

GPS fault detection and exclusion using moving average filters

Y.-H. Tsai, F.-R. Chang and W.-C. Yang

Abstract: A new approach based on the moving average (MA) is proposed to perform satellite failure detection and exclusion (FDE). By taking the average of the last few sums of the squares of the GPS range residual errors, the MA filter is applied to speed up failure detection. The detection threshold cannot be obtained directly because the cumulative distribution of a random process with an MA filter is unknown. Therefore, the Markov chain approach is applied to resolve the threshold. In addition, variations in the number of visible satellites may cause problems in data fusion. The probability integral transformation (PIT) method is adopted to overcome this. After a satellite failure is detected, the multivariate MA filter is used to reduce the incorrect exclusion rate (IER) by taking the average of the last few parity vectors. Simulation results show that, in comparison with the conventional least-squares-residuals method, the MA filter demonstrates higher performance in detecting small failures and a similar level of performance in detecting large failures. Moreover, simulation results also verify that the proposed method has lower IER than the conventional parity space method.

1 Introduction

Fault detection and exclusion (FDE) in the Global Positioning System (GPS) is a crucial issue in aviation navigation, as satellite failures may result in serious deviations of aircraft from their intended flight paths. Sometimes the term 'receiver autonomous integrity monitoring (RAIM)' is adopted for a similar concern. The purpose of FDE is to 'detect' the presence of unacceptably large position error and, further, to 'exclude' the source causing the error. To achieve this goal, a number of useful FDE algorithms have been published over the last few decades. Parkinson and Axelrad [1] suggested a least-squares residual method for autonomous GPS satellite failure detection and exclusion. Sturza [2] proposed the standard parity space algorithm to detect the satellite failure and further to exclude the failed one. To be precise, the FDE algorithms used in these earlier papers are static and depend only on current measurements. Brown and Hwang [3] proposed a dynamic algorithm by using a parallel bank of Kalman filters to detect failures. The GPS measurement is first subtracted by a specified type of failure and then applied to a Kalman filter. For each type of failure, a corresponding Kalman filter is necessary. Younes *et al.* [4] proposed a sequential RAIM algorithm to detect satellite malfunction and also to exclude the failed satellite. To achieve this goal, the CUSUM (cumulative sum) algorithm is adopted to detect the occurrences of mean changes in GPS

least square residuals. Later, Yang *et al.* [5] proposed the exponential weighted moving average (EWMA) filter, which uses all past data, to perform failure detection. However, that proposed method focuses only on failure detection.

Because failure may exist in measurements already made before the failure is detected, an algorithm based on the moving average (MA), which accumulates previous finite data, is proposed by the authors to perform fault detection and exclusion. It is assumed throughout this paper that at most one satellite failure occurs at a time. The proposed algorithm includes two parts: fault detection and fault exclusion. In the first part, on MA filter is proposed to speed up failure detection by taking the average of the last few sums of the squares of the GPS range residual errors (SSE). Speeding up failure detection might provide more time for pilots to prevent serious deviations of aircraft from their intended flight paths. In order to calculate the detection threshold under a specified false alarm rate (FAR), the MA filter is first transformed into a state-space model, and then the threshold can be approximated by a 'discrete finite-state Markov chain'. Because the numbers of visible satellites may vary with time, the calculated SSE may have different distributions and cannot be directly applied to the MA filter. Thus, the probability integral transformation (PIT) [6] is adopted to deal with this problem. In the second part, the multivariate MA filter is proposed to reduce the incorrect exclusion rate by taking the average of the last few parity vectors. The errors in the pseudorange include the ionospheric delay, tropospheric delay, multipath, cross-correlation effect, satellite clock offset etc. To satisfy the assumption of Gaussian distribution of the error in the pseudorange, a dual frequency including L2 and L5 signals [7] is adopted in this paper to remove the ionospheric delay [8] and the cross-correlation effect [7]. Simulation results show that, in comparison with the conventional fault detection methods, the MA filter demonstrates higher performance in detecting small failures and, in detecting large failures, demonstrates a similar level of

© IEE, 2004

IEE Proceedings online no. 20040728

doi: 10.1049/ip-rsn:20040728

Paper first received 5th January and in revised form 13th May 2004

Y.-H. Tsai and F.-R. Chang are with the Department of Electrical Engineering, National Taiwan University, Taipei, Taiwan, Republic of China

W.-C. Yang is with the Division of Research and Development in Intelligent Business Technology Inc., 16F, No. 30, Pei-Ping E. Rd., Taipei, Taiwan, Republic of China

performance. Moreover, simulation results also verify that the proposed method has lower IER than the parity space method has.

2 Background on fault detection and exclusion

The use of dual frequencies can eliminate ionospheric delay and the cross-correlation effect so that the error in the pseudorange measurement can be approximated as Gaussian noise. As such, conventional fault detection and exclusion algorithms can be derived as follows. Assume $\boldsymbol{\rho}(k)$ is the ionosphere-free pseudorange measurement obtained using dual frequencies [8]. When the nominal user position is taken as the reference point, the linearised ionosphere-free measurement equation can be obtained as

$$\mathbf{y}(k) = \mathbf{H}(k)\mathbf{x}(k) + \mathbf{e}(k) \quad (1)$$

where $\mathbf{y}(k)$ is the $n \times 1$ measurement vector, which is the difference between $\boldsymbol{\rho}(k)$ and the predicted range based on the nominal user position; $\mathbf{H}(k)$ is the $n \times 4$ observation matrix arrived at by linearising around the nominal user position and clock bias; $\mathbf{x}(k)$ is the 4×1 state vector comprised of the true position deviation from the nominal position, plus the user clock bias deviation; $\mathbf{e}(k)$ is an $n \times 1$ zero mean Gaussian noise vector with covariance matrix $\sigma^2\mathbf{I}$; and n is the number of visible satellites.

The least-squares-residuals method of fault detection is derived as follows. According to [1], the least-squares estimate of the state vector is $\hat{\mathbf{x}}_{LS}(k) = (\mathbf{H}^T(k)\mathbf{H}(k))^{-1}\mathbf{H}^T(k)\mathbf{y}(k)$, the estimate of $\mathbf{y}(k)$ is $\hat{\mathbf{y}}(k) = \mathbf{H}(k)\hat{\mathbf{x}}_{LS}(k)$, and the pseudorange residual vector is $\mathbf{w}(k) = \mathbf{y}(k) - \hat{\mathbf{y}}(k)$. By defining the sum of the squares of the range residual errors (SSE) as $SSE(k) \equiv \mathbf{w}^T(k)\mathbf{w}(k)$. Then the normalised SSE can be obtained as

$$s(k) = SSE(k)/\sigma^2 \quad (2)$$

Parkinson and Axelrad [1] showed that the distribution of $s(k)$ is $\chi^2(n-4)$, where $\chi^2(v)$ represents the chi-square distribution with v degrees of freedom. Then $s(k)$ will be compared with a threshold T_d to judge whether the system has failed or not. The threshold value under a specified false alarm rate (FAR) can be calculated directly through the cumulative distribution function of $\chi^2(n-4)$.

