition $\boldsymbol{H}$ composed by Eqs. (28)–(31). This completes the proof. □

Theorem 1 is the parameter updating recursive algorithm of the detection model. Knowing from the theorem proofing process, parameter updating recursive algorithm demands a known PDF of $y(n)$ previously. Next we study the realization of the algorithm with the assumption that $f(y(n)|\boldsymbol{x}(n),\boldsymbol{Q})$ is known.

In normal state, the estimated output value $\hat{y}(n)$ shall get close to actual $y(n)$ as much as possible giving $\boldsymbol{x}(n)$. $y(n)$ herein is viewed as a random variable, and $\hat{y}(n)$ denotes its expectation. So, assume that the PDF of $y(n)$ is Gaussian distribution as follows:

$$f(y(n)|\boldsymbol{x}(n),\boldsymbol{Q}) = \frac{1}{\sqrt{2\pi\sigma}} \exp\left\{-\frac{(y(n)-\hat{y}(n))^2}{2\sigma}\right\} \tag{32}$$

where $\boldsymbol{Q} = [\boldsymbol{S}^{\mathrm{T}}\ \sigma]^{\mathrm{T}}$ is a parameter vector, σ variance, and $\boldsymbol{S} = [\boldsymbol{V}^{\mathrm{T}}\ \boldsymbol{u}^{\mathrm{T}}]^{\mathrm{T}} = [S_1\ S_2 \ldots\ S_{L+p+N}]^{\mathrm{T}}$.

Due to the independence between the elements of $\boldsymbol{S}$ and, $\boldsymbol{\Xi}(\boldsymbol{Q}(n))$ and $\boldsymbol{\Gamma}(\boldsymbol{Q}(n))$ in Eq. (13) can be expressed as

$$\boldsymbol{\Xi}(\boldsymbol{Q}(n)) = \begin{bmatrix} \boldsymbol{\Xi}'(\boldsymbol{Q}(n)) & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{\Xi}''(\boldsymbol{Q}(n)) \end{bmatrix} \tag{33}$$

$$\boldsymbol{\Gamma}(\boldsymbol{Q}(n)) = [\boldsymbol{\Gamma}'(\boldsymbol{Q}(n))^{\mathrm{T}} \quad \boldsymbol{\Gamma}''(\boldsymbol{Q}(n))^{\mathrm{T}}]^{\mathrm{T}} \tag{34}$$

where $\boldsymbol{\Xi}'(\boldsymbol{Q}(n))$ and $\boldsymbol{\Gamma}'(\boldsymbol{Q}(n))$ are the first-order derivatives with respect to $\boldsymbol{S}$, and $\boldsymbol{\Xi}''(\boldsymbol{Q}(n))$ and $\boldsymbol{\Gamma}''(\boldsymbol{Q}(n))$ the second order derivatives with respect to σ. Obviously, there is

$$[\boldsymbol{\Xi}(\boldsymbol{Q}(n))]^{-1} = \begin{bmatrix} [\boldsymbol{\Xi}'(\boldsymbol{Q}(n))]^{-1} & \boldsymbol{0} \\ \boldsymbol{0} & [\boldsymbol{\Xi}''(\boldsymbol{Q}(n))]^{-1} \end{bmatrix} \tag{35}$$

Considering $\boldsymbol{S}$ only, the parameter recursive algorithm Eqs. (27) and (13) can be transformed into the following form by Eqs. (34) and (35):

$$\boldsymbol{S}(n+1) = \boldsymbol{S}(n) + \frac{1}{n}[\boldsymbol{\Xi}'(\boldsymbol{Q}(n))]^{-1}\boldsymbol{\Gamma}'(\boldsymbol{Q}(n)) \tag{36}$$

$$\boldsymbol{S}(n+1) = \prod_H \left\{ \boldsymbol{S}(n) + \frac{1}{n}[\boldsymbol{\Xi}'(\boldsymbol{Q}(n))]^{-1}\boldsymbol{\Gamma}'(\boldsymbol{Q}(n)) \right\} \tag{37}$$

In Eq. (37), $\boldsymbol{S}(n)$ is known. The parameter recursive calculation can be completed as long as $\boldsymbol{\Xi}'(\boldsymbol{Q}(n))$ and $\boldsymbol{\Gamma}'(\boldsymbol{Q}(n))$ are known. Their expressions are as follows:

(1) If $a,b = 1, 2, \ldots, L + p$, the ath element of $\boldsymbol{\Gamma}'(\boldsymbol{Q}(n))$ and the entries of $\boldsymbol{\Xi}'(\boldsymbol{Q}(n))$ at time instant n are

$$[\boldsymbol{\Gamma}'(\boldsymbol{Q}(n))]_a = \frac{(y(n) - \hat{y}(n))}{\sigma(n)} \sum_{j=1}^{N} \mu_j(n) \frac{\partial \hat{\beta}_j(n)}{\partial S_a} \bigg|_{\boldsymbol{S}=\boldsymbol{S}(n)} \tag{38}$$

$$\begin{aligned} [\boldsymbol{\Xi}'(\boldsymbol{Q}(n))]_{a,b} &= E\Big\{ \tfrac{1}{\sigma} \tfrac{\partial \hat{y}(n)}{\partial S_a} \tfrac{\partial \hat{y}(n)}{\partial S_b} - \tfrac{1}{\sigma} \tfrac{\partial^2 \hat{y}(n)}{\partial S_a \partial S_b} \\ &\times (y(n) - \hat{y}(n)) | \boldsymbol{Q}(n) \Big\} \\ &= \tfrac{1}{\sigma(n)} \left[\sum_{j=1}^{N} \mu_j(n) \tfrac{\partial \hat{\beta}_j(n)}{\partial S_a} \right] \left[\sum_{j=1}^{N} \mu_j(n) \tfrac{\partial \hat{\beta}_j(n)}{\partial S_b} \right] \bigg|_{\boldsymbol{S}=\boldsymbol{S}(n)} \end{aligned} \tag{39}$$

