



Analytic Confusion Matrix Bounds for Fault Detection and Isolation Using a Sum-of-Squared- Residuals Approach

Dan Simon
Cleveland State University, Cleveland, Ohio

Donald L. Simon
Glenn Research Center, Cleveland, Ohio

NASA STI Program . . . in Profile

Since its founding, NASA has been dedicated to the advancement of aeronautics and space science. The NASA Scientific and Technical Information (STI) program plays a key part in helping NASA maintain this important role.

The NASA STI Program operates under the auspices of the Agency Chief Information Officer. It collects, organizes, provides for archiving, and disseminates NASA's STI. The NASA STI program provides access to the NASA Aeronautics and Space Database and its public interface, the NASA Technical Reports Server, thus providing one of the largest collections of aeronautical and space science STI in the world. Results are published in both non-NASA channels and by NASA in the NASA STI Report Series, which includes the following report types:

- **TECHNICAL PUBLICATION.** Reports of completed research or a major significant phase of research that present the results of NASA programs and include extensive data or theoretical analysis. Includes compilations of significant scientific and technical data and information deemed to be of continuing reference value. NASA counterpart of peer-reviewed formal professional papers but has less stringent limitations on manuscript length and extent of graphic presentations.
- **TECHNICAL MEMORANDUM.** Scientific and technical findings that are preliminary or of specialized interest, e.g., quick release reports, working papers, and bibliographies that contain minimal annotation. Does not contain extensive analysis.
- **CONTRACTOR REPORT.** Scientific and technical findings by NASA-sponsored contractors and grantees.

- **CONFERENCE PUBLICATION.** Collected papers from scientific and technical conferences, symposia, seminars, or other meetings sponsored or cosponsored by NASA.
- **SPECIAL PUBLICATION.** Scientific, technical, or historical information from NASA programs, projects, and missions, often concerned with subjects having substantial public interest.
- **TECHNICAL TRANSLATION.** English-language translations of foreign scientific and technical material pertinent to NASA's mission.

Specialized services also include creating custom thesauri, building customized databases, organizing and publishing research results.

For more information about the NASA STI program, see the following:

- Access the NASA STI program home page at <http://www.sti.nasa.gov>
- E-mail your question via the Internet to help@sti.nasa.gov
- Fax your question to the NASA STI Help Desk at 443-757-5803
- Telephone the NASA STI Help Desk at 443-757-5802
- Write to:
NASA Center for AeroSpace Information (CASI)
7115 Standard Drive
Hanover, MD 21076-1320



Analytic Confusion Matrix Bounds for Fault Detection and Isolation Using a Sum-of-Squared-Residuals Approach

Dan Simon
Cleveland State University, Cleveland, Ohio

Donald L. Simon
Glenn Research Center, Cleveland, Ohio

National Aeronautics and
Space Administration

Glenn Research Center
Cleveland, Ohio 44135

Level of Review: This material has been technically reviewed by technical management.

Available from

NASA Center for Aerospace Information
7115 Standard Drive
Hanover, MD 21076-1320

National Technical Information Service
5285 Port Royal Road
Springfield, VA 22161

Available electronically at <http://gltrs.grc.nasa.gov>

Analytic Confusion Matrix Bounds for Fault Detection and Isolation Using a Sum-of-Squared-Residuals Approach

Dan Simon
Cleveland State University
Cleveland, Ohio 44115

Donald L. Simon
National Aeronautics and Space Administration
Glenn Research Center
Cleveland, Ohio 44135

Abstract

Given a system which can fail in 1 of n different ways, a fault detection and isolation (FDI) algorithm uses sensor data in order to determine which fault is the most likely to have occurred. The effectiveness of an FDI algorithm can be quantified by a confusion matrix, which indicates the probability that each fault is isolated given that each fault has occurred. Confusion matrices are often generated with simulation data, particularly for complex systems. In this paper we perform FDI using sums of squares of sensor residuals (SSRs). We assume that the sensor residuals are Gaussian, which gives the SSRs a chi-squared distribution. We then generate analytic lower and upper bounds on the confusion matrix elements. This allows for the generation of optimal sensor sets without numerical simulations. The confusion matrix bounds are verified with simulated aircraft engine data.

1 Introduction

Many different methods of fault detection and isolation (FDI) have been proposed. Frequency domain methods include monitoring resonances [1] or modes [2]. Filter-based methods include observers [3], unknown input observers [4], Kalman filters [5], particle filters [6], sliding mode observers [7], H_∞ filters [8], and set membership filters [9]. There are also methods based on computer intelligence [10], include fuzzy logic [11], neural networks [12], genetic algorithms [13], and expert systems [14]. Other methods include those based on Markov models [15], system identification [16], wavelets [17], Bayesian inference [18], control input manipulation [19], and the parity space approach [20]. Many other FDI methods have also been proposed [21], some of which apply to special types of systems.

The parity space approach to FDI compares the sensor residual vector to nominal user-specified fault vectors, and the closest fault vector is isolated as the most likely fault. If the

sensor residual vectors are Gaussian, the parity space approach allows an analytic computation of the confusion matrix. The FDI approach that we propose is philosophically similar to the parity space approach, but instead of using fault vectors, we use sum-of-squared residuals (SSRs) in order to detect and isolate a fault. We do not compare our approach with other FDI methods, but our approach is chosen because of its amenability to a new statistical method for the calculation of confusion matrix bounds.

If sensor residuals are Gaussian, the SSRs have a chi-squared distribution [22]. This allows for the specification of SSR bounds for fault detection which have a known false negative rate and false positive rate. We can also compare the SSRs for each fault type to determine which fault is most likely to have occurred, and then find analytic bounds for fault isolation probabilities. Our FDI algorithm is new, but the primary contribution of this paper is to show how confusion matrix element bounds can be derived analytically. The FDI algorithm that we propose is fairly simple, but the confusion matrix analysis that we develop is novel and its ideas may be adaptable to other FDI algorithms.

Our approach is to first specify the magnitude of each fault that we want to detect, along with a target false positive rate (FPR). For each fault we then find the sensor set that gives the largest true positive rate (TPR) for the given FPR. Then we use statistical approaches to find confusion matrix bounds. The confusion matrix bounds are the outputs of this process. We cannot specify desired confusion matrix bounds ahead of time; the bounds are the dependent variables of the sensor selection process.

The goal of this paper is threefold. Our first goal is to present our SSR-based FDI algorithm, which we do in Section 2. Our second goal is to derive confusion matrix bounds, which we do in Section 3. Our third goal is to confirm the theory with simulation results, which we do in Section 4 using an aircraft turbofan engine model. Section 5 presents some discussion and conclusions.

2 A Simple SSR-Based FDI Algorithm

This section presents an overview of our simple SSR-based FDI algorithm, gives an overview of confusion matrix theory, and provides some fundamental lemmas that are used later. We consider FDI of a static, linear system. Sensor residuals are computed at each measurement time to perform fault detection, and the sum of the squares of the sensor residuals (SSR) are used to perform fault isolation. If the sensor residuals are Gaussian, then the SSRs have a chi-squared distribution, which allows the formulation of analytic bounds on the confusion matrix elements as discussed in Sections 3.1-3.3.

The residual of the i th sensor is denoted as y_i and is a measurement of the difference between the sensor output and its nominal output in the no-fault case. In the no-fault case, y_i is zero mean. In the fault case, the mean of y_i is μ_i . In either case, the standard deviation of y_i is σ_i . The mean μ_i depends on which fault occurs, but for simplicity of notation we do

not indicate that dependence in our notation. An SSR is given as

$$S = \sum_{i=1}^k (y_i/\sigma_i)^2 \quad (1)$$

where k is the number of sensors used in this particular SSR. If each y_i is a zero-mean Gaussian random variable, then S is a random variable with a chi-squared distribution [22]. Its probability density function (pdf), and cumulative distribution function (cdf), are

$$f(x, k) = \begin{cases} 2^{-k/2}(\Gamma(k/2))^{-1}x^{k/2-1}e^{-x/2} & x > 0 \\ 0 & x \leq 0 \end{cases} \quad (2)$$

$$F(x, k) = \begin{cases} \gamma(k/2, x/2)(\Gamma(k/2))^{-1} & x > 0 \\ 0 & x \leq 0 \end{cases} \quad (3)$$

where $\Gamma(\cdot)$ is the gamma function and $\gamma(\cdot)$ is the lower incomplete gamma function.

$$\Gamma(z) = \int_0^{\infty} t^{z-1}e^{-t} dt \quad (4)$$

$$\gamma(z, a) = \int_0^a t^{z-1}e^{-t} dt \quad (5)$$

We will use a user-specified threshold T to detect a fault.

$$S \geq T \rightarrow \text{fault detected} \quad (6)$$

$$S < T \rightarrow \text{no fault detected} \quad (7)$$

Note that fault isolation is a different issue than fault detection. Detection means that $S_i \geq T_i$ for fault detection algorithm i . However, it may be that $S_i \geq T_i$ for more than one value of i . In that case multiple faults have been detected and a fault isolation algorithm is required to isolate the most likely fault.