The parity space method can also perform the fault detection, and more, to exclude the range measurements associated to the fault satellite. According to [9], there exists an $n \times (n-4)$ parity matrix $\mathbf{P}(k)$ satisfying the following equation

$$\mathbf{P}(k)\mathbf{H}(k) = \mathbf{0} \text{ and } \mathbf{P}(k)\mathbf{P}^T(k) = \mathbf{I}_{n-4} \quad (3)$$

where n is the number of satellites in view at time k . After the parity matrix $\mathbf{P}(k)$ is found, the $(n-4) \times n$ parity vector can be defined as

$$\mathbf{p}(k) \equiv \mathbf{P}(k)\mathbf{y}(k) \quad (4)$$

Brown [10] showed that

$$\mathbf{p}^T(k)\mathbf{p}(k) = SSE(k) \quad (5)$$

and thus, as in the least-squares-residual method, the parity space method can also be used to perform the failure detection. In order to maintain sufficient redundancy, at least five visible satellites are required. After the detection of satellite malfunction, the failed satellite must be excluded to ensure uninterrupted navigation. Given the event of a failure vector $\mathbf{b}(k)$, (1) becomes

$$\mathbf{y}(k) = \mathbf{H}(k)\mathbf{x}(k) + \mathbf{b}(k) + \mathbf{e}(k) \quad (6)$$

$\mathbf{b}(k)$ is an $n \times 1$ vector of uncompensated measurement failures with the h_f th element equal to b and zeros elsewhere, where b is the magnitude of the failure and h_f is the channel number of the failed satellite. If no failure occurs, then $\mathbf{b}(k) = \mathbf{0}$. Based on the standard parity vector exclusion algorithm [2, 9], the algorithm to identify the failed satellite is as follows

$$n_f = \arg \max_{i=1, \dots, n} \frac{|\mathbf{p}^T(k)\mathbf{p}_i(k)|}{|\mathbf{p}_i(k)|} \quad (7)$$

where n_f is the identified channel number of the possible failed satellite at time k , and $\mathbf{p}_i(k)$ is the i th column vector of the parity matrix $\mathbf{P}(k)$. $\mathbf{p}_i(k)$ is also called the i th channel vector since it is related to the i th satellite.

3 Fault detection algorithms using moving average filters

To speed up the satellite failure detection, the moving average (MA) filter is proposed. The number of satellites in view is assumed to be constant at the moment. The case when the number of visible satellites changes will be described later. The scheme of the MA filter is based on the following statistic

$$z(k) = \sum_{i=1}^m w_i s(k-i+1) \quad (8)$$

where k is the running time index, m is the window size, $s(\cdot)$ is as defined in (2), and w_i is the weight of the filter with non-negative value and satisfies $\sum_{i=1}^m w_i = 1$. As no satellite failure occurs, $s(\cdot)$ is chi-square distributed with v degrees of freedom, and then the expectation value of $s(\cdot)$ will equal to the associated degrees of freedom, v . Therefore, the initial conditions of (8) are set as $s(0) = s(-1) = \dots = v$. Take expectation on both sides of (8), and it can be obtained that the expectation value of $z(\cdot)$ will equal to the expectation value of $s(\cdot)$ as no satellite failure occurs. For the special case of $m = 1$, $z(k)$ is equal to $s(k)$, and the result of the MA filter is exactly the same as the least-squares-residuals. If a satellite failure is detected at time k , then the past data $s(k-1), \dots, s(k-m)$ might contain a bias already. Therefore, the past data cannot be applied to the MA filter any more, and the filter should be reset. The process of reset is based on the initial conditions of (8) as $s(k-1) = \dots = s(k-m) = v$. As a result, the cumulative distribution function (cdf) of $z(\cdot)$ cannot be directly obtained through algebraic calculation. Thus, the threshold cannot be calculated in the traditional way as in the previous Section.

The MA filter can be converted into a state space model, which describes the relationship between the current data and the previous one and matches the properties of Markov chain. Therefore, the Markov chain approach is proposed to obtain the threshold value of the MA filter. First, (8) is transformed to a state space model. Set $\kappa_i(k) = s(k-i+1)$, for $i = 1, \dots, m$, and define the state vector $\boldsymbol{\kappa}(k) = [\kappa_1(k) \dots \kappa_i(k) \dots \kappa_m(k)]^T$. Then $\boldsymbol{\kappa}(k)$ can be represented in a dynamic equation as

$$\boldsymbol{\kappa}(k) = \boldsymbol{\Phi}\boldsymbol{\kappa}(k-1) + \boldsymbol{\beta}s(k) \quad (9)$$

where

$$\boldsymbol{\Phi} = \{\phi_{ij}\} \text{ with } \phi_{ij} = \begin{cases} 1, & i=j+1 \\ 0, & \text{otherwise} \end{cases}, \text{ and } \boldsymbol{\beta} = [1 \ 0 \ \dots \ 0]^T$$

The initial conditions of (9) are set as $\boldsymbol{\kappa}(0) = [E[s(\cdot)]] \cdots E[s(\cdot)]^T$. Then (8) can be represented as

$$z(k) = \mathbf{w}^T \boldsymbol{\kappa}(k) \quad (10)$$

where $\mathbf{w} = [w_1 \cdots w_m]^T$ is the weighting vector satisfying $\sum_{i=1}^m w_i = 1$. Since $z(k)$ is related to $\boldsymbol{\kappa}(k)$ through (10), the calculation of the threshold of $z(k)$ can be performed on $\boldsymbol{\kappa}(k)$.

Because the state vector $\boldsymbol{\kappa}(k)$ in (9) is a Markov process, calculation of the threshold under a specific false alarm rate (FAR) α can be approximated by modelling the process $\boldsymbol{\kappa}(k)$ as a discrete Markov chain with a stationary transition probability matrix. There are two types of Markov states: transient states and a terminating state. When $z(k)$ is less than or equal to the threshold value T_d , the process belongs to transient states. Otherwise, it belongs to the terminating state. Once the state becomes terminating, it will not go back to the transient state. A window size equal to 2 is discussed first, and the general case will be described in the Appendix. The corresponding state equation is as follows:

$$\boldsymbol{\kappa}(k) = \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix} \boldsymbol{\kappa}(k-1) + \begin{bmatrix} 1 \\ 0 \end{bmatrix} s(k) \quad (11)$$

where $\boldsymbol{\kappa}(k) = [\kappa_1(k) \ \kappa_2(k)]^T$ and both $\kappa_1(k)$ and $\kappa_2(k)$ are non-negative because $s(k)$ is non-negative. The process is in transient states at time k if and only if $w_1 \kappa_1(k) + w_2 \kappa_2(k) = z(k) \leq T_d$, where w_1 and w_2 are weights satisfying $w_1 + w_2 = 1$. Therefore, the transient states are bounded by the triangle area in Fig. 1. To calculate the transition probability matrix, this area is divided into L subareas, $S_1, S_2, \dots, S_i, \dots, S_L$, each of which represents a transient state. The division is performed along the κ_1 -axis with the same width, $2\delta = w_1^{-1} T_d / L$. The center of the i th subarea, S_i , is (μ_i, ζ_i) , where $\mu_i = (2i-1)\delta$ and $\zeta_i = w_2^{-1} (T_d - w_1 \mu_i) / 2$. Then, $\boldsymbol{\kappa}(k)$ belongs to transient state S_i if and only if

$$\begin{cases} \kappa_1(k) \in [\mu_i - \delta, \mu_i + \delta] \\ w_1 \kappa_1(k) + w_2 \kappa_2(k) \leq T_d \end{cases} \quad (12)$$