(2) If $a,b = L + p + 1, \ldots, L + p + N$, there are

$$[\boldsymbol{\Gamma}'(\boldsymbol{Q}(n))]_a = \frac{\hat{\beta}_{a-L-p}(n)(y(n) - \hat{y}(n))}{\sigma(n)} \bigg|_{\boldsymbol{S}=\boldsymbol{S}(n)} \tag{40}$$

$$[\boldsymbol{\Xi}'(\boldsymbol{Q}(n))]_{a,b} = \frac{\hat{\beta}_{a-L-p}(n)\hat{\beta}_{b-L-p}(n)}{\sigma(n)} \bigg|_{\boldsymbol{S}=\boldsymbol{S}(n)} \tag{41}$$

In Eqs. (38) and (39), $\partial \hat{\beta}_j / \partial S_a$ needs to be calculated, so it is given in the following text.

Assume

$$B_j = \prod_{k=1}^{L} \left(w_k \beta_{j,k} + 1 - w_k \sum_{i=1}^{N} \beta_{i,k} \right) - \prod_{k=1}^{L} \left(1 - w_k \sum_{i=1}^{N} \beta_{i,k} \right) \tag{42}$$

$$\begin{aligned} C = &\sum_{j=1}^{N} \prod_{k=1}^{L} \left(w_k \beta_{j,k} + 1 - w_k \sum_{i=1}^{N} \beta_{i,k} \right) \\ &- (N-1) \prod_{k=1}^{L} \left(1 - w_k \sum_{i=1}^{N} \beta_{i,k} \right) - \prod_{k=1}^{L} (1 - w_k) \end{aligned} \tag{43}$$

Then there is

$$\hat{\beta}_j = B_j / C \tag{44}$$

Define

$$\xi_1(q) = \prod_{\substack{k=1 \\ k \neq q}}^{L} \left(1 - w_k \sum_{i=1}^{N} \beta_{i,k} \right) \tag{45}$$

$$\chi_1(q,j) = \prod_{\substack{k=1 \\ k \neq q}}^{L} \left(w_k \beta_{j,k} + 1 - w_k \sum_{i=1}^{N} \beta_{i,k} \right) \tag{46}$$

The first-order derivatives of $\hat{\beta}_j (j = 1, 2, \ldots, N)$ with respect to $S_a (a = 1, 2, \ldots, L + p)$ are represented as

$$\frac{\partial \hat{\beta}_j}{\partial S_a} = \frac{1}{C^2} \left(\frac{\partial B_j}{\partial S_a} C - \frac{\partial C}{\partial S_a} B_j \right) \tag{47}$$

$$\frac{\partial B_j}{\partial w_q} = \left(\beta_{j,q} - \sum_{i=1}^{N} \beta_{i,q} \right) \chi_1(q,j) + \xi_1(q) \sum_{i=1}^{N} \beta_{i,q} \tag{48}$$

$$\begin{aligned} \frac{\partial C}{\partial w_q} = &\sum_{j=1}^{N} \left(\beta_{j,q} - \sum_{i=1}^{N} \beta_{i,q} \right) \chi_1(q,j) + (N-1) \xi_1(q) \sum_{i=1}^{N} \beta_{i,q} \\ &+ \prod_{\substack{k=1 \\ k \neq q}}^{L} (1 - \omega_k) \end{aligned} \tag{49}$$

$$\frac{\partial w_q}{\partial \theta_q} = \frac{\left(\prod_{m=1}^{p} (\alpha_m^q)^{\delta_m} \right) \left(\sum_{\substack{l=1 \\ l \neq q}}^{L} \theta_l \prod_{m=1}^{p} (\alpha_m^l)^{\delta_m} \right)}{\left(\sum_{l=1}^{L} \theta_l \prod_{m=1}^{p} (\alpha_m^l)^{\delta_m} \right)^2} \tag{50}$$

$$\frac{\partial w_q}{\partial \delta_c} = \frac{\sum_{\substack{l=1 \\ l \neq c}}^{L} \left[\left(\theta_q \theta_l \prod_{m=1}^{p} (\alpha_m^l)^{\delta_m} \right) \left(\ln (\alpha_c^q) - \ln (\alpha_c^l) \right) \right]}{\left(\sum_{l=1}^{L} \theta_l \prod_{m=1}^{p} (\alpha_m^l)^{\delta_m} \right)^2} \tag{51}$$

where $j = 1, 2, \ldots, N$; $q = 1, 2, \ldots, L$; $c = 1, 2, \ldots, p$.