If a fault occurs, then the y_i terms in Equation 1 will not, in general, be zero mean. In this case S has a noncentral chi-squared distribution [22]. Its pdf and cdf are given as

$$f(x, k, \lambda) = \frac{1}{2}e^{-(x+\lambda)/2}(x/\lambda)^{k/4-1/2}I_{k/2-1}(\sqrt{\lambda x}) \quad (8)$$

$$F(x, k, \lambda) = \sum_{j=0}^{\infty} e^{-\lambda/2} \frac{(\lambda/2)^j}{j!} F(x, k + j/2) \quad (9)$$

where λ is related to the means μ_i of the sensor residuals y_i , and $I(\cdot)$ is a modified Bessel function of the first kind.

$$\lambda = \sum_{i=1}^k \left(\frac{\mu_i}{\sigma_i} \right)^2 \quad (10)$$

$$I_m(y) = (y/2)^m \sum_{j=0}^{\infty} \frac{(y^2/4)^j}{j! \Gamma(m + j + 1)} \quad (11)$$

We use the notation $f_x(w)$ and $F_x(w)$ to denote the pdf and cdf of the random variable x evaluated at w . In terms of FDI, w will be equal to the SSR. If the random variable that is being used is clear from the surrounding text and mathematics, we shorten the notation to $f(x)$ and $F(x)$. We use $f(x, k)$ and $F(x, k)$ to denote the pdf and cdf of the chi-squared distribution, and $f(x, k, \lambda)$ and $F(x, k, \lambda)$ to denote the pdf and cdf of the noncentral chi-squared distribution.

A confusion matrix specifies the likelihood of isolating each fault, and can be used to quantify the performance of an FDI algorithm. A typical confusion matrix for FDI is shown in Table 1. The rows correspond to fault conditions, and the columns correspond to fault isolation results. The element in the i th row and j th column is the probability that fault j is isolated when fault i occurs. Ideally the confusion matrix would be an identity matrix, which would indicate perfect fault isolation.

In Table 1, CCR_i is the probability that fault i is correctly isolated given that it occurs. CNR is the probability that a no-fault condition is correctly indicated given that no fault occurs. M_{ij} is the probability that fault i is incorrectly isolated given that fault j occurs. M_{i0} is the probability that fault i is incorrectly isolated given that no fault occurs. M_{0i} is the probability that no fault is isolated given that fault i occurs.

Table 1: Typical confusion matrix format. The rows correspond to fault conditions, and the columns correspond to fault isolation results.

	Fault 1	Fault 2	...	Fault n	No fault
Fault 1	CCR_1	M_{21}	...	M_{n1}	M_{01}
Fault 2	M_{12}	CCR_2	...	M_{n2}	M_{02}
...
Fault n	M_{1n}	M_{2n}	...	CCR_n	M_{0n}
No fault	M_{10}	M_{20}	...	M_{n0}	CNR

In the remainder of this paper we use the following lemmas to derive results. These lemmas can be proven using standard definitions and results derived from probability theory [23, 24].

Lemma 1 *The probability that a realization of the random variable x is greater than a realization of the random variable y is given as*

$$P(x > y) = \int_{-\infty}^{\infty} \int_y^{\infty} f(x, y) dx dy \quad (12)$$

where $f(x, y)$ is the joint pdf of x and y . If x and y are independent, this can be written in terms of the marginal pdf's as

$$P(x > y) = \int_{-\infty}^{\infty} \left[\int_z^{\infty} f_x(w) dw \right] f_y(z) dz \quad (13)$$

$$= \int_{-\infty}^{\infty} [1 - F_x(z)] f_y(z) dz \quad (14)$$

Lemma 2 *If $y = T + x$, where x is a random variable and T is a constant, then*

$$f_y(w) = f_x(w - T) \quad (15)$$

$$F_y(w) = F_x(w - T) \quad (16)$$

Lemma 3 *If $y = T - x$, where x is a random variable and T is a constant, then*

$$f_y(w) = f_x(T - w) \quad (17)$$

$$F_y(w) = 1 - F_x(T - w) \quad (18)$$

Lemma 4 *If $z = \min(x, y)$, where x and y are independent random variables, then*

$$f_z(w) = f_x(w)(1 - F_y(w)) + f_y(w)(1 - F_x(w)) \quad (19)$$

Lemma 5 *If $z = \min(x, T)$, where x is a random variable and T is a constant, then*

$$F_z(w) = \begin{cases} F_x(w) & w < T \\ 1 & w \geq T \end{cases} \quad (20)$$

$$f_z(w) = \begin{cases} f_x(w) & w < T \\ 0 & w > T \\ (1 - F_x(w))\delta(0) & w = T \end{cases} \quad (21)$$

where $\delta(\cdot)$ is the continuous-time impulse function.

Lemma 6 *If $z = \max(x, y)$, where x and y are independent random variables, then*

$$f_z(w) = f_x(w)F_y(w) + f_y(w)F_x(w) \quad (22)$$

Lemma 7 *If $z = \max(x, T)$, where x is a random variable and T is a constant, then*

$$F_z(w) = \begin{cases} F_x(w) & w \geq T \\ 0 & w < T \end{cases} \quad (23)$$

$$f_z(w) = \begin{cases} f_x(w) & w > T \\ 0 & w < T \\ F_x(w)\delta(0) & w = T \end{cases} \quad (24)$$

3 Confusion Matrix Bounds

This section derives analytic confusion matrix bounds for our SSR-based FDI algorithm. Section 3.1 deals with the no-fault case and derives bounds for the correct no-fault rate (CNR), which is the probability that no fault is detected given that no fault occurs. It also derives bounds for the FPR, which is the probability that at least one fault is detected given that no fault occurred. Finally it derives an upper bound for the no-fault misclassification rate, which is the probability that a given fault is isolated given that no fault occurred. Section 3.2 deals with the fault case and derives bounds for the correction classification rate (CCR), which is the probability that a given fault is correctly isolated given that it occurred. Section 3.3 also deals with the fault case and derives upper bounds for the fault misclassification rate, which is the probability that the incorrect fault is isolated given that some other fault occurred. Section 3.4 summarizes the bounds and their use in the confusion matrix, and Section 3.5 discusses the required computational effort.

3.1 No-Fault Case

The true negative rate (TNR) is the probability that $S < T$ given that there are no faults. The false positive rate (FPR) is the probability that $S \geq T$ given that there are no faults. If only one SSR is considered, these probabilities are given as

$$\text{TNR}(T, k) = F(T, k) \tag{25}$$

$$\text{FPR}(T, k) = 1 - F(T, k) \tag{26}$$

Figure 1 illustrates TNR and FPR for a chi-squared SSR containing $k = 10$ sensors and a detection threshold $T = 25$.

3.1.1 False positive rate: Two fault detection algorithms

Our FDI algorithm will have n fault detection algorithms running in parallel, where n is the number of possible faults. This will allow for the calculation of n SSRs, which will allow for their comparisons for fault isolation.

In this section we derive the FPR if two fault detection algorithms are running in parallel. Suppose that algorithm 1 attempts to detect fault 1 using k_{1a} sensors and threshold T_1 , and algorithm 2 attempts to detect fault 2 using k_{2a} sensors and threshold T_2 . We use the