Assume the distribution of $\boldsymbol{\kappa}(k)$ can be described by the probability vector $\boldsymbol{\pi}(k) = [\pi_1(k) \cdots \pi_i(k) \cdots \pi_L(k)]^T$, where $\pi_i(k)$ represents the probability of $\boldsymbol{\kappa}(k)$ being in state S_i . Note that $\sum_{i=1}^L \pi_i(k)$ is the probability that no alarm is activated and the satellites are all judged as normal at time k . Then the transition of the probability vector can be represented as

$$\boldsymbol{\pi}(k) = \mathbf{T} \boldsymbol{\pi}(k-1) \quad (13)$$

where \mathbf{T} is the transition probability matrix among transient states [11]. \mathbf{T} is defined as

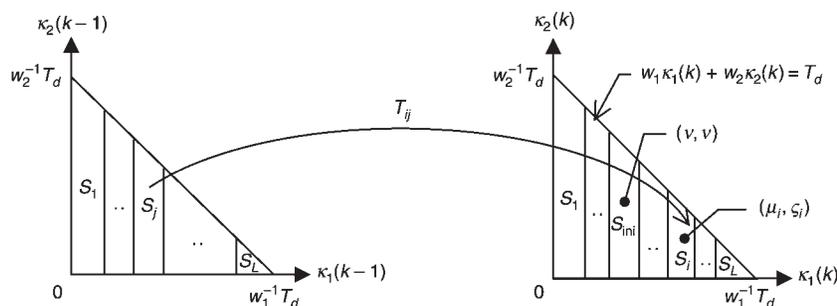


Fig. 1 Transient of Markov chain for MA filter with window size 2

$$\mathbf{T} = \begin{bmatrix} T_{1,1} & \cdots & T_{1,L} \\ \vdots & \ddots & \vdots \\ T_{L,1} & \cdots & T_{L,L} \end{bmatrix} \quad (14)$$

where $T_{ij} \equiv \Pr[\boldsymbol{\kappa}(k) \text{ goes to state } S_j | \boldsymbol{\kappa}(k-1) \text{ was in state } S_i]$. As shown in Fig. 1, T_{ij} can be approximated as

$$T_{ij} = \Pr \left[\begin{aligned} & \{\mu_i - \delta \leq \kappa_1(k) \leq \mu_i + \delta\} \\ & \cap \{w_1 \kappa_1(k) + w_2 \kappa_2(k) \leq T_d\} \\ & \mid \kappa_1(k-1) = \mu_j \cap \kappa_2(k-1) = \zeta_j \end{aligned} \right] \quad (15)$$

In the previous equation, the elements of $\boldsymbol{\kappa}(k-1)$ being in state S_j are approximated by setting $\kappa_1(k-1) = \mu_j$ and $\kappa_2(k-1) = \zeta_j$. Substitute $\kappa_1(k) = s(k)$ and $\kappa_2(k) = \kappa_1(k-1) = \mu_j$ into (15), and we have

$$\begin{aligned} T_{ij} &= \Pr \left[\{\mu_i - \delta \leq s(k) \leq \mu_i + \delta\} \cap \{w_1 s(k) + w_2 \mu_j \leq T_d\} \right] \\ &= \Pr \left[\{\mu_i - \delta \leq s(k) \leq \mu_i + \delta\} \cap \{s(k) \leq w_1^{-1} (T_d - w_2 \mu_j)\} \right] \\ &= \begin{cases} \Pr[\underline{B}_{ij} \leq s(k) \leq \bar{B}_{ij}], & \text{if } \underline{B}_{ij} \leq \bar{B}_{ij} \\ 0, & \text{otherwise} \end{cases} \end{aligned} \quad (16)$$

where $\underline{B}_{ij} = \mu_i - \delta$ and $\bar{B}_{ij} = \min(\mu_i + \delta, w_1^{-1} (T_d - w_2 \mu_j))$. The division is performed only on the x_1 -axis because $x_2(k-1) = \zeta_j$ will not appear in (16) during the transition.

The initial conditions of the probability vector $\boldsymbol{\pi}(0)$ are set as a vector with all elements equal to zero, except that $\pi_{\text{ini}}(0)$ is set to 1, where $\pi_{\text{ini}}(0)$ corresponds to state S_{ini} containing $\boldsymbol{\kappa}(0) = [v \ v]^T$. Figure 1 also depicts the above. The time to false alarm is defined as the time needed from the beginning of detection to the declaration of a false alarm, the mean of which is called mean time to false alarm (MTFA) [12]. MTFA can be represented as

$$M_{FA} = \mathbf{I}(\mathbf{I}_L - \mathbf{T})^{-1} \boldsymbol{\pi}(0) \quad (17)$$

where \mathbf{I} is a $1 \times L$ row vector with all elements equal to one, and \mathbf{I}_L is a $L \times L$ identity matrix. In fact, the false alarm rate (FAR) is equal to the inverse of MTFA, i.e. $M_{FA} = 1/\alpha$.

The procedures to obtain T_d^* under a specific window size m^* and weight $w_i = w_i^*$ for $i = 1, \dots, m$ are summarised in the following steps [13].

Step 1: Set $m = m^*$, $w_i = w_i^*$ for $i = 1, \dots, m$, degrees of freedom $v = 2$ and $M_{FA}^* = 1/\alpha$.

Step 2: Make an initial guess T_d^0 and calculate the corresponding M_{FA}^0 .

Step 3: If M_{FA}^0 is larger than M_{FA}^* , decrease T_d^0 ; otherwise, increase T_d^0 .

Step 4: If $|M_{FA}^0 - M_{FA}^*| \leq \text{tolerance}$, then the T_d^* corresponding to m^* and w_i^* is obtained.

The process is repeated until adequate resolution is attained.

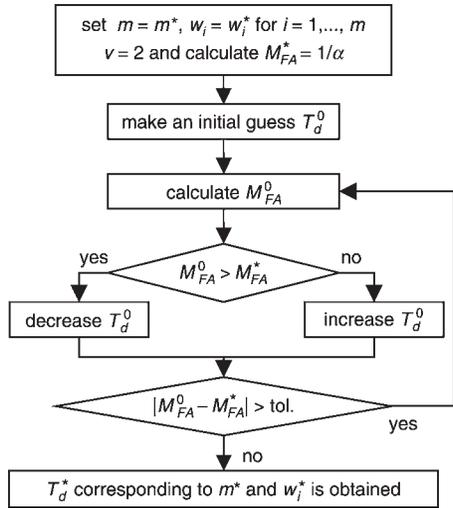


Fig. 2 Flow chart for the calculation of threshold

Table 1: Threshold value of MA filter with six satellites in view under FAR = 1/15000

Window size	Threshold
1	19.2316
2	12.0159
3	9.3713
4	7.9669
5	7.0898

The above procedures are also depicted in Fig. 2. Owing to space limitations, only the resulting threshold values with equal weights are listed in Table 1, where α was set to 1/15000 [14].