Besides, $\sigma(n)$ also needs to be calculated, and it can be estimated through

$$\begin{aligned} \sigma(n) &= \arg\max_{\sigma} \lg f(y(n)|x(n), \boldsymbol{Q})|_{\boldsymbol{S}=\boldsymbol{S}(n)} \\ &= (y(n) - \hat{y}(n))^2|_{\boldsymbol{S}=\boldsymbol{S}(n)} \end{aligned} \tag{52}$$

Then the recursive algorithm with the constraints shown in Eq. (37) should be given in detail. First, Constraints 1 and 2 can be replaced as

$$0 \leqslant S_d \leqslant 1 \quad (d = 1, 2, \ldots, L + p) \tag{53}$$

Constraint 3 is equivalent to

$$\begin{aligned} &h_g(S_{L+p+i}, S_{L+p+j}) = \mu_i - \mu_j < 0 \\ &(i = 1, 2, \ldots, N-1; j = i+1, i+2, \ldots, N) \end{aligned} \tag{54}$$

where $g = (i-1)(N-1) - \sum_{k=1}^{i-2} (i-k-1) + j - i$.

The constraints given by Eq. (53) require that the parameters S_d ($d = 1, 2, \ldots, L + p$) derived from Eq. (36) have up and down limits. Using the projection algorithm, we define the projecting operator as

$$\boldsymbol{\pi}_1(\boldsymbol{S}(n+1)) = \sum_{d=1}^{L+p} \widehat{S}_d(n+1) \boldsymbol{e}_d \tag{55}$$

where $\boldsymbol{e}_d$ is a vector whose dth element is 1 and the other elements are 0, $\widehat{S}_d(n+1)$ is obtained from

$$\widehat{S}_d(n+1) = \begin{cases} 0, & S_d(n+1) < 0 \\ 1, & S_d(n+1) > 1 \\ S_d(n+1), & 0 \leqslant S_d(n+1) \leqslant 1 \end{cases} \tag{56}$$

For constraints given in Eq. (54), let

$$h(S) = [h_1(S) \quad h_2(S) \quad \ldots \quad h_G(S)]^{\mathrm{T}}$$

where $G = (N-1)(N-2) - \sum_{k=1}^{N-3} (N-2-k) + 1$. Suppose that $\boldsymbol{H}(\boldsymbol{S})$ denotes the Jacobian matrix of $\boldsymbol{h}(\boldsymbol{S})$, then we deal with the constraints given by Eq. (54) applying the following algorithm:[30]

$$\boldsymbol{\Phi}(S(n)) = I_N - H(S(n))^{\mathrm{T}} \times (H(S(n))H(S(n))^{\mathrm{T}})^{-1}H(S(n)) \quad (57)$$

where $\boldsymbol{I}_N$ is the identity matrix whose dimension is N.

In addition, it is possible for BRB detection system that only some belief rules are activated at time instant n, which results in the parameters updated partially. This may make the matrix $\boldsymbol{\Xi}'(\boldsymbol{Q}(n))$ be singular, and it needs an amendment.[10] Hence, the revision coefficient $\vartheta \geqslant 0$ is introduced to ensure $\boldsymbol{\Xi}'(\boldsymbol{Q}(n))$ positive definiteness. Besides, the step factor $\zeta \geqslant 1$ is adopted to improve the algorithm's convergence rate. At last, the parameter updating recursive algorithm of the detection system can be described as

$$S(n+1) = \boldsymbol{\pi}_1\{\overline{S}(n+1)\} \quad (58)$$

$$\overline{S}(n+1) = S(n) + \frac{\zeta}{n}\boldsymbol{\pi}_2\{S(n)\} \times [\boldsymbol{\Xi}'(Q(n)) + \vartheta I_{L+p+N}]^{-1}\boldsymbol{\Gamma}'(\boldsymbol{Q}(n)) \quad (59)$$

with

$$\boldsymbol{\pi}_1\{\overline{S}(n+1)\} = \sum_{d=1}^{L+p}\widehat{S}_d(n+1)e_d \quad (60)$$

$$\boldsymbol{\pi}_2\{S(n)\} = \begin{bmatrix} I_{L+p} & \mathbf{0}_{(L+p)\times N} \\ \mathbf{0}_{N\times(L+p)} & \boldsymbol{\Phi}(S(n)) \end{bmatrix} \quad (61)$$

where $\boldsymbol{I}_{L+p}$ and $\boldsymbol{I}_{L+p+N}$ are all identity matrixes with $L+p$ dimension and $L+p+N$ dimension respectively.

As a result, the procedure for the recursive algorithm for updating the BRB-based model based on the numerical output and normal distribution assumption may be summarized as the following steps:

Step 1 Given the initial values of the parameter vector $\boldsymbol{S}(0)$ and the variance $\sigma(0)$. $\boldsymbol{S}(0)$ must satisfy the constraints (53) and (54).

Step 2 When the observations $\boldsymbol{x}(0)$ and $y(0)$ are available, the recursive algorithm Eq. (58) is used to estimate $\boldsymbol{S}(1)$. Then $\sigma(1)$ is estimated using Eq. (52).

Step 3 When $\boldsymbol{x}(n)$,$y(n)$ and $\boldsymbol{S}(n)$ are available, $\sigma(n)$ is estimated using Eq. (52), and $\boldsymbol{S}(n+1)$ is obtained from Eq. (58).

Step 4 Once the BRB-based model is updated, $\hat{y}(n+1)$ is estimated by Eq. (9), and then the next updating step is performed.

Remark 1. Due to the proposed algorithms under the assumptions of Gaussian distributions, they converge for Ergodic assumptions.

Remark 2. The proposed recursive algorithms for updating BRB-based detection model are based on the stochastic approximation algorithm. In addition, the convergence theorem of the stochastic approximation algorithm has been proved by Kushner et al.[32] Based on Remarks 1 and 2, there is no need to analyze the convergence of the recursive algorithm again.