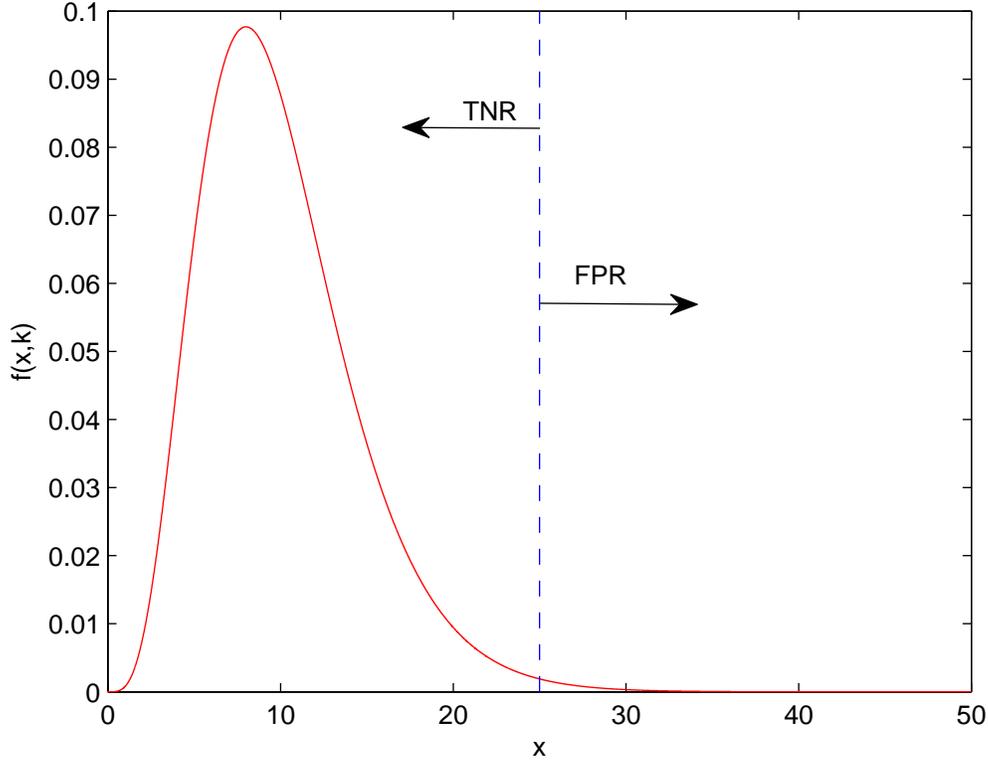


Figure 1: Illustration of the chi-squared pdf of an SSR with $k = 10$ sensors. The true negative rate is the area to the left of the user-specified threshold $T = 25$, and the false positive rate is the area to the right of the threshold.

notation

$$Y_1 = \{\text{sensors unique to algorithm 1}\} \quad (27)$$

$$Y_2 = \{\text{sensors unique to algorithm 2}\} \quad (28)$$

$$Y_c = \{\text{sensors common to algorithms 1 and 2}\} \quad (29)$$

$$\mathcal{Y}_1 = \{\text{all sensors used by algorithm 1}\} \quad (30)$$

$$= \{Y_1, Y_c\} \quad (31)$$

$$\mathcal{Y}_2 = \{\text{all sensors used by algorithm 2}\} \quad (32)$$

$$= \{Y_2, Y_c\} \quad (33)$$

Note that some of the sensors used to detect fault 1 might also be used to detect fault 2, but some of the sensors used to detect fault 1 might not be used for fault 2 (and vice versa). Our algorithm therefore takes into account those sensor residuals that are unique to a given fault, and also those residuals that are common between faults. We use the notation \mathcal{Y}_{1i} to

denote the i th normalized residual of the sensors used in algorithm 1, with similar meanings for \mathcal{Y}_{2i} , Y_{1i} , Y_{2i} , and Y_{ci} . That is,

$$S_1 = \sum \mathcal{Y}_{1i}^2 \quad (34)$$

$$= \sum Y_{1i}^2 + \sum Y_{ci}^2 \quad (35)$$

If there are no faults, then the probability that either fault 1 or fault 2 will be detected is

$$\text{FPR} = P \left[\left(\sum \mathcal{Y}_{1i}^2 > T_1 \right) \text{ or } \left(\sum \mathcal{Y}_{2i}^2 > T_2 \right) \right] \quad (36)$$

$$= P \left[\left(\sum Y_{1i}^2 + \sum Y_{ci}^2 > T_1 \right) \text{ or } \left(\sum Y_{2i}^2 + \sum Y_{ci}^2 > T_2 \right) \right] \quad (37)$$

$$= P \left[\left(\sum Y_{ci}^2 > T_1 - \sum Y_{1i}^2 \right) \text{ or } \left(\sum Y_{ci}^2 > T_2 - \sum Y_{2i}^2 \right) \right] \quad (38)$$

$$= P \left[\sum Y_{ci}^2 > \min \left(T_1 - \sum Y_{1i}^2, T_2 - \sum Y_{2i}^2 \right) \right] \quad (39)$$

Note that $\sum Y_{ci}^2$, $\sum Y_{1i}^2$, and $\sum Y_{2i}^2$ are three independent random variables. If neither Y_1 , Y_2 , nor Y_c are empty, we can use Lemmas 1, 3, and 4 to obtain

$$\text{FPR} = \int_{-\infty}^{\infty} [1 - F(y, k_c)] [f(T_1 - y, k_1)F(T_2 - y, k_2) + f(T_2 - y, k_2)F(T_1 - y, k_1)] dy \quad (40)$$

$$k_1 = |Y_1|, \quad k_2 = |Y_2|, \quad k_c = |Y_c| \quad (41)$$

Note that we are using the notation $|Z|$ to denote the number of elements in the set Z . If Y_1 is empty (sensor set 1 does not have any unique sensors), but Y_2 and Y_c are not empty, Equation 39 can be written as

$$\text{FPR} = P \left[\sum Y_{ci}^2 > \min \left(T_1, T_2 - \sum Y_{2i}^2 \right) \right] \quad (42)$$

$$= P \left[\sum Y_{ci}^2 > \min(T_1, a) \right] \quad (43)$$

where a is defined by the above equation. Lemma 3 tells us that

$$f_a(y) = f(T_2 - y, k_2) \quad (44)$$

$$F_a(y) = 1 - F(T_2 - y, k_2) \quad (45)$$

Equations 42 and 43 can be written as

$$\text{FPR} = P \left[\sum Y_{ci}^2 > b \right] \quad (46)$$

$$b = \min(T_1, a) \quad (47)$$

Lemma 5 tells us that

$$f_b(y) = \begin{cases} f_a(y) & y < T_1 \\ 0 & y > T_1 \\ (1 - F_a(y))\delta(0) & y = T_1 \end{cases} \quad (48)$$

We can use Lemma 1 in conjunction with Equations 42–48 to write

$$FPR = \int_{-\infty}^{\infty} [1 - F(y, k_c)] f_b(y) dy \quad (49)$$

$$= \int_{-\infty}^{T_1} [1 - F(y, k_c)] f_a(y) dy + \int_{T_1}^{T_1} [1 - F(y, k_c)] [1 - F_a(y)] \delta(0) dy \quad (50)$$

$$= \int_{-\infty}^{T_1} [1 - F(y, k_c)] f(T_2 - y, k_2) dy + [1 - F(T_1, k_c)] [1 - F_a(T_1)] \quad (51)$$

$$= \int_{-\infty}^{T_1} [1 - F(y, k_c)] f(T_2 - y, k_2) dy + [1 - F(T_1, k_c)] F(T_2 - T_1, k_2) \quad (52)$$

$$= \int_{-\infty}^{\min(T_1, T_2)} [1 - F(y, k_c)] f(T_2 - y, k_2) dy + [1 - F(T_1, k_c)] F(T_2 - T_1, k_2) \quad (53)$$

If Y_2 is empty (sensor set 2 does not have any unique sensors), but Y_1 and Y_c are not empty, we can use Lemmas 1, 3, and 5 to obtain

$$FPR = P \left[\sum Y_{ci}^2 > \min \left(T_1 - \sum Y_{1i}^2, T_2 \right) \right] \quad (54)$$

$$= \int_{-\infty}^{\min(T_1, T_2)} [1 - F(y, k_c)] f(T_1 - y, k_1) dy + [1 - F(T_2, k_c)] F(T_1 - T_2, k_1) \quad (55)$$

If Y_c is empty (the two sensor sets do not have any common sensors), but Y_1 and Y_2 are not empty, we can use Lemmas 3 and 4 to obtain

$$FPR = P \left[0 > \min \left(T_1 - \sum Y_{1i}^2, T_2 - \sum Y_{2i}^2 \right) \right] \quad (56)$$

$$= \int_{-\infty}^0 [f(T_1 - y, k_1)F(T_2 - y, k_2) + f(T_2 - y, k_2)F(T_1 - y, k_1)] dy \quad (57)$$

3.1.2 False positive rate: More than two fault detection algorithms

The previous section derived the FPR given that two fault detection algorithms are running in parallel. Now suppose that there are $n > 2$ fault detection algorithms. In this case we can write

$$FPR = P [(S_1 > T_1) \text{ or } \dots \text{ or } (S_n > T_n)] \quad (58)$$

$$= 1 - P [(S_1 < T_1), \dots, (S_n < T_n)] \quad (59)$$

$$= 1 - \text{CNR} \quad (60)$$

where CNR is the correct no-fault rate. CNR is the probability that all of the SSRs are below their detection thresholds given that no fault occurred, and can be written as

$$\text{CNR} = P [(S_1 < T_1), \dots, (S_n < T_n)] \quad (61)$$

This can be bounded as

$$\prod_{i=1}^n \text{TNR}(T_i, k_i) \leq \text{CNR} \leq \min_i \text{TNR}(T_i, k_i) \quad (62)$$

The lower bound will be exact if none of the fault detection algorithms have any sensors in common, which means that $\text{TNR}(T_i, k_i)$ and $\text{TNR}(T_j, k_j)$ are independent for all $i \neq j$. The upper bound will be exact if there is some m such that \mathcal{Y}_m is a superset of \mathcal{Y}_i for all $i \neq m$, and $\text{TNR}(T_m, k_m) \leq \text{TNR}(T_i, k_i)$ for all $i \neq m$.