4 Data adjustment for different numbers of satellites

Thus far, the number of satellites in view has been supposed to be constant. In real situations, however, the number of visible satellites may change with time. Because the conventional parity space method [2] is a snapshot-type method, the determination of thresholds depends only on the current number of the visible satellites. However, the MA-filter takes the average of the last few data, which may be produced from different numbers of visible satellites. These data cannot be directly applied to the MA-filter because they may have different distributions. Thus, the probability integral transformation (PIT) [6] is proposed to pre-process the data into a useable form.

The PIT procedure is to transform a random variable with a specific continuous distribution into another one with a different distribution that retains the same cumulative probability value. Because the test statistic is chi-square distributed, we will focus on this distribution. Suppose that X and Y are two random variables, having chi-square distributions with μ and ν degrees of freedom, respectively. The associated cumulative distribution functions are with cdf $F_\mu(\cdot)$ and $F_\nu(\cdot)$. Assume that x and y are realisations of X and Y , and let $F_\mu^{-1}(\cdot)$ be the inverse function of $F_\mu(\cdot)$. Define $z = F_\nu(y)$, and set $x = F_\mu^{-1}(z)$. The transformation from y into x can be condensed into a single formula as

$$x = F_\mu^{-1}(F_\nu(y)) \quad (18)$$

During the transformation, the cumulative probabilities of x and y are equal since $F_\mu(x) = F_\nu(y)$.

From 'table of the χ^2 distribution' in [15], the chi-square cdf with ν degrees of freedom is

$$F_\nu(Y) = \begin{cases} 1 - e^{-Y/2} \left(\operatorname{erfcx}\left(\sqrt{\frac{Y}{2}}\right) + \sqrt{\frac{2}{\pi}} \sum_{k=0}^{\frac{\nu-3}{2}} \frac{2^k k!}{(2k+1)!} Y^{\frac{2k+1}{2}} \right) & \nu \text{ is odd} \\ 1 - e^{-Y/2} \left(\sum_{k=0}^{\frac{\nu-2}{2}} \frac{Y^k}{2^k k!} \right) & \nu \text{ is even} \end{cases} \quad (19)$$

where

$$\operatorname{erfcx}(x) = \frac{2}{\sqrt{\pi}} e^{-x^2} \int_x^\infty e^{-u^2} du$$

is the scaled complementary error function. In (19) it is difficult to find a closed form of the inverse chi-square cdf in most cases. However, since the cdf of $\chi^2(2)$ is $F_2(Y) = 1 - e^{-0.5Y}$, the inverse function can be obtained as

$$F_2^{-1}(Z) = -2 \log(1 - Z) \quad (20)$$

Substitute (19) and (20) into (18), and then the formula of PIT can be represented as

$$X = F_2^{-1}(F_\nu(Y)) = -2 \log(1 - F_\nu(Y)) = \begin{cases} Y - 2 \log \left(\operatorname{erfcx}\left(\sqrt{\frac{Y}{2}}\right) + \sqrt{\frac{2}{\pi}} \sum_{k=0}^{\frac{\nu-3}{2}} \frac{2^k k!}{(2k+1)!} Y^{\frac{2k+1}{2}} \right) & \nu \text{ is odd} \\ Y - 2 \log \left(\sum_{k=0}^{\frac{\nu-2}{2}} \frac{Y^k}{2^k k!} \right) & \nu \text{ is even} \end{cases} \quad (21)$$

where X and Y are random variables, having chi-square distributions with 2 and ν degrees of freedom respectively. From (21), the PIT can be represented in a closed form if ν is even. An extra function $\operatorname{erfcx}(x)$ is needed in the formula if ν is odd.

In order to help explain how PIT works, a simple example is demonstrated here. Suppose there are 10 satellites in

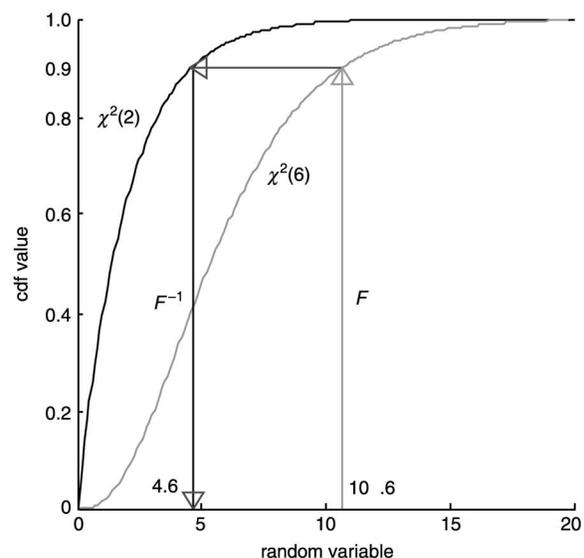


Fig. 3 Illustration of PIT method

view, and the statistic SSE is consequently χ^2 (6). If an SSE = 10.6 is obtained, then the datum 10.6 is discounted as 4.6, as if there are six satellites in view. Since $F_6(10.6) = 0.9$, the value 0.9 is transferred to 4.6 via $F_2^{-1}(0.9) = 4.6$. Figure 3 illustrates the above procedure.

5 Fault exclusion algorithms using multivariate moving average filters

In this Section, an exclusion algorithm based on the multivariate MA filter is introduced to reduce the incorrect exclusion rate (IER). The multivariate MA filter is defined as

$$\mathbf{p}_{MA}(k) = \sum_{i=1}^{\tau} \omega_i \mathbf{p}(k-i+1) \quad (22)$$

where $\mathbf{p}(\cdot)$ is the $(n-4) \times 1$ parity vector defined in (4), ω_i is the weight of the filter with non-negative value and satisfies $\sum_{i=1}^{\tau} \omega_i = 1$, k is the running time index, and τ is the window size. The initial conditions of (22) are set as $\mathbf{p}(0) = \mathbf{p}(-1) = \dots = \mathbf{0}$. The parity matrix $\mathbf{P}(k)$ can be regarded as a constant matrix because the $\mathbf{H}(k)$ matrix changes slowly in a short period of time. Therefore (22) can be simplified as

$$\mathbf{p}_{MA}(k) = \mathbf{P}(k) \mathbf{y}_{MA}(k) \quad (23)$$

where $\mathbf{y}_{MA}(k) = \sum_{i=1}^{\tau} \omega_i \mathbf{y}(k-i+1)$. According to (7), the exclusion algorithm based on (23) can be defined as

$$n_f = \arg \max_{i=1, \dots, n} \frac{|\mathbf{p}_{MA}^T(k) \mathbf{p}_i(k)|}{|\mathbf{p}_i(k)|} \quad (24)$$

where n_f is the channel of the failed satellite at time k , and $\mathbf{p}_i(k)$ is the i th column vector of the $(n-4) \times n$ parity matrix $\mathbf{P}(k)$.