Comparison analysis of algorithms: the recursive algorithm is inferior to the residual test in terms of the calculation complexity, but the latter one is not so effective for the soft fault, and the proposed recursive algorithm just aims at solving this problem. Compared with the afore-mentioned multi-objective optimization learning method, the recursive algorithm has the following two advantages:

(1) Its calculation complexity is relatively lower and thus it fits to the online implementation. The multi-objective optimization learning method is based on the least mean squared error measure and the parameter estimation is achieved by the traditional nonlinear programming with an off-line nature. In addition, this method adopts offline data to train model parameters again and again, and then the trained model is used online. However, repetitious training implies completing multiple cycles of calculation until getting the parameters with satisfactory precision, so this process is rather time-consuming and its computing burden is difficult to be expressed as an analytic form. The proposed algorithm is a recursive method and has analytic expressions, so the algorithm can immediately update the parameters of BRB as long as the system receives new observations. That is to estimate model parameters in real time through Eq. (58), which ensures the speed of the algorithm and satisfies the real-time requirement in engineering application. This paper validates the above analysis through experiments and the simulation results, as shown in Section 5.
(2) The precision of the parameter estimation is relatively high. Indeed, the offline approach can make use of the acquired data to train the model parameter such that more precise model can be attained, yet this kind of high-precision model is concerned with the training data. Since the system studied here is a dynamic navigation one and the system is constantly affected by various uncertain disturbances, the model parameters have to be updated in real time, which cannot be fulfilled by offline approach. Therefore, as the system works, the estimation precision will deteriorate gradually. On the contrary, the online updating method can use the real-time data and update the parameter accordingly, making it as close to the real system as possible. It can be concluded that the online approach possesses better dynamic adaptability and the precision of the parameter estimation is relatively high. As far as we know, real-time estimation is desirable in practice so that the condition of a monitored system is repeatedly updated during its operation to ensure that the most recently calculated state value accurately reflects the current reality of the system.

5. Navigation system fault detection experiment

5.1. Experiment scheme design

To validate the proposed detection method, this section develops an experimental study. Since it is hard to obtain the SINS/GPS/CNS integrated navigation data in high dynamic conditions through outfield test, we collect the data by hardware-in-the-loop simulation. The field data collection process is illustrated in Fig. 2. A standard flying trajectory is set previously. Then NS300 satellite signal simulator is used to generate the high dynamic GPS radio-frequency signal, and GN0204 receiver is used to receive satellite signal and process measurement information about pseudo-range and pseudo-range rate. SINS and CNS data are generated by the signal simulators designed in accordance with the flying trajectory. Finally, all generated measurement information is transferred to the information fusion and fault detection device for verifying the proposed method.

In addition, we compare our method with the residual chi-square test, the offline optimal training method proposed in Ref. [13] based on nonlinear multi-target optimism and the fuzzy neural network (FNN) method proposed in Ref. [33] The detailed parameters are set as follows.

(1) Setting of flying trajectory: Aircraft initial position is at 34.3441° north latitude, 108.7346° north longitude, 20.5 km above sea level; initial velocity and attitude are both zero; initial director is north straight; flight process is demonstrated in Table 2.
(2) Sensor precision grade: Gyro random drift rate is 0.01 (°)/h, accelerometer measured constant drift 30 μg, star sensor precision 10″, GPS receiver position precision 10 m, and the measured velocity precision 0.1 m/s.
(3) Fault simulation and parameter setting: To verify the proposed online fault detection method towards sudden fault as well as gradual fault, fault information is set artificially by adding sudden and gradual faults into measure information. Fault information is set up as Table 3. False alarm rate is $\alpha = 0.01$, and threshold corresponding to SINS/GPS system is 16.812.
(4) Initial setting of model parameters: Initial rule weights and the corresponding output belief are given in Table 1. The initial attribute weights are $\delta_1 = \delta_2 = 1$; the step factor $\zeta = 300$ and the singular matrix revision coefficient $\vartheta = 0.15$; the initial utility $[u_1\ u_2] = [0.94\ 0.66]$.

5.2. Experimental result and analysis

Based on the above simulation conditions, in ideal case, the system FDF value is shown in Fig. 3. The horizontal dot dash line describes the threshold value in Fig. 3. From Fig. 3, the FDF value does not surpass threshold at 261 s, but surpass threshold when the gradual fault lasting for 7 s or so. This happens because the fault degree is equivalent to noise degree in the initial period. In fact, the FDF value is about 2 m calculated with the set gradual-changing rate at 268 s.

Fig. 4 shows the FDF value obtained by residual chi-square method and Fig. 5 is the corresponding position errors. From Fig. 4, it can be observed that the residual method is not sensitive to detect gradual fault, unable to provide reliable diagnosis result if the gradual fault exists. It can also be seen that false alarm appears in the period of 376–450 s after the gradual fault is gone. The reason is that the gradual fault which is not discovered in initial period pollutes the system estimation states to some extent, and thus a large error is generated subsequently causing the false alarm. This point can also be proved from the east-orientation position and north-orientation position errors shown in Fig. 5. The positioning result shows that during the existence of gradual fault and false alarm (about 260–450 s), position errors exceed normal range. Then, due to the adaptive adjustment of the filter, the system convergences back to normal range again. However, with regard to sudden fault, the residual method expresses nice diagnosis ability without delay or error detection occurring in experiment.