We can use Equations 60 and 62 to obtain bounds for the FPR when $n > 2$ fault detection algorithms are running in parallel.

$$1 - \min_i \text{TNR}(T_i, k_i) \leq \text{FPR} \leq 1 - \prod_{i=1}^n \text{TNR}(T_i, k_i) \quad (63)$$

3.1.3 Fault misclassification rates in the no-fault case

Given that no fault occurred, the probability that fault i is incorrectly isolated is called the misclassification rate, M_{i0} . In this section we derive upper bounds for this probability.

Suppose that we have n fault detectors running in parallel. Given that a fault is detected, that is, that the SSR of at least one fault detector exceeds its threshold, we propose isolating the fault using the following logic.

$$\hat{p} = \operatorname{argmax}_p \left(\sum_i \mathcal{Y}_{pi}^2 - T_p \right) \quad (64)$$

Two fault detection algorithms

Suppose that we have only two fault detection algorithms, algorithms 1 and 2. Given that no fault occurred, the probability that fault 1 is isolated is called the marginal misclassification of fault 1 relative to fault 2, and is given as

$$M_{10,2} = P[(S_1 > T_1), (S_1 - T_1 > S_2 - T_2)] \quad (65)$$

$$= P \left[\sum_i Y_{1i}^2 > \max \left(T_1 - \sum_i Y_{ci}^2, \sum_i Y_{2i}^2 + T_1 - T_2 \right) \right] \quad (66)$$

If neither Y_1 , Y_2 , nor Y_c are empty, we can use Lemmas 1, 2, 3, and 6 to obtain

$$M_{10,2} = \int_{-\infty}^{\infty} [1 - F(y, k_1)] [F(y + T_2 - T_1, k_2) f(T_1 - y, k_c) + f(y + T_2 - T_1, k_2) (1 - F(T_1 - y, k_c))] dy \quad (67)$$

In order to find finite integration limits that do not result in an excessive loss of accuracy in the computation of $M_{10,2}$, we use our knowledge that $f(w, k) = 0$ for $w < 0$, and we find

values of z_2 and z_c such that $f(z_2, k_2) < \epsilon$ and $f(z_c, k_c) < \epsilon$ for some small user-specified threshold ϵ . This gives

$$M_{10,2} = \int_L^U [1 - F(y, k_1)] [F(y + T_2 - T_1, k_2) f(T_1 - y, k_c) + f(y + T_2 - T_1, k_2) (1 - F(T_1 - y, k_c))] dy \quad (68)$$

$$L = \min(T_1 - T_2, T_1 - z_c) \quad (69)$$

$$U = \max(T_1, z_2 + T_1 - T_2) \quad (70)$$

If Y_1 is empty (sensor set 1 does not have any unique sensors), but Y_2 and Y_c are not empty, the marginal misclassification rate can be derived as

$$M_{10,2} = P \left[0 > \max \left(\sum_i Y_{2i}^2 + T_1 - T_2, T_1 - \sum_i Y_{ci}^2 \right) \right] \quad (71)$$

$$= P \left(\sum_i Y_{2i}^2 < T_2 - T_1 \right) P \left(\sum_i Y_{ci}^2 > T_1 \right) \quad (72)$$

$$= F(T_2 - T_1, k_2) [1 - F(T_1, k_c)] \quad (73)$$

If Y_2 is empty (sensor set 2 does not have any unique sensors), but Y_1 and Y_c are not empty, the marginal misclassification rate can be derived as

$$M_{10,2} = P \left[\sum_i Y_{1i}^2 > \max \left(T_1 - \sum_i Y_{ci}^2, T_1 - T_2 \right) \right] \quad (74)$$

We can use Lemmas 1, 3, and 7 to write this equation as

$$M_{10,2} = \int_{-\infty}^{\infty} [1 - F(y, k_1)] [f(T_1 - y, k_c) + (1 - F(T_1 - y, k_c)) \delta(y - T_1 + T_2)] dy \quad (75)$$

$$= [1 - F(T_1 - T_2, k_1)] [1 - F(T_2, k_c)] + \int_{T_1 - T_2}^{T_1} [1 - F(y, k_1)] f(T_1 - y, k_c) dy \quad (76)$$

If Y_c is empty (the two sensor sets do not have any common sensors), but Y_1 and Y_2 are not empty, the marginal misclassification rate can be derived as

$$M_{10,2} = P \left[\sum_i Y_{1i}^2 > \max \left(\sum_i Y_{2i}^2 + T_1 - T_2, T_1 \right) \right] \quad (77)$$

We can use Lemmas 1, 2, and 7 to write this equation as

$$M_{10,2} = \int_{-\infty}^{\infty} [1 - F(y, k_1)] [f(y + T_2 - T_1, k_2) + F(y + T_2 - T_1, k_2) \delta(y - T_1)] dy \quad (78)$$

$$= [1 - F(T_1, k_1)] F(T_2, k_2) + \int_{T_1}^{\infty} [1 - F(y, k_1)] f(y + T_2 - T_1, k_2) dy \quad (79)$$

In order to find finite integration limits that do not result in an excessive loss of accuracy in the computation of $M_{10,2}$, we find z such that $f(z, k_2) < \epsilon$ for some small user-specified threshold ϵ . This gives

$$M_{10,2} = [1 - F(T_1, k_1)] F(T_2, k_2) + \int_{T_1}^{z+T_1-T_2} [1 - F(y, k_1)] f(y + T_2 - T_1, k_2) dy \quad (80)$$

Given that no fault occurred and there are only two fault detection algorithms, the probability that fault 1 is isolated is given by Equation 68, 73, 76, or 80.

More than two fault detection algorithms

If we have $n > 2$ fault detection algorithms, the probability that fault 1 is isolated given that no fault occurred is the probability that SSR_1 is greater than its threshold, and greater than all of the other SSRs relative to their thresholds.

$$\begin{aligned} M_{10} &= P[(S_1 > T_1), (S_1 - T_1 > S_2 - T_2), \dots, (S_1 - T_1 > S_n - T_n)] & (81) \\ &\leq \min_i M_{10,i} & (82) \end{aligned}$$

So in order to obtain an upper bound for the misclassification rate of fault 1 given that no fault occurred, we use Equations 68, 73, 76, or 80 as appropriate to find the marginal misclassification rates. Then we use Equation 82 to find an upper bound for M_{10} .

3.2 Correct Fault Classification Rates

The true positive rate (TPR) is defined as the probability that a fault is correctly detected ($S > T$) given that it occurs. This does not take fault isolation into account. It only considers the probability that the SSR is greater than its detection threshold. The false negative rate (FNR) is defined as the probability that no fault is detected ($S < T$) given that a fault occurs. If only one SSR is considered, these probabilities can be written as

$$FNR_i = F(T_i, k_{ia}, \lambda_{ia}) \quad (83)$$

$$TPR_i = 1 - F(T_i, k_{ia}, \lambda_{ia}) \quad (84)$$

where k_{ia} is the total number of sensors in \mathcal{Y}_i , and λ_{ia} is the non-centrality parameter of S_i given that fault i occurred. Note that $k_{ia} \geq k_i$ because k_i is the number of unique sensors in \mathcal{Y}_i , but k_{ia} is the total number of sensors in \mathcal{Y}_i . Figure 2 illustrates TPR and FNR for a chi-squared SSR containing $k = 10$ sensors, $\lambda = 40$, and a detection threshold $T = 20$.

Given that some fault occurs, we might isolate the correct fault or we might isolate an incorrect fault. The probability of isolating the correct fault is called the correct classification rate (CCR). In this section we derive lower and upper bounds for the CCR.