To show that the proposed algorithm can reduce the IER, the following discussion is given. Substituting (6) into (23) results in

$$\mathbf{p}_{MA}(k) = \mathbf{P}(k) [\mathbf{b}_{MA}(k) + \mathbf{e}_{MA}(k)] \quad (25)$$

where $\mathbf{b}_{MA}(k) = \sum_{i=1}^{\tau} \omega_i \mathbf{b}(k-i+1)$ is the weighted sum of failure vectors and $\mathbf{e}_{MA}(k) = \sum_{i=1}^{\tau} \omega_i \mathbf{e}(k-i+1)$ is a zero mean Gaussian noise vector with covariance matrix $(\sum_{i=1}^{\tau} \omega_i^2) \sigma^2 \mathbf{I}_n$. In the derivation of (25), the approximation $\mathbf{H}(k-1) = \mathbf{H}(k-2) = \dots = \mathbf{H}(k-\tau+1) = \mathbf{H}(k)$ was made because $\mathbf{H}(k)$ changes slowly in a short period of time. Then $\mathbf{p}_{MA}(k)$ will be a Gaussian noise vector with mean $\mathbf{P}(k) \mathbf{b}_{MA}(k)$ and covariance matrix $(\sum_{i=1}^{\tau} \omega_i^2) \sigma^2 \mathbf{I}_{n-4}$. Since $\sum_{i=1}^{\tau} \omega_i^2 < 1$, the variance in each element of $\mathbf{p}_{MA}(k)$ is smaller than that for each element of $\mathbf{p}(k)$. The reduction in variance will provide a cleaner fault observation, and thus the IER will be reduced. If any satellite in view passes below the mask angle, then only the previous measurements of the remaining satellites are reserved to perform the fault exclusion. However, if an extra satellite passes above the mask angle, then the past data cannot be directly applied to the filter, and thus the multivariate MA filter will be reset.

6 Simulation results

Monte Carlo simulations are conducted to verify the proposed FDE algorithm. The software package ‘Satellite Navigation TOOLBOX for Matlab’ by GPSofT LLC is adopted in the simulation. It assumed a 24-satellite constellation with perfectly circular orbits. Dual frequencies are used to eliminate the ionospheric delay. Therefore, only the thermal noise and tropospheric delay are left in the ionosphere-free pseudorange measurements. A total of 1152

(24×48) space-time sample points were produced according to the user locations and simulation times mentioned in [16]. The user locations cover the 24 geographic locations [13], and the simulation time is every half hour for 24 hours starting at midnight at the beginning of the GPS week. In addition, the receiver mask angle is set as 7.5° .

To show that the proposed MA filter can speed up the failure detection, ramp-type and step-type pseudorange failures were used to simulate satellite malfunction. Ramp-type failure refers to a failure growing linearly with time, and step-type failure refers to a constant bias happening at a specified time and continuing after this. An MA filter with equal weights was adopted in the simulation of failure detection because the test statistic obtained has minimum variance when no satellite failure occurs. The corresponding threshold values can be found in Table 1. The procedure for using the MA filter to detect satellite failure for each point is summarised as follows.

Step 1: Set a specified window size m (ranging from 1 to 5) and the time index as one ($k = 1$).

Step 2: Calculate $s(k)$ from the user-satellite geometry through (1) and (2).

Step 3: Check the number of satellites in view. If it is not equal to the specified value 6, then the PIT method (21) is applied.

Step 4: Calculate $z(k)$ from (8).

Step 5: Compare the test statistic $z(k)$ with the threshold T_d under the specified window size. If the test statistic exceeds T_d , then the detection time is recorded. Otherwise the time index is increased by one ($k = k + 1$), and repeat step 2 through step 5.

Based on the simulation environments described above, a detection time (DT) can be obtained for each point. The DT is defined as the time needed from the onset of the failure to the announcement of an alarm signal. The average detection time (ADT) is the sample mean of the 1152 values of the DT. The simulation results of the ADT for ramp-type pseudorange failure are plotted in Fig. 4a. On average, an MA filter with a window size larger than 1 has a shorter DT than the original SSE (window size = 1). The resulting ADT is expressed as a percentage improvement of ADT (PIADT) through the following equation:

$$\text{PIADT} = \frac{\text{ADT}(\text{LSR}) - \text{ADT}(\text{MA})}{\text{ADT}(\text{LSR})} \times 100\% \quad (26)$$

where ADT(MA) is the ADT when the MA filter is applied and ADT(LSR) is the ADT when the least-square-residual is applied. The result is depicted in Fig. 4b. It can be seen that the best improvement percentage is 26%; i.e. the detection is accelerated by 26%. Under small ramp-type failures (slope = 0.2, 0.5, 1.0 m/s), the PIADT will increase as the window size increases, while under large ramp-type failures (slope = 5, 10, 15 m/s), the resulting ADT is within 6 s for all detectors; this means that the window size has little influence on the PIADT. The simulation results of ADT for step-type pseudorange failures are plotted in Fig. 5a. The data in the Figure are also transferred into PIADT and are plotted in Fig. 5b. In this Figure, the best improvement percentage is 77%. Under small step-type failures (step = 20, 25, 30 m), the PIADT will also increase as the window size increases. However, under large step-type failures (step = 40 m), their performances are similar. To sum up, in comparison with conventional fault detection methods, the MA filter has higher performance when detecting small failures and, has similar performance when detecting large failures.

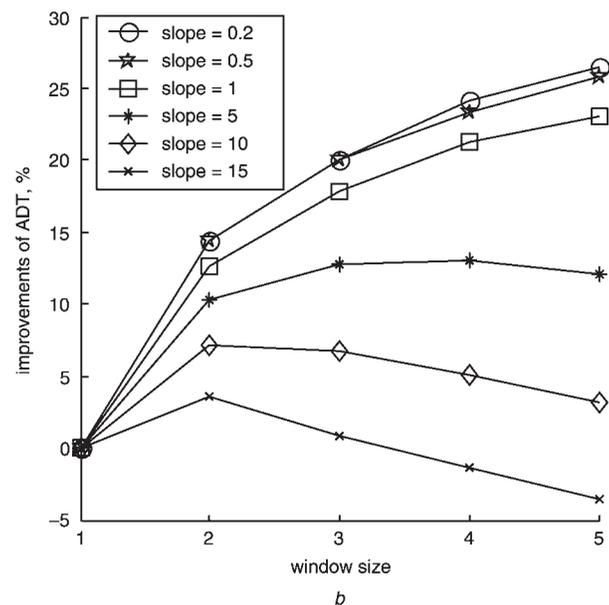
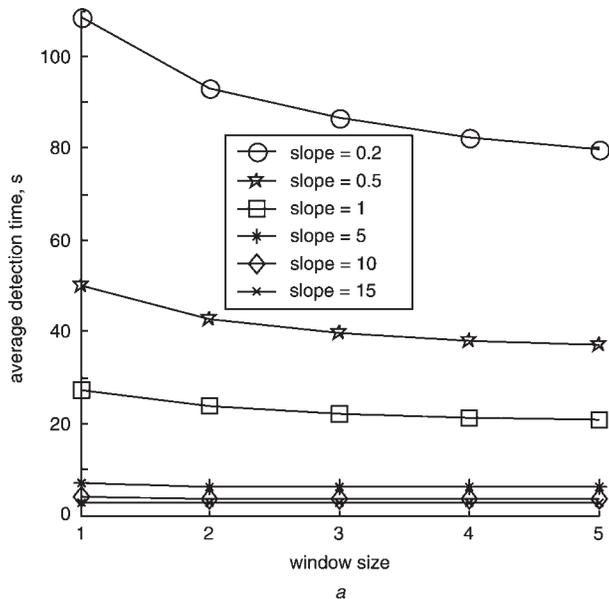


Fig. 4 Ramp-type pseudorange errors

a ADT
b PIADT

Although calculation of the threshold for the MA filter is complex, it is independent of satellite geometry. Therefore, it can be computed offline and tabulated in computer memory. Compared with the snapshot method, extra computational burdens for online operation including the MA filter and the PIT are needed. In fact, computation of the MA filter will take only a little effort, and the major computational burden will be for the PIT process. However, the calculation of PIT has been simplified in Section 4.