Fig. 6 shows the result generated by the offline optimal training method mentioned above. For the sake of

Table 2 Table of flight process.

Flight time interval (s)	State description and relevant parameters
0–100	Uniformly accelerated flight, accelerated velocity is 0.5g
101–140	Upward climb flight, pitch angel rate is 0.5 (°)/s
141–340	Uniform climb flight
341–380	Downward flight, pitch angle rate is −0.5 (°)/s
381–390	Roll flight, roll angle rate −1 (°)/s
391–612	Turning flight
613–622	Roll flight, roll angle rate 1 (°)/s
623–922	Uniform flight

Table 3 Setting of faults.

Navigation system	Flight time interval (s)	Fault mode	Measurement	Fault value
GPS	260–375	Gradual change	Satellite 2's pseudo-range	0.3 m/s
GPS	660–690	Sudden fault	Satellite 9's pseudo-range	60 m

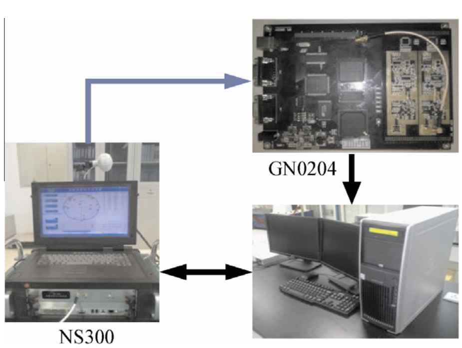

Fig. 2 Field data collection process.

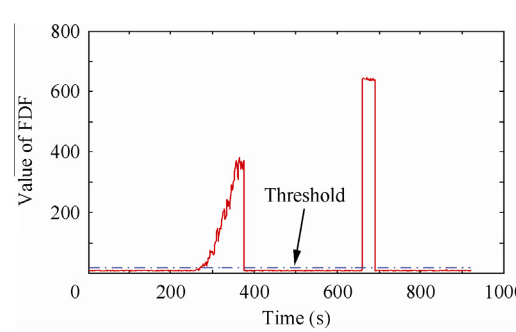

Fig. 3 Theoretical value of FDF.

comparison, the theoretical value is also showed in actual line while detection result in short dashed line in Fig. 6. Besides, two local amplificatory figures are illustrated in Fig. 6. In the simulated result, it is shown that this offline method is also quite effective to discover both gradual and sudden faults, but un-detection and false alarm exist. From the local amplificatory figures, we can see that the gradual fault is not detected until almost 30 s later. And at this moment, the fault value is about 9 m in the satellite pseudo-range direction, and the value is more than normal noise. Besides, there is an almost 20 s false alarm after the gradual fault disappears. Also, there is somewhat false alarm after the sudden fault disappears. In addition, during the period without any faults, FDF value with this offline method has an obvious error. Nevertheless, the detected value is within threshold, having no effect on the final discrimination of the fault.

The detection result using FNN is illustrated in Fig. 7, where the theory result is also compared with the real one. It can be observed from Fig. 7 that this approach can also detect the sudden and soft faults. However, since FNN is also an offline training approach but used online, missing detection and false alarm also exist in the detection. For example, the soft fault was not detected until 15 s later, and the false alarm occurred after both the end of soft and sudden faults. In addition, in local zooming part of the Fig. 7, at the end of the experiment, FNN also has some minor false alarm, which is caused by the dynamic response of the system. The dynamic variation of the navigation system decreases the approaching performance of the FNN with respect to the real system at the early stage, and makes the detection results imprecise. This indicates that in order to solve the fault detection problem for dynamic navigation systems, only prior knowledge and historical data are not enough, and the new observation information is also needed to constantly adjust the parameters of the detection model.

The result generated by the proposed online updating method is shown in Fig. 8. Similarly, we compare our method with the theoretical value. Also, the FDF values of gradual and sudden faults are amplified, corresponding to Zoom A and Zoom B, respectively. In Fig. 8, compared with theoretical value regarding gradual fault, the proposed method has a delay of 5 s which means a maximum 3.6 m fault value in the corresponding pseudo-range direction. Since the fault value of 3.6 m is approximately equivalent to the noise degree, the detection delay influencing position results does not exist basically. In addition, when the gradual fault disappears, the system can convergence back to normal range timely, and there is no false alarm. As to sudden fault, the proposed method is able to detect the fault timely and there is no false alarm after the fault disappears. Furthermore, during the period without any faults, the value generated by the proposed method goes well consistent with theoretical value. Table 4 shows some updated BRB parameters. We can see from Table 4 that every rule weight has been updated. The weight of rule 4 is only 0.0624, and it demonstrates that this conditional rule plays a smaller role in fault detection. What is more, the updated input attribute weights are $\delta_1 = 0.8676$, $\delta_2 = 0.5147$, respectively,

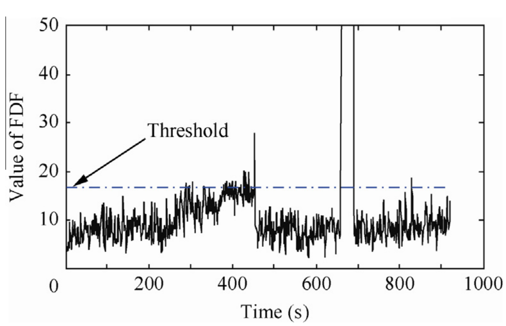

Fig. 4 FDF of residual chi-square method.