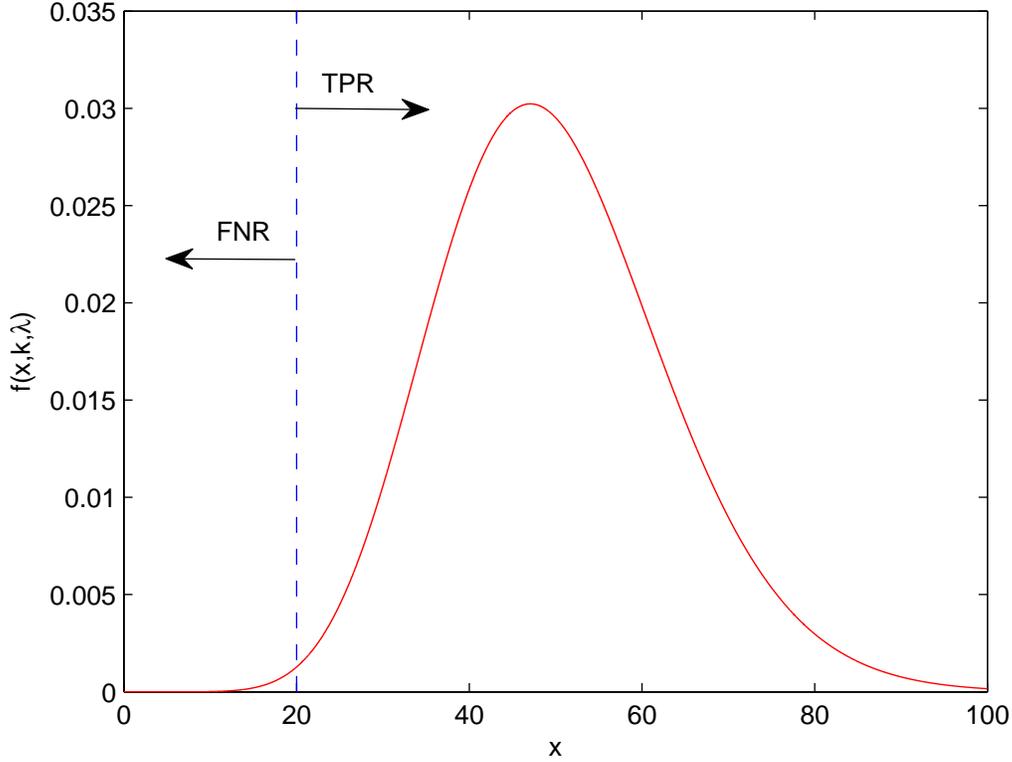


Figure 2: Illustration of the noncentral chi-squared pdf of an SSR with $k = 10$ sensors and $\lambda = 40$. The false negative rate is the area to the left of the user-specified threshold $T = 20$, and the true positive rate is the area to the right of the threshold.

3.2.1 Lower bounds for the correct classification rate

Suppose we have only two fault detectors, algorithms 1 and 2, and fault 1 occurs. Consider the probability that SSR_1 is larger than SSR_2 relative to their thresholds. We call this the marginal detection rate D_{12} . Note that we are not considering whether or not the SSRs exceed their threshold. We are only considering how large the SSRs are relative to their thresholds. The marginal detection rate is given as

$$D_{12} = P[(S_1 - T_1) > (S_2 - T_2)] \quad (85)$$

$$= P\left[\left(\sum_i Y_{1i}^2 - T_1 + T_2\right) > \sum_i Y_{2i}^2\right] \quad (86)$$

Assuming that neither Y_1 nor Y_2 are empty (that is, both sensor sets have some unique sensors), we can use Lemmas 1 and 2 to obtain

$$D_{12} = \int_{-\infty}^{\infty} \left(\int_y^{\infty} f(x + T_1 - T_2, k_1, \lambda_1) dx \right) f(y, k_2) dy \quad (87)$$

$$= \int_0^{\infty} (1 - F(y + T_1 - T_2, k_1, \lambda_1)) f(y, k_2) dy \quad (88)$$

where

$$\lambda_1 = \sum Y_{1i}^2 \quad (89)$$

If Y_1 is empty (sensor set 1 does not have any unique sensors), but Y_2 is not empty, the marginal detection rate can be derived as

$$D_{12} = P \left(\sum_i Y_{2i}^2 < T_2 - T_1 \right) \quad (90)$$

$$= F(T_2 - T_1, k_2) \quad (91)$$

If Y_2 is empty (sensor set 2 does not have any unique sensors), but Y_1 is not empty, the marginal detection rate can be derived as

$$D_{12} = P \left(\sum_i Y_{1i}^2 > T_1 - T_2 \right) \quad (92)$$

$$= 1 - F(T_1 - T_2, k_1, \lambda_1) \quad (93)$$

Now consider the case where there are $n > 2$ fault detection algorithms. Given that fault 1 occurred, the probability that SSR_1 is larger than SSR_i for all $i > 1$, relative to their thresholds, is given as

$$D_1 = P[(S_1 - T_1 > S_2 - T_2), \dots, (S_1 - T_1 > S_n - T_n)] \quad (94)$$

$$\geq \prod_{i=2}^n D_{1i} \quad (95)$$

where the inequality arises because the events might not be independent. If none of the fault detection algorithms have any sensors in common, then the events are independent and the bound is exact.

Now consider the probability that fault 1 is isolated given that fault 1 occurred. This can be written as

$$CCR_1 = P[(S_1 > T_1), (S_1 - T_1 > S_2 - T_2), \dots, (S_1 - T_1 > S_n - T_n)] \quad (96)$$

$$\geq TPR_1 D_1 \quad (97)$$

where the inequality arises because the events might not be independent. Equation 97 gives a lower bound for the correct classification rate for fault 1.

3.2.2 Upper bounds for the correction classification rate

Next we find an upper bound for the CCR. To begin, suppose that we have only two fault detectors, algorithms 1 and 2. Given that fault 1 occurs, the probability that it is correctly isolated is called the marginal CCR. This can be written as

$$\text{CCR}_{12} = P[(S_1 - T_1 > S_2 - T_2), (S_1 > T_1)] \quad (98)$$

$$= P \left[\sum_i Y_{1i}^2 > \max \left(\sum_i Y_{2i}^2 + T_1 - T_2, T_1 - \sum_i Y_{ci}^2 \right) \right] \quad (99)$$

Suppose that both sensor sets have some unique sensors, and the sensor sets also have some common sensors. That is, neither Y_1 , Y_2 , nor Y_c are empty. Then we can use Lemmas 1, 2, 3, and 6 to obtain

$$\text{CCR}_{12} = \int_{-\infty}^{\infty} [1 - F(y, k_1, \lambda_1)] [F(y + T_2 - T_1, k_2) f(T_1 - y, k_c, \lambda_c) + f(y + T_2 - T_1, k_2) (1 - F(T_1 - y, k_c, \lambda_c))] dy \quad (100)$$

where λ_c is defined analogously to λ_1 (see Equation 89). In order to limit the integration interval while maintaining accuracy in the computation of CCR_{12} , we use our knowledge that $f(w, k)$ and $f(w, k, \lambda)$ are both zero for $w < 0$, and we numerically find values of z_2 and z_c such that $f(z_2, k_2) < \epsilon$ and $f(z_c, k_c, \lambda_c) < \epsilon$ for some small user-specified threshold ϵ . This gives

$$\text{CCR}_{12} = \int_L^U [1 - F(y, k_1, \lambda_1)] [F(y + T_2 - T_1, k_2) f(T_1 - y, k_c, \lambda_c) + f(y + T_2 - T_1, k_2) (1 - F(T_1 - y, k_c, \lambda_c))] dy \quad (101)$$

$$L = \min(T_1 - T_2, T_1 - z_c) \quad (102)$$

$$U = \max(T_1, z_2 + T_1 - T_2) \quad (103)$$

If Y_1 is empty (sensor set 1 does not have any unique sensors), but Y_2 and Y_c are not empty, the marginal CCR can be derived as

$$\text{CCR}_{12} = P \left[0 > \max \left(\sum_i Y_{2i}^2 + T_1 - T_2, T_1 - \sum_i Y_{ci}^2 \right) \right] \quad (104)$$

$$= P \left(\sum_i Y_{2i}^2 < T_2 - T_1 \right) P \left(\sum_i Y_{ci}^2 > T_1 \right) \quad (105)$$

$$= F(T_2 - T_1, k_2) [1 - F(T_1, k_c, \lambda_c)] \quad (106)$$

If Y_2 is empty (sensor set 2 does not have any unique sensors), but Y_1 and Y_c are not empty, the marginal CCR can be derived as

$$\text{CCR}_{12} = P \left[\left(\sum_i Y_{1i}^2 > T_1 - T_2 \right), \left(\sum_i Y_{1i}^2 > T_1 - \sum_i Y_{ci}^2 \right) \right] \quad (107)$$

$$= P \left[\sum_i Y_{1i}^2 > \max \left(T_1 - \sum_i Y_{ci}^2, T_1 - T_2 \right) \right] \quad (108)$$