Furthermore, to show that the proposed multivariate MA filter can reduce the incorrect exclusion rate, ramp-type and step-type failures are also used to simulate satellite malfunction. The procedure using the multivariate MA filter to exclude the failed satellite is summarised as follows.

- Step 1: Set a specified window size τ (ranging from 1 to 5).
- Step 2: Calculate parity matrix and parity vector from the user-satellite geometry through (3) and (4).
- Step 3: Calculate $p_{MA}(k)$ from (23) under the specified window size.

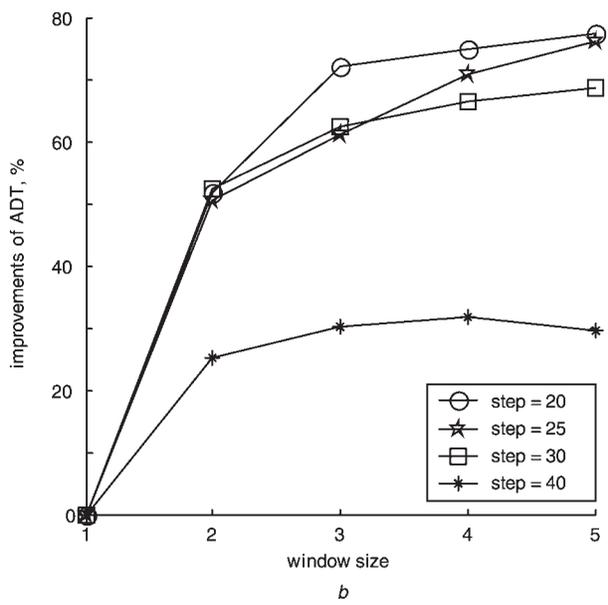
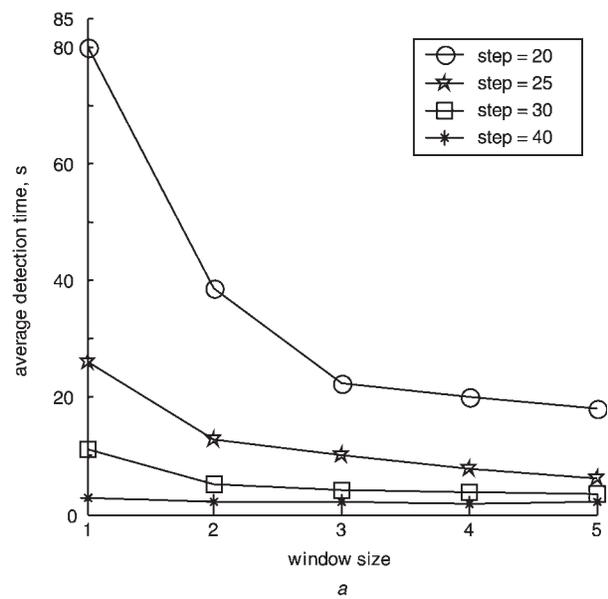


Fig. 5 Step-type pseudorange errors

a ADT
b PIADT

Step 4: Identify the channel number of the failed satellite, n_f , from (24).

Step 5: Comparing n_f with the channel number of the true failed satellite, if they did not match, then an incorrect exclusion (IE) is recorded.

An IE occurs when the receiver performs a valid detection, but the failed satellite remains in the solution after the exclusion operation [17]. The incorrect exclusion rate (IER), which is used as a performance index, is defined as

$$\text{IER} = \frac{\text{no. of incorrect exclusions}}{\text{total no. of exclusions}} \times 100\% \quad (27)$$

IER can be used to verify the superior exclusion capability of the proposed multivariate MA filter. In the simulation, the number of total exclusion is chosen as 1152. Simulation results for the ramp-type failure with slope 0.5 and 10 m/s are given in Figs. 6a and 6b, respectively. The values of records on the abscissa denote the time duration from the onset of satellite failure. Both Figures show that the proposed multivariate MA filter with a window size larger

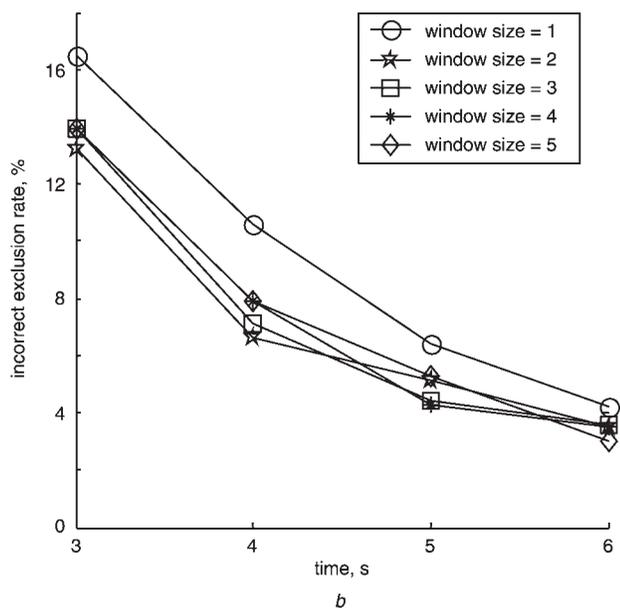
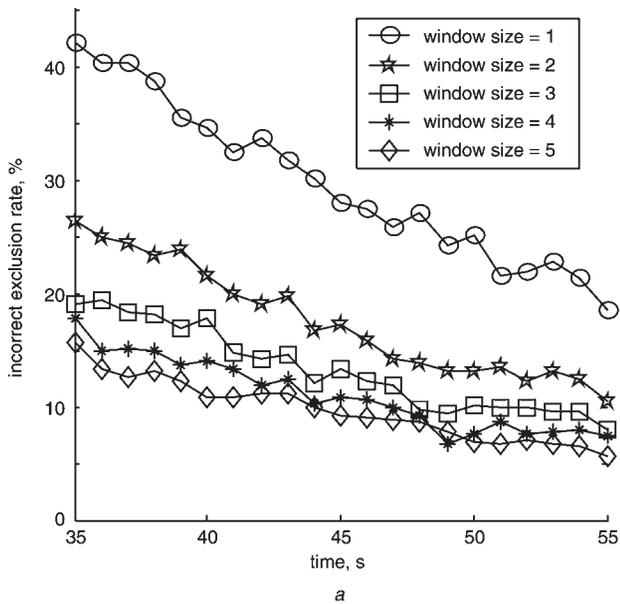


Fig. 6 IER under a ramp-type failure
 a Slope = 0.5 m/s
 b Slope = 10 m/s

than 1 has lower IER than the original parity space method (window size = 1). Simulation results for step-type failure with bias 25 and 40m are given in Figs. 7a and 7b, respectively. These Figures also verify that the proposed method has lower IER than the original parity space method. In conclusion, multivariate MA filters can reduce the IER in excluding the failed satellite.