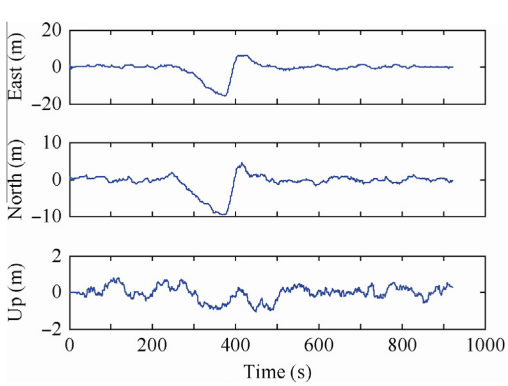

Fig. 5 Navigation system position errors using residual chi-square method.

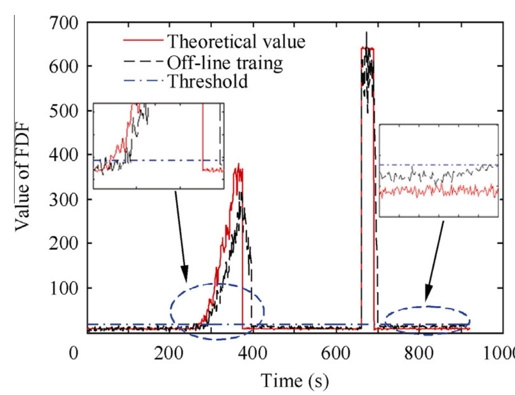

Fig. 6 FDF value of offline optimism study training method.

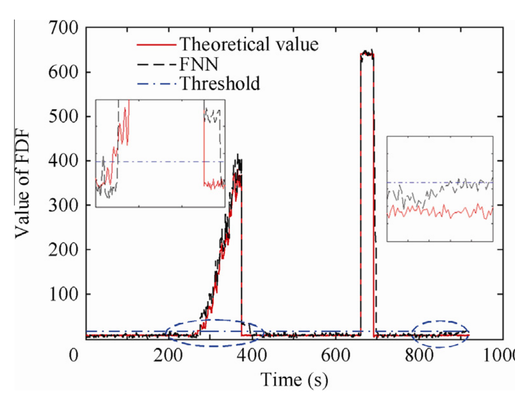

Fig. 7 FDF value of FNN method.

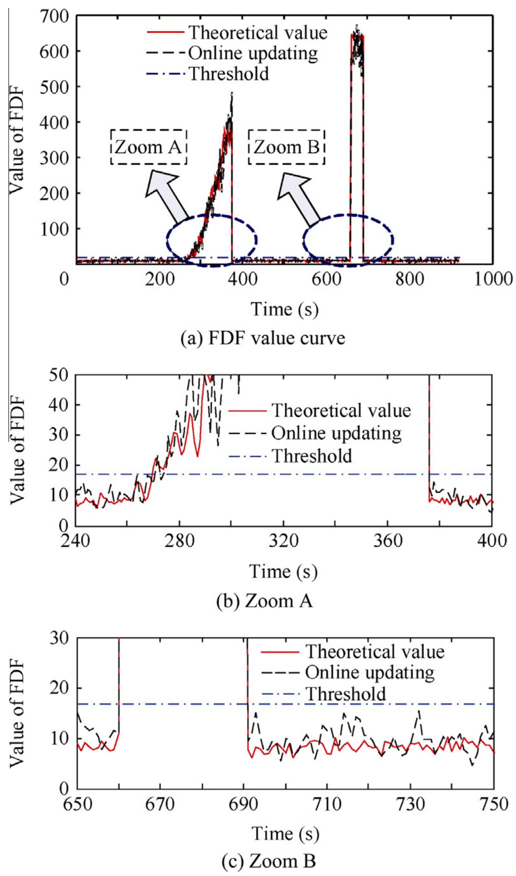

Fig. 8 FDF value of online updating method.

and this demonstrates that ***r*** plays a leading role in fault detection, which matches the reality as well.

To further illustrate the performance of the proposed recursive algorithm, the comparison of detection result and the time-consuming performance is conducted in Table 5, where the prediction precision, the fault detection performance and the operation effectiveness are measure by the root mean square error (RMSE), the time of false alarm and missing detection and the one step time-consuming. The "normal" refers to the case where there is no fault time period in the observation data, the "soft fault" and "sudden fault" refer to the corresponding fault time period, and the "all process" refers to the entire experiment period. It has to be noted that the time consumed here also considers the off-line training process, which is necessary for dynamic systems.

Table 5 indicates that the online updating method is superior to the residual one in terms of the estimation precision and the fault detection performance, though the residual method is a bit faster. This is because the analytical model of the residual uses the intermediate fertility variables. Nevertheless, the residual method is the least sensitive one with respect to the other three methods, and its fault detection performance is the worst. Since the latest observation information is used, the online updating method is also superior to the offline training and FNN approaches in terms of the estimation precision and fault detection. Compared with the aforementioned two algorithms, the time consumed by the online updating method is also the shortest when the time of parameter optimization is also considered, which satisfies the dynamic requirements of the navigation system. In addition, compared with offline training, FNN also has better fault-detection performance and efficiency. The main cause is that the optimization model is multi-objective optimization.[20] In reality, to solve multi-

Table 4 Updated BRB for fault detection of navigation.