We can use Lemmas 1, 3, and 7 to write this equation as

$$\text{CCR}_{12} = \int_{T_1-T_2}^{\infty} [1 - F(y, k_1, \lambda_1)] [f(T_1 - y, k_c, \lambda_c) + (1 - F(T_1 - y, k_c, \lambda_c))\delta(y - T_1 + T_2)] dy \quad (109)$$

$$= [1 - F(T_1 - T_2, k_1, \lambda_1)] [1 - F(T_2, k_c, \lambda_c)] + \int_{T_1-T_2}^{T_1} [1 - F(y, k_1, \lambda_1)] f(T_1 - y, k_c, \lambda_c) dy \quad (110)$$

If Y_c is empty (the two sensor sets do not have any common sensors), but Y_1 and Y_2 are not empty, the marginal CCR can be derived as

$$\text{CCR}_{12} = P \left[\sum_i Y_{1i}^2 > \max \left(\sum_i Y_{2i}^2 + T_1 - T_2, T_1 \right) \right] \quad (111)$$

We can use Lemmas 1, 2, and 7 to write this equation as

$$\text{CCR}_{12} = \int_{T_1}^{\infty} [1 - F(y, k_1, \lambda_1)] [f(y + T_2 - T_1, k_2) + F(y + T_2 - T_1, k_2)\delta(y - T_1)] dy \quad (112)$$

$$= [1 - F(T_1, k_1, \lambda_1)] F(T_2, k_2) + \int_{T_1}^{\infty} [1 - F(y, k_1, \lambda_1)] f(y + T_2 - T_1, k_2) dy \quad (113)$$

In order to limit the integration interval while maintaining accuracy in the computation of CCR_{12} , we find z such that $f(z, k_2) < \epsilon$ for some small user-specified threshold ϵ . This gives

$$\text{CCR}_{12} = [1 - F(T_1, k_1, \lambda_1)] F(T_2, k_2) + \int_{T_1}^{z+T_1-T_2} [1 - F(y, k_1, \lambda_1)] f(y+T_2-T_1, k_2) dy \quad (114)$$

The overall CCR for fault 1 is the probability that SSR_1 is greater than its threshold, and greater than all of the other SSRs relative to their thresholds.

$$\text{CCR}_1 = P[(S_1 > T_1), (S_1 - T_1 > S_2 - T_2), \dots, (S_1 - T_1 > S_n - T_n)] \quad (115)$$

$$\leq \min_i \text{CCR}_{1i} \quad (116)$$

So in order to obtain an upper bound for the CCR of fault 1, we use Equations 102, 106, 110, and 114 as appropriate to find the marginal CCRs. Then we use Equation 116 to find the upper bound for CCR_1 .

3.3 Fault Misclassification Rates

In this section we derive upper bounds for the probability that a fault is incorrectly isolated. If fault 1 occurs, the probability that fault 2 is isolated is called the misclassification rate M_{21} .

First suppose that we have two fault detection algorithms, algorithms 1 and 2. The misclassification rate can then be written as

$$M'_{21} = P[(S_2 - T_2 > S_1 - T_1), (S_2 > T_2)] \quad (117)$$

$$= P \left[\sum_i Y_{2i}^2 > \max \left(\sum_i Y_{1i}^2 + T_2 - T_1, T_2 - \sum_i Y_{ci}^2 \right) \right] \quad (118)$$

where we use the prime symbol on M'_{21} to denote that only two detection algorithms are in use. Suppose that both sensor sets have some unique sensors, and the sensor sets also have some common sensors. That is, neither Y_1 , Y_2 , nor Y_c are empty. We can then use Lemmas 1, 2, 3, and 6 to obtain

$$M'_{21} = \int_{-\infty}^{\infty} [1 - F(y, k_2)] [F(y + T_1 - T_2, k_1, \lambda_1) f(T_2 - y, k_c, \lambda_c) + f(y + T_1 - T_2, k_1, \lambda_1) (1 - F(T_2 - y, k_c, \lambda_c))] dy \quad (119)$$

In order to limit the integration interval while maintaining accuracy in the computation of M'_{21} , we use our knowledge that $f(w, k, \lambda)$ is zero for $w < 0$, and we find values of z_1 and z_c such that $f(z_1, k_1, \lambda_1) < \epsilon$ and $f(z_c, k_c, \lambda_c) < \epsilon$ for some small user-specified threshold ϵ . This gives

$$M'_{21} = \int_L^U [1 - F(y, k_2)] [F(y + T_1 - T_2, k_1, \lambda_1) f(T_2 - y, k_c, \lambda_c) + f(y + T_1 - T_2, k_1, \lambda_1) (1 - F(T_2 - y, k_c, \lambda_c))] dy \quad (120)$$

$$L = \min(T_2 - T_1, T_2 - z_c) \quad (121)$$

$$U = \max(T_2, z_1 + T_2 - T_1) \quad (122)$$

If Y_1 is empty (sensor set 1 does not have any unique sensors), but Y_2 and Y_c are not empty, the misclassification rate can be derived as

$$M'_{21} = P \left[\sum_i Y_{2i}^2 > \max \left(T_2 - T_1, T_2 - \sum_i Y_{ci}^2 \right) \right] \quad (123)$$

We can use Lemmas 1, 3, and 7 to write this as

$$M'_{21} = \int_{T_2-T_1}^{\infty} [1 - F(y, k_2)] [f(T_2 - y, k_c, \lambda_c) + \quad (124)$$

$$(1 - F(T_2 - y, k_c, \lambda_c))\delta(y - T_2 + T_1)] dy$$

$$= [1 - F(T_2 - T_1, k_2)] [1 - F(T_1, k_c, \lambda_c)] + \quad (125)$$

$$\int_{T_2-T_1}^{T_2} [1 - F(y, k_2)] f(T_2 - y, k_c, \lambda_c) dy$$

If Y_2 is empty (sensor set 2 does not have any unique sensors), but Y_1 and Y_c are not empty, the misclassification rate can be derived as

$$M'_{21} = P \left[0 > \max \left(\sum_i Y_{1i}^2 + T_2 - T_1, T_2 - \sum_i Y_{ci}^2 \right) \right] \quad (126)$$

$$= P \left(\sum_i Y_{1i}^2 < T_1 - T_2 \right) P \left(\sum_i Y_{ci}^2 > T_2 \right) \quad (127)$$

$$= F(T_1 - T_2, k_1, \lambda_1) [1 - F(T_2, k_c, \lambda_c)] \quad (128)$$

If Y_c is empty (the two sensor sets do not have any common sensors), but Y_1 and Y_2 are not empty, the misclassification rate can be derived as

$$M'_{21} = P \left[\sum_i Y_{2i}^2 > \max \left(\sum_i Y_{1i}^2 + T_2 - T_1, T_2 \right) \right] \quad (129)$$

We can use Lemmas 1, 2, and 7 to write this equation as

$$M'_{21} = \int_{T_2}^{\infty} [1 - F(y, k_2)] [f(y + T_1 - T_2, k_1, \lambda_1) + \quad (130)$$

$$F(y + T_1 - T_2, k_1, \lambda_1)\delta(y - T_2)] dy$$

$$= [1 - F(T_2, k_2)] F(T_1, k_1, \lambda_1) + \int_{T_2}^{\infty} [1 - F(y, k_2)] f(y + T_1 - T_2, k_1, \lambda_1) dy \quad (131)$$

In order to limit the integration interval while maintaining accuracy in the computation of M'_{21} , we find z such that $f(z, k_1, \lambda_1) < \epsilon$ for some small user-specified threshold ϵ . This gives

$$M'_{21} = [1 - F(T_2, k_2)] F(T_1, k_1, \lambda_1) + \int_{T_2}^{z+T_2-T_1} [1 - F(y, k_2)] f(y + T_1 - T_2, k_1, \lambda_1) dy \quad (132)$$

Given that we have $n > 2$ detection algorithms running in parallel, the misclassification rate M_{21} is bounded from above by M'_{21} . So in order to obtain an upper bound for M_{21} , we use Equation 121, 125, 128, or 132 as appropriate. This can be repeated for all marginal

misclassification rates M_{ij} . This gives

$$M_{ij} = P[(S_2 - T_2 > S_1 - T_1), (S_2 - T_2 > S_3 - T_3), \dots, (S_2 - T_2 > S_n - T_n), (S_2 > T_2)] \quad (133)$$

$$\leq M'_{ij}, \quad (i = 1, \dots, n), (j = 1, \dots, n), (i \neq j) \quad (134)$$

Finally we consider the probability M_{0i} that no fault is indicated when fault i occurs. This misclassification rate can be bounded from above as

$$M_{0i} = P(S_1 < T_1, \dots, S_n < T_n) \quad (135)$$

$$\leq P(S_i < T_i) \quad (136)$$

$$M_{0i} \leq F(T_i, k_{ia}, \lambda_{ia}) \quad (137)$$

3.4 Summary of Confusion Matrix Bounds

Recall the confusion matrix shown in Table 1. The rows correspond to fault conditions, and the columns correspond to fault isolation results. The element in the i th row and j th column is the probability that fault j is isolated when fault i occurs. The previous sections derived the following bounds.