7 Conclusions

The authors proposed a moving average (MA) filter to speed up satellite failure detection and a multivariate MA filter to reduce the incorrect exclusion rate. The detection threshold can be obtained via the Markov chain approach under a specific false alarm rate. Although calculation of the threshold for the MA filter is complex, it is independent of satellite geometry and can be computed offline and tabulated in advance. To simulate satellite malfunction, ramp-type failure and step-type failure were applied. Simulation results show that, in comparison with the conventional fault detection methods, the MA filter has

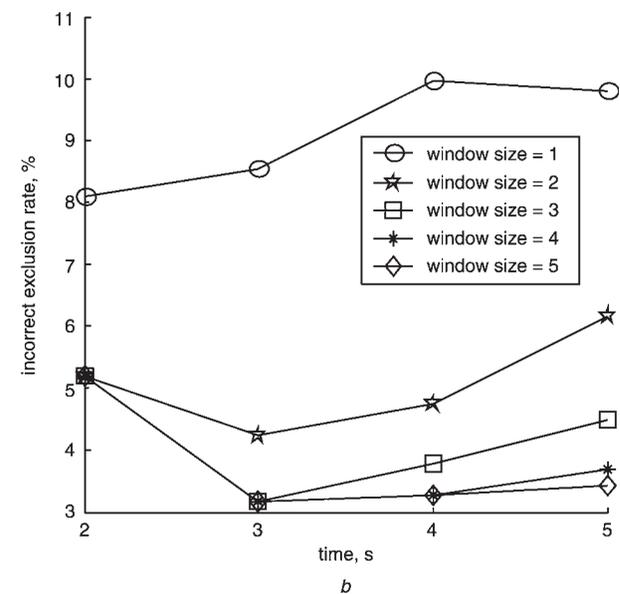
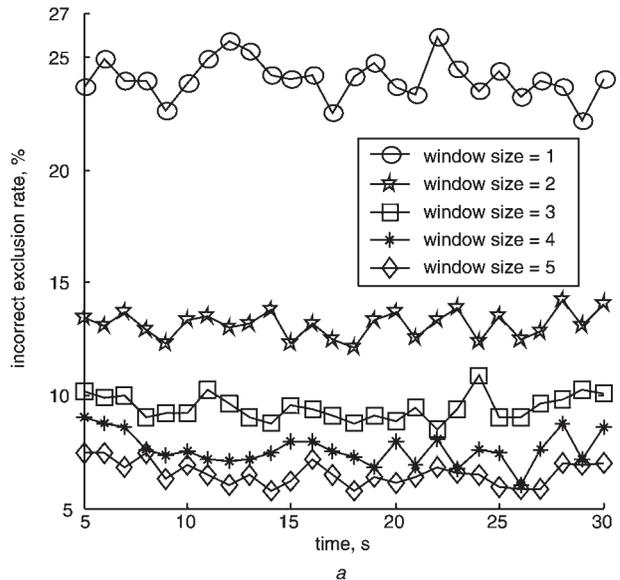


Fig. 7 IER under a step-type failure
 a Bias = 25 m
 b Bias = 40 m

higher performance when detecting small failures and, when detecting large failures, similar performance. Moreover, simulation results also verify that the proposed method has lower IER than the parity space method has. In conclusion, the proposed multivariate MA filters can reduce the IER in excluding the failed satellite. Using a multivariate MA filter for detection will be for future work. However, determination of the threshold and variation of the number of visible satellites still need to be solved.

8 Acknowledgments

The authors would like to express their deepest gratitude to Dr Chih-Min Fan for his kind assistance in correcting the present paper.

9 References

- 1 Parkinson, B.W., and Axelrad, P.: 'Autonomous GPS integrity monitoring using the pseudorange residual', *Navig., J. Inst. Navig.*, 1988, **35**, (2), pp. 255-274
- 2 Sturza, M.A.: 'Navigation system integrity monitoring using redundant measurements', *Navig., J. Inst. Navig.*, 1988, **35**, (4), pp. 483-501

- 3 Brown, R.G., and Hwang, P.: 'GPS failure detection by autonomous means within the cockpit'. Proc. 42nd Annual Meeting of the Institute of Navigation, June 1986, pp. 5-12
- 4 Younes, A., Benhallam, A., Bakhache, B., and Nikiforov, I.: 'Sequential RAIM: theory and application to civil aviation needs'. Proc. 11th Int. Meeting of the Satellite Division of the Institute of Navigation, September 1998, pp. 1993-2002
- 5 Yang, W.-C., Chang, F.-R., and Fan, C.-M.: 'Threshold values of EWMA-filter for GPS RAIM requirement'. Proc. 4th Pacific Int. Conf. on Aerospace Science and Technology, May 2001, pp. 63-68
- 6 Quesenberry, C.P.: 'Probability integral transformations', in Kotz, S., and Johnson, N.L. (Eds.): 'Encyclopedia of statistical sciences' (John Wiley & Sons, 1986)
- 7 Fontana, D.R., Cheung, W., and Stansell, T.: 'The modernized L2 civil signal: leaping forward in the 21st century', *GPS World*, 2001, pp. 28-34, <http://www.gpsworld.com>
- 8 Hofmann-Wellenhof, B., Lichtenegger, H., and Collins, J.: 'Global Positioning System: theory and practice' (Springer, Wien, New York, 1997, 4th revised edn.)
- 9 Diggelen, V.F., and Brown, A.: 'Mathematical aspects of GPS RAIM'. Proc. IEEE Position Location and Navigation Symp., 1994, pp. 733-738
- 10 Brown, R.G.: 'A baseline GPS RAIM scheme and a note on the equivalence of three RAIM methods', *Navig., J. Inst. Navig.*, 1992, **39**, (3), pp. 301-316
- 11 Isaacson, D.L.: 'Markov chains: theory and applications' (John Wiley & Sons, 1976)
- 12 Yang, W.-C., Chang, F.-R., Tsai, Y.-H., and Fan, C.-M.: 'Early detection of target manoeuvres under a specific false alarm rate', *IEE Proc., Radar Sonar Navig.*, 2003, **150**, (5), pp. 350-355
- 13 Tsai, Y.-H., Chang, F.-R., and Yang, W.-C.: 'Moving average filters for faster GPS receiver autonomous integrity monitoring'. Proc. 2002 ION National Technical Meeting, Jan. 2002, pp. 666-675
- 14 Minimum operational performance standards for airborne equipment using Global Positioning System/Wide Area Augmentation System. Document RTCA/DO-229, Radio Technical Commission for Aeronautics, 1997
- 15 Lancaster, H.O.: 'The chi-squared distribution' (John Wiley & Sons, 1969)
- 16 Minimum operational performance standards for airborne supplemental navigation equipment using Global Positioning System (GPS). Document RTCA/DO-208, Radio Technical Commission for Aeronautics, 1991
- 17 Lee, Y., Dyke, K.V., Declene, B., Studenny, J., and Beckmann, M.: 'Summary of RTCA SC-159 GPS integrity Working Group Activities', *Navig., J. Inst. Navig.*, 1996, **43**, (3), pp. 195-226