Rule serial number	r	$\dot{r}$	$[\beta_{1,k}\quad \beta_{2,k}]$	Rule weight
1	S	S	[0.9984 0.0016]	1.0000
2	S	M	[0.8356 0.1644]	0.9887
3	S	B	[0.2987 0.7013]	0.9625
4	M	S	[0.7569 0.2431]	0.0624
5	M	M	[0.5362 0.4638]	0.8754
6	M	B	[0.3695 0.6305]	0.9168
7	B	S	[0.2263 0.7737]	1.0000
8	B	M	[0.0024 0.9976]	0.9994
9	B	B	[0.0001 0.9999]	1.0000

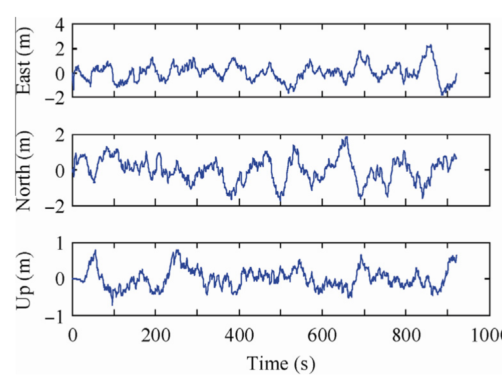

Fig. 9 Navigation system positioning errors using online updating method.

Table 5 Comparison of detection results under four methods.

Item		Residual test	Off-line training[13]	FNN[33]	Online updating
RMSE	Normal	3.5005	48.7655	36.0699	2.7520
	Soft fault	124.6764	42.9577	17.9069	14.7459
	Sudden fault	4.6224	42.9231	8.9509	5.5401
	All process	66.7060	53.4600	33.9767	6.0053
False detection time (s)	False alarm	17	29	24	3
	Missing detection	109	31	15	5
Updating/training time (s)		1.52×10^{-6}	0.80902	0.01542	0.00392

objective optimization problem is much more difficult than to solve single-objective optimization problem. Practically, it is very difficult to obtain the general optimal solution for multi-objective optimization problems.

Fig. 9 shows the final position errors corresponding to the proposed method. And the results further verify that the proposed online fault detection method correctly estimates fault function values, providing a reliable basis for disposing the fault and achieving smooth position errors with a precision of 2 m.

6. Conclusion

As an important guarantee to system reliability, the online fault detection technique of the integrated navigation system draws extensive attention. This paper investigates an online fault detection technique with the SINS/GPS/CNS integrated navigation platform. In particular, we study a fault detection method based on non-analytical model instead of the common analytical model which is generally unable to thoroughly solve the gradual and sudden fault. The navigation system fault detection model is established based on BRB. Compared with the traditional IF-THEN rule base, information included by BRB is richer in terms of knowledge description. To realize online fault detection and overcome the shortcomings of inadequate tracking and long-time consumption owned by the current BRB parameter offline optimal method, a parameter recursive updating algorithm based on EM is proposed, and the online parameter estimation of the fault detection system is realized. An experimental study for the aircraft navigation system fault detection is examined to demonstrate how the proposed detection method can be implemented. The results show that the solution is able to online update the detection model parameters effectively and the output value can well track the fault state of system. It is superior to offline optimal training method in terms of its detection performance (especially in the probability of false detection) and thus increases the reliability and positioning accuracy of the navigation system. It has to be noted that the proposed method has been verified very well using Scheme 1 in the hardware-in-the-loop experiment. Considering the actual flying test, the verification of the proposed method using Schemes 2 and 3 will also be studied in the future.

Acknowledgements

The authors would like to thank the anonymous reviewers for their critical and constructive comments that have considerably helped in the revision of an earlier version of the paper. The work was partially supported by the National High-tech Research and Development Program of China (No. 2011AA7053016) and National Natural Science Foundation of China (No. 61174030).