- CCR_i for $i \in [1, n]$ is the probability that fault i is correctly isolated given that it occurs, and its lower and upper bounds are given in Equations 97 and 116.
- CNR is the probability that a no-fault condition is correctly indicated given that no fault occurs, and its lower and upper bounds are given in Equation 62.
- M_{ij} for $i, j \in [1, n]$ and $i \neq j$ is the probability that fault i is incorrectly isolated given that fault j occurs, and its upper bound is given in Equation 134.
- M_{i0} for $i \in [1, n]$ is the probability that fault i is incorrectly isolated given that no fault occurs, and its upper bound is given in Equation 82.
- M_{0i} for $i \in [1, n]$ is the probability that no fault is isolated given that fault i occurs, and its upper bound is given in Equation 137.

3.5 Computational Effort

Usually confusion matrices are obtained with simulations. In order to derive an experimental confusion matrix with N faults, the number of matrix elements that needs to be calculated is on the order of N^2 . (For example, if the number of faults doubles, then the computational effort quadruples.) Also, the required number of simulations for each matrix element calculation is on the order of N . This is because as the number of possible faults increases, the number of simulations required to obtain the same statistical accuracy increases in direct proportion. Therefore the computational effort required for the experimental determination of a confusion matrix is on the order of N^3 .

The bounds derived in this paper also require computational effort that is on the order of N^3 . This is because each of the bounds summarized in Section 3.4 required computational effort on the order of N , and the number of matrix elements is on the order of N^2 .

4 Simulation Results

In this section we use simulation results to verify the theoretical bounds of the preceding sections. We consider the problem of isolating an aircraft turbofan engine fault which is modeled by the NASA Commercial Modular Aero-Propulsion System Simulation (C-MAPSS) [25]. There are five possible faults that can occur: fan, low pressure compressor (LPC), high pressure compressor (HPC), high pressure turbine (HPT), and low pressure turbine (LPT). These five faults entail shifts of both efficiency and flow capacity from nominal values. The fault magnitudes that we try to detect are 2.5% for the fan, 20% for the LPC, 2% for the HPC, 1.5% for the HPT, and 2% for the LPT. These magnitudes were chosen to give reasonable fault detection ability. The LPC fault has a very small fault signature, so it is not detectable unless its magnitude is relatively large.

The available sensors and their standard deviations are shown in Table 2. The fault influence coefficient matrix shown in Table 3 was generated using C-MAPSS and is based on [26]. The numbers in Table 3 are the partial derivatives of the sensor outputs with respect to the fault conditions, normalized to the fault percentages discussed above, and normalized to one standard deviation of the sensor noise.

Table 2: Aircraft engine sensors and standard deviations. The temperature sensor standard deviations are given in units of degrees Rankine. All other sensor standard deviations are given as a percentage relative to their nominal values.

Symbol	Description	Standard deviation
Nc	Core speed	0.25%
P15	Bypass duct pressure	0.5%
P24	LPC outlet pressure	0.5%
Ps30	HPC outlet pressure	0.5%
T24	LPC outlet temperature	1 deg
T30	HPC outlet temperature	2 deg
T48	HPT outlet temperature	8 deg
Wf	Fuel flow	0.75%

In order to decide which sensors to use for detecting each fault, we included one sensor at a time in a fault detection algorithm, starting with the sensor with the largest fault signature. This gave a total of eight possible sensor sets for each fault detection algorithm. We specified a maximum allowable false positive rate of 0.0001, and calculated the smallest threshold for each possible sensor set that would give an FPR no greater than 0.0001 (see Equation 26 and Figure 1). Given the threshold, we next calculated the true positive rate (see Equation 84 and Figure 2). We then chose the sensor set that gave the largest TPR.

Table 3: Fault signatures. The rows show the five different fault conditions. The columns show the mean sensor value residuals for the given fault conditions, normalized to one standard deviation.

		Sensors							
		Nc	P15	P24	Ps30	T24	T30	T48	Wf
Faults	Fan	0.80	2.10	1.80	4.05	0.43	0.49	1.21	3.40
	LPC	0.00	0.00	4.80	1.20	3.20	0.20	0.80	0.27
	HPC	0.72	0.08	0.32	0.64	0.54	5.23	3.08	1.60
	HPT	0.96	0.12	0.39	3.27	0.72	2.63	4.40	2.18
	LPT	1.20	0.12	0.60	3.56	0.90	3.34	0.03	2.32

As an example, consider the fan fault with the normalized fault signatures shown in Table 3. The sensors with the largest fault signatures in descending order are Ps30, Wf, T30, P15, P24, T48, Nc, and T24. This gives eight potential sensor sets for detecting a fan fault: the first potential set uses only sensor Ps30, the second potential set uses Ps30 and Wf, and so on. The potential sensor sets along with their detection thresholds and TPRs are shown in Table 4. Table 4 shows that using five sensors gives the largest TPR given the constraint that $FPR \leq 0.0001$.

Table 4: Potential sensor sets for detecting a fan fault. The thresholds were determined by constraining $FPR \leq 0.0001$. This table shows that using five sensors gives the largest TPR subject to the FPR constraint.

Sensors	Detection threshold T_1	True positive rate TPR_1
Ps30	15.1	0.563
Ps30, Wf	18.4	0.865
Ps30, Wf, T30	21.1	0.926
Ps30, Wf, T30, P15	23.5	0.949
Ps30, Wf, T30, P15, P24	25.7	0.959
Ps30, Wf, T30, P15, P24, T48	27.9	0.958
Ps30, Wf, T30, P15, P24, T48, Nc	29.9	0.952
Ps30, Wf, T30, P15, P24, T48, Nc, T24	31.8	0.942

This process described in the previous paragraph was repeated for each fault shown in Table 3. The resulting sensor sets are shown in Table 5 and were therefore chosen for FDI for each fault type. Note that given an FPR constraint, the detection threshold is a function only of the number of sensors in each sensor set; the detection threshold is independent of the specific fault signatures. This is illustrated in Figure 1 where it is seen that $f(x, k)$ is a function only of x and k (the number of sensors).

Finally we used the fault isolation method shown in Equation 64, and we used the equations developed in the previous sections to determine theoretical lower and upper bounds for the confusion matrix as summarized in Section 3.4. We then ran 100,000 fault simulations in order to obtain an experimental confusion matrix. Table 6 shows the theoretical lower

bounds of the diagonal elements of the confusion matrix. Lower bounds of the off-diagonal elements were not obtained because we are typically more interested in upper bounds of off-diagonal elements. Table 7 shows the theoretical upper bounds of the confusion matrix. Table 8 shows the experimental confusion matrix. These tables show that the theoretical results derived in this paper give reasonably tight bounds to the experimental confusion matrix values.

Recall that we used an FPR of 0.0001 to choose our sensor sets and detection thresholds. This means that the first five elements in the last row of Table 7 are guaranteed to be no greater than 0.0001. It further means that the element in the lower right corner of Table 6 is guaranteed to be no greater than $1 - 5(0.0001) = 0.9995$.

Table 5: Sensor sets for fault detection. These sensor sets give the largest TPR for each fault given the constraint that $FPR \leq 0.0001$.

Fault i	Sensor set i	Detection threshold T_i	True positive rate TPR_i
Fan	Ps30, Wf, T30, P15, P24	25.7	0.959
LPC	P24, T24	18.4	0.943
HPC	T30, T48	18.4	0.970
HPT	T48, Ps30, T30, Wf	23.5	0.970
LPT	Ps30, T30, Wf	21.1	0.844

Table 6: Lower bounds of diagonal confusion matrix elements. Lower bounds for the off-diagonal elements have not been obtained. The rows specify the actual fault condition, and the columns specify the diagnosis.

	Fan	LPC	HPC	HPT	LPT	No Fault
Fan	0.6691					
LPC		0.9342				
HPC			0.8692			
HPT				0.9345		
LPT					0.6623	
No Fault						0.9992

Table 7: Upper bounds of confusion matrix elements. The rows specify the actual fault condition, and the columns specify the diagnosis.