10 Appendix: Calculation of threshold value for window size equal to m

All elements $\kappa_i(k)$ in (9) are non-negative because $s(k)$ is non-negative. The process is in transient states at time k if and only if $\sum_{i=1}^m w_i \kappa_i(k) = z(k) \leq T_d$, where w_i are weights satisfying $\sum_{i=1}^m w_i = 1$. To calculate the transition probability matrix, the hyperspace related to transient states is divided into L subareas, $S_1, S_2, \dots, S_i, \dots, S_L$, each of which represents a transient state. The divisions are performed along all axes except the κ_m -axis. The division is performed on the κ_l -axis with the same width, $2\delta_l = w_l^{-1} T_d / L_l$, for $l = 1, \dots, m-1$. Assume the centre of the i th subarea, S_i to be $(\mu_i^1, \mu_i^2, \dots, \mu_i^{m-1}, \zeta_i)$. Then, $\kappa(k)$ belongs to transient state S_i if and only if

$$\begin{cases} \kappa_l(k) \in [\mu_i^l - \delta_l, \mu_i^l + \delta_l], & \text{for } l = 1, \dots, m-1 \\ \sum_{i=1}^m w_i \kappa_i(k) < T_d \end{cases} \quad (28)$$

The transition probability matrix among transient states is as (14), where $T_{ij} \equiv \Pr[\kappa(k) \text{ goes to state } S_i | \kappa(k-1) \text{ was in state } S_j]$. T_{ij} can be approximated as

$$\begin{aligned} T_{ij} = & \Pr \left\{ \left\{ \mu_i^1 - \delta_1 \leq \kappa_1(k) \leq \mu_i^1 + \delta_1 \right\} \right. \\ & \cap \left\{ \mu_i^2 - \delta_2 \leq \kappa_2(k) \leq \mu_i^2 + \delta_2 \right\} \\ & \cap \dots \cap \left\{ \mu_i^{m-1} - \delta_{m-1} \leq \kappa_{m-1}(k) \leq \mu_i^{m-1} + \delta_{m-1} \right\} \\ & \cap \left\{ \sum_{l=1}^m w_l \kappa_l(k) \leq T_d \right\} \left| \left\{ \kappa_1(k-1) = \mu_j^1 \right\} \right. \\ & \left. \cap \left\{ \kappa_2(k-1) = \mu_j^2 \right\} \cap \dots \cap \left\{ \kappa_m(k-1) = \zeta_j \right\} \right\} \end{aligned} \quad (29)$$

In the previous equation, the elements of $\kappa(k-1)$ which are in state S_j are approximated by setting $\kappa_l(k-1) = \mu_j^l$ for $l = 1, \dots, m-1$ and $\kappa_m(k-1) = \zeta_j$. Substitute $\kappa_1(k) = s(k)$, and $\kappa_l(k) = \kappa_{l-1}(k-1) = \mu_j^{l-1}$ for $l = 2, \dots, m$ into (29), and we have

$$\begin{aligned} T_{ij} = & \Pr \left\{ \left\{ \mu_i^1 - \delta_1 \leq s(k) \leq \mu_i^1 + \delta_1 \right\} \right. \\ & \cap \left\{ \mu_i^2 - \delta_2 \leq \mu_j^1 \leq \mu_i^2 + \delta_2 \right\} \\ & \cap \dots \cap \left\{ \mu_i^{m-1} - \delta_{m-1} \leq \mu_j^{m-2} \leq \mu_i^{m-1} + \delta_{m-1} \right\} \\ & \left. \cap \left\{ w_1 s(k) + \sum_{l=2}^m w_l \mu_j^{l-1} \leq T_d \right\} \right\} \\ = & \Pr \left\{ \left\{ \mu_i^1 - \delta_1 \leq s(k) \leq \mu_i^1 + \delta_1 \right\} \right. \\ & \cap \left\{ \left| \mu_j^1 - \mu_i^2 \right| \leq \delta_2 \right\} \cap \dots \cap \left\{ \left| \mu_j^{m-2} - \mu_i^{m-1} \right| \leq \delta_{m-1} \right\} \\ & \left. \cap \left\{ s(k) \leq w_1^{-1} \left(T_d - \sum_{l=2}^m w_l \mu_j^{l-1} \right) \right\} \right\} \\ = & \Pr \left\{ \left\{ \left[\mu_i^1 - \delta_1 \leq s(k) \leq \mu_i^1 + \delta_1 \right] \right. \right. \\ & \left. \cap \left[s(k) \leq w_1^{-1} \left(T_d - \sum_{l=2}^m w_l \mu_j^{l-1} \right) \right] \right\} \\ & \left. \cap \left\{ \left| \mu_j^1 - \mu_i^2 \right| \leq \delta_2 \right\} \cap \dots \cap \left\{ \left| \mu_j^{m-2} - \mu_i^{m-1} \right| \leq \delta_{m-1} \right\} \right\} \\ = & \Pr \left\{ \left\{ \underline{B}_{ij} \leq s(k) \leq \bar{B}_{ij} \right\} \cap \left\{ \left| \mu_j^1 - \mu_i^2 \right| \leq \delta_2 \right\} \right. \\ & \left. \cap \dots \cap \left\{ \left| \mu_j^{m-2} - \mu_i^{m-1} \right| \leq \delta_{m-1} \right\} \right\} \end{aligned} \quad (30)$$

where

$$\underline{B}_{ij} = \mu_i^1 - \delta_1 \text{ and } \bar{B}_{ij} = \min \left(\mu_i^1 + \delta_1, w_1^{-1} \left(T_d - \sum_{l=2}^m w_l \mu_j^{l-1} \right) \right)$$

In the last line of (30), $\left\{ \left| \mu_j^1 - \mu_i^2 \right| \leq \delta_2 \right\} \cap \dots \cap \left\{ \left| \mu_j^{m-2} - \mu_i^{m-1} \right| \leq \delta_{m-1} \right\}$ can be viewed as constraints since δ_j^l and μ_j^l for $l = 1, \dots, m-1$ are deterministic. Thus (30) can be simplified as follows:

$$\begin{aligned} T_{ij} = & \Pr \left\{ \left\{ \underline{B}_{ij} \leq s(k) \leq \bar{B}_{ij} \right\} \cap \left\{ \left| \mu_j^1 - \mu_i^2 \right| \leq \delta_2 \right\} \right. \\ & \left. \cap \dots \cap \left\{ \left| \mu_j^{m-2} - \mu_i^{m-1} \right| \leq \delta_{m-1} \right\} \right\} \\ = & \begin{cases} \Pr \left[\underline{B}_{ij} \leq s(k) \leq \bar{B}_{ij} \right], & \text{if } \underline{B}_{ij} \leq \bar{B}_{ij}, \left| \mu_j^1 - \mu_i^2 \right| \leq \delta_2, \dots, \\ & \text{and } \left| \mu_j^{m-2} - \mu_i^{m-1} \right| \leq \delta_{m-1} \\ 0 & \text{otherwise} \end{cases} \end{aligned} \quad (31)$$

The division is not performed on the κ_m -axis because $\kappa_m(k-1) = \zeta_j$ does not appear in (30) during the transition. The initial conditions of the probability vector, $\pi(0)$, are set as a vector with all elements equal to zero, except $\pi_{\text{ini}}(0)$ is set to 1, where $\pi_{\text{ini}}(0)$ corresponds to state S_{ini} containing $\kappa(0) = [v \ \dots \ v]^T$. MTF can be obtained through (17), and then the procedures described at the end of Section 3 can be applied to calculate the threshold.