References

1. Lughofer ED, Klement EP, Lujan JM, Guardiola C. Model-based fault detection in multi-sensor measurement systems. In: *Proceedings of 2004 2nd International IEEE Conference on Intelligent Systems*. 2004. p. 184–9.
2. Si XS, Hu CH, Zhou ZJ. Fault prediction model based on evidential reasoning approach. *Sci China Ser F-Inf Sci* 2010;**53**(10):2032–46.
3. Li ZZ, Liu GH, Zhang R, Zhu ZC. Fault detection, identification and reconstruction for gyroscope in satellite based on independent component analysis. *Acta Astronaut* 2011;**68**(7-8):1015–23.
4. Han PX, Mu RJ, Cui NG. Effective fault diagnosis based on strong tracking UKF. *Aircraft Eng Aerosp Technol* 2011;**83**(5):275–82.
5. Rapoport I, Oshman Y. A Cramer-Rao-type estimation lower bound for system with measurement fault. *IEEE Trans Automatic Control* 2005;**50**(9):1234–45.
6. Li C, Qin HL, Tan ZZ. Intelligent fault-tolerant algorithm with two-stage and feedback for integrated navigation federated filtering. *J Syst Eng Electron* 2011;**22**(2):274–82.
7. Stach W, Kurgan L, Pedryca W. Numerical and linguistic prediction of time series with the use of fuzzy cognitive maps. *IEEE Trans Fuzzy Syst* 2008;**16**(1):61–72.
8. Si XS, Hu CH, Yang JB, Zhang Q. On the dynamic evidential reasoning algorithm for fault prediction. *Expert Syst Appl* 2011;**38**(5):5061–80.
9. Yang JB, Liu J, Wang J, Sii HSS, Wang HW. Belief rule-base inference methodology using the evidential reasoning approach-RIMER. *IEEE Trans Syst Man Cybern Part A: Syst Humans* 2006;**36**(2):266–85.
10. Zhou ZJ, Hu CH, Yang JB, Xu DL, Zhou DH. Online updating belief rule based system for pipeline leak detection under expert intervention. *Expert Syst Appl* 2009;**36**(4):7700–9.
11. Xu DL, Liu J, Yang JB, Liu GP, Wang J, Jenkinson I, et al. Inference and learning methodology of belief-rule-based expert system for pipeline leak detection. *Expert Syst Appl* 2007;**32**(1):103–13.
12. Liu J, Yang JB, Ruan D, Martinez L, Wang J. Self-tuning of fuzzy belief rule bases for engineering system safety analysis. *Ann Oper Res* 2008;**163**(1):143–68.
13. Yang JB, Liu J, Xu DL, Wang J, Wang HW. Optimal learning method for training belief rule based systems. *IEEE Trans Syst Man Cybern Part A: Syst Humans* 2007;**37**(4):569–85.
14. Huang CL, Yong K. *Multiple attribute decision making methods and applications: a state-of-art survey*. Berlin: Springer-Verlag; 1981.
15. Shafer G. *A mathematical theory of evidence*. Princeton: Princeton University Press; 1976.
16. Zadel LZ. Fuzzy sets. *Inf Control* 1965;**8**(3):338–53.
17. Yang JB, Xu DL. On the evidential reasoning algorithm for multiple attribute decision analysis under uncertainty. *IEEE Trans Syst Man and Cybern Part A: Syst Humans* 2002;**32**(3):289–304.
18. Si XS, Hu CH, Yang JB, Zhou ZJ. A new prediction model based on belief rule base for system's behavior prediction. *IEEE Trans Fuzzy Syst* 2011;**19**(4):456–71.
19. Hu CH, Si XS. Real-time parameters estimation of belief rule base health condition for inertial platform. *Acta Aeronautica et Astronautica Sinica* 2010;**31**(7):1454–65 [Chinese].
20. Hu CH, Si XS, Yang JB. Systems reliability forecasting based on evidential reasoning algorithm with nonlinear optimization. *Expert Syst Appl* 2010;**37**(3):2550–62.
21. Wang YM, Yang JB, Xu DL. Environmental impact assessment using the evidential reasoning approach. *Eur J Oper Res* 2006;**174**(3):1885–913.
22. Yang JB, Wang YM, Xu DL, Chin KS. The evidential reasoning approach for MADA under both probabilistic and fuzzy uncertainties. *Eur J Oper Res* 2006;**171**(1):309–43.
23. Yang F, Cheng C, Zhang GY, Cheng YM. Simulation research of sub-orbital vehicle re-entry flight segment navigation technology. *J Astronaut* 2010;**31**(3):729–33 [Chinese].

24. Gao YG, Wang SC, Liu ZG, Zhao X. Fault detection method based on LS-SVM for federated Kalman filter. *Control Decision* 2011;**26**(9):1–4 [Chinese].
25. Zhao X, Wang SC, Guo Y, Zhang JS, Liu ZG. SINS/SAR/GPS navigation system intelligent fault diagnosis method. In: *2011 2nd International conference on Intelligent Control and Information Processing (ICICIP 2011)*; 2011; p. 754–9.
26. Duan ZH, Cai ZX, Yu JX. An adaptive particle filter for soft fault compensation of mobile robots. *Sci China, Ser F: Inf Sci* 2008;**51**(12):2033–46.
27. Wang GD, Zhang SK, Yang RL. An adaptive outlier algorithm based on Kalman filtering for Beidou satellite passive combination navigation. *J Electron Inf Technol* 2008;**30**(8):1982–5 [Chinese].
28. Wang SC, Zhao X, Min HB, Liu ZG, Si XS. Study on an intelligent fault-tolerant technique for multiple satellite configured navigation under highly dynamic conditions. *Sci China Inf Sci* 2011;**54**(3):529–41.
29. Yang JB. Rule and utility based evidential reasoning approach for multi-attribute decision analysis under uncertainties. *Eur J Oper Res* 2001;**131**(1):31–61.
30. Chung PJ, Bohme JF. Recursive EM and SAGE-inspired algorithms with application to DOA estimation. *IEEE Trans Signal Process* 2005;**53**(8):2664–77.
31. Zhou ZJ, Hu CH, Yang JB, Xu DL, Zhou DH. Online updating belief-rule-based systems for pipeline leak detection under expert intervention. *Expert Syst Appl* 2009;**36**(4):7700–9.
32. Kushner HJ, Lakshmivarahan S. Numerical studies of stochastic approximation procedures for constrained problems. *IEEE Trans Automatic Control* 1977;**22**(3):428–39.
33. Ayoubi M, Isermann R. Neuro-fuzzy system for diagnosis. *Fuzzy Sets Syst* 1997;**89**(3):289–307.

Zhao Xin received the B.S. and M.S. degrees, both in control engineering, from the High-Tech Institute of Xi'an, China, in 2006 and 2009, respectively. He is currently working towards the Ph.D. degree with the same institute. His current research interests include integrated navigation, fault diagnosis, and algorithm quality evaluation.

Wang Shicheng is a professor and Ph.D. supervisor at the High-Tech Institute of Xi'an, China. He received the Ph.D. degree from the same institute in 1998. He current research interests include navigation, guidance and control, as well as system simulation.