	Fan	LPC	HPC	HPT	LPT	No Fault
Fan	0.7761	0.0000	0.0001	0.1115	0.1899	0.0408
LPC	0.0076	0.9356	0.0000	0.0000	0.0000	0.0573
HPC	0.0097	0.0000	0.8936	0.0769	0.0160	0.0303
HPT	0.0015	0.0000	0.0270	0.9445	0.0014	0.0300
LPT	0.0874	0.0000	0.0030	0.1066	0.7422	0.1557
No Fault	0.0000	0.0000	0.0000	0.0000	0.0001	0.9999

Table 8: Experimental confusion matrix. The rows specify the actual fault condition, and the columns specify the diagnosis. The numbers are based on 100,000 simulations of each fault. The numbers in each row should add up to 1 (within rounding error).

	Fan	LPC	HPC	HPT	LPT	No Fault
Fan	0.7614	0.0000	0.0000	0.0370	0.1677	0.0338
LPC	0.0073	0.9354	0.0000	0.0000	0.0000	0.0573
HPC	0.0051	0.0000	0.8875	0.0713	0.0074	0.0288
HPT	0.0016	0.0000	0.0276	0.9422	0.0011	0.0275
LPT	0.0831	0.0000	0.0015	0.0993	0.6680	0.1481
No Fault	0.0000	0.0000	0.0000	0.0000	0.0001	0.9997

Note that it is possible for an element in the experimental confusion matrix in Table 8 to lie outside the bounds shown in Tables 6 and 7. For example, compare the numbers in the fourth row and first column in Tables 7 and 8. This is because the numbers in Table 8 are experimentally obtained on the basis of a finite number of simulations, and are guaranteed to lie within their theoretical bounds only as the number of simulations approaches infinity. In fact, that is one of the strengths of the analytic method proposed in this paper. The analytic bounds are definite, but simulations are subject to random effects. Also, simulations can give misleading conclusions if the simulation has errors. One common simulation error is the non-randomness of commonly used random number generators [27].

In summary, the user specifies the maximum FPR for each fault, and then finds the sensor set that has the largest TPR given the FPR constraint. Analytic confusion matrix bounds are then obtained using the theory in this paper. If the results are not satisfactory, the user can iterate by changing the maximum FPR constraint. For example, if a TPR is too small, then the user will have to increase the FPR constraint. If the confusion matrix bounds of fault isolation probabilities are not satisfactory, the user will have to iterate on the FPR constraints in order to obtain different confusion matrix bounds.

5 Conclusion

This paper has introduced a new fault detection and isolation (FDI) algorithm and derived analytical confusion matrix bounds. The FDI algorithm is fairly simple and has not been compared with other algorithms. The main contribution of this paper is the generation of analytic confusion matrix bounds, and the possibility that our methodology could be adapted to other FDI algorithms. Usually confusion matrices are obtained with simulations. Such simulations have several potential drawbacks. First, they can be time consuming. Second, they can give misleading conclusions if not enough simulations are run to give statistically significant results. Third, they can give misleading conclusions if the simulation has errors (for example, if the output of the random number generator does not satisfy statistical tests for randomness). The theoretical confusion matrix bounds derived in this paper do not depend on a random number generator and can be used in place of simulations.

Further work in this area could follow several directions. First, the tightness of the confusion matrix bounds could be quantified. This paper derives bounds but does not guarantee how loose or tight those bounds are. Second, the bounds could be modified to be tighter. Third, bounds could be attempted for methods other than the FDI algorithm proposed here. The fault isolation method we used isolates the fault that has the largest sum of squares of sensor residuals (SSR) relative to its detection threshold. Other fault isolation methods could normalize the relative SSR to its standard deviation, or could normalize the absolute SSR to its detection threshold. Preliminary efforts have not been successful in deriving bounds for these fault isolation methods, but additional effort might bring success. All of these FDI methods are static, which means that faults are isolated using measurements at a single time. Better fault isolation might be achieved if dynamic system information is used. The derivation of theoretical confusion matrix bounds in this case would require additional work.

References

- [1] S. Chinchalkar, Determination of crack location in beams using natural frequencies, *Journal of Sound and Vibration*, vol. 247, pp. 417-429, 2001.
- [2] T. Tsai and Y. Wang, Vibration analysis and diagnosis of a cracked shaft, *Journal of Sound and Vibration*, vol. 192, pp. 607-620, 1996.
- [3] H. Wang, Z. Huang, and D. Steven, On the use of adaptive updating rules for actuator and sensor fault diagnosis, *Automatica*, vol. 33, pp. 217-224, 1997.
- [4] J. Chen, R. Patton, and H. Zhang, Design of unknown input observers and robust fault detection filters, *International Journal of Control*, vol. 63, no. 1, pp. 85-105, 1996.
- [5] J. Korbicz, J. Koscielny, Z. Kowalczyk, and W. Cholewa, *Fault Diagnosis: Models, Artificial Intelligence, Applications*, Springer, 2004.
- [6] P. Li and V. Kadiramanathan, Fault detection and isolation in non-linear stochastic systems – A combined adaptive Monte Carlo filtering and likelihood ratio approach, *International Journal of Control*, vol. 77, pp. 1101-1114, 2004.
- [7] C. Tan and C. Edwards, Sliding mode observers for robust detection and reconstruction of actuator sensor faults, *International Journal of Robust and Nonlinear Control*, vol. 13, no. 5, pp. 443-463, 2003.
- [8] I. Yaesh and U. Shaked, Robust H_∞ deconvolution and its application to fault detection, *Journal of Guidance, Control and Dynamics*, vol. 23, no. 6, pp. 1101-1112, 2000.
- [9] C. Ocampo-Martinez, S. Tornil, and V. Puig, Robust fault detection using interval constraints satisfaction and set computations, *IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes*, August 30 – September 1, 2006, Beijing, pp. 1285-1290.

- [10] W. Fenton, T. MicGinnity, and L. Maguire, Fault diagnosis of electronic systems using intelligent techniques: A review, *IEEE Transactions on Systems, Man and Cybernetics: Part C – Applications and Reviews*, vol. 31, no. 3, pp. 269-281, 2001.
- [11] H. Schneider and P. Frank, Observer-based supervision and fault detection in robots using nonlinear and fuzzy-logic residual evaluation, *IEEE Transactions on Control System Technology*, vol. 4, pp. 274-282, 1996.
- [12] M. Napolitano, C. Neppach, V. Casdorff, S. Naylor, M. Innocenti, and G. Silvestri, Neural-network-based scheme for sensor failure detection, identification and accommodation, *Journal of Guidance, Control and Dynamics*, vol. 18, pp. 1280-1286, 1995.
- [13] Z. Yangping, Z. Bingquan, and W. DongXin, Application of genetic algorithms to fault diagnosis in nuclear power plants, *Reliability Engineering and System Safety*, vol. 67, no. 2, pp. 153-160, February 2000.
- [14] W. Gui, C. Yang, J. Teng, and W. Yu, Intelligent fault diagnosis in lead-zinc smelting process, *IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes*, August 30 – September 1, 2006, Beijing, pp. 234-239.
- [15] S. Lu and B. Huang, Condition monitoring of model predictive control systems using Markov models, *IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes*, August 30 – September 1, 2006, Beijing, pp. 264-269.
- [16] R. Isermann, Supervision, fault-detection and fault-diagnosis methods – An introduction, *Control Engineering Practice*, vol. 5, no. 5, pp. 639-652, 1997.
- [17] X. Deng and X. Tian, Multivariate statistical process monitoring using multi-scale kernel principal component analysis, *IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes*, August 30 – September 1, 2006, Beijing, pp. 108-113.
- [18] A. Pernestål, M. Nyberg, and B. Wahlberg, A Bayesian approach to fault isolation – Structure estimation and inference, *IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes*, August 30 – September 1, 2006, Beijing, pp. 450-455.
- [19] S. Campbell and R. Nikoukhah, *Auxiliary Signal Design for Failure Detection*, Princeton University Press, 2004.
- [20] F. Gustafsson, Statistical signal processing approaches to fault detection, *IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes*, August 30 – September 1, 2006, Beijing, pp. 24-35.
- [21] J. Gertler, *Fault Detection and Diagnosis in Engineering Systems*, CRC, 1998.
- [22] M. Abramowitz and I. Stegun, *Handbook of Mathematical Functions*, Dover Publications, 1965.

- [23] A. Papoulis and S. Pillai, Probability, Random Variables, and Stochastic Processes, McGraw-Hill, 2002.
- [24] P. Peebles, Probability, Random Variables, and Random Signal Principles, McGraw-Hill, 2001.
- [25] D. Frederick, J. DeCastro, and J. Litt, User's Guide for the Commercial Modular Aero-Propulsion System Simulation (C-MAPSS), NASA Technical Memorandum TM-2007-215026.
- [26] D. L. Simon, J. Bird, C. Davison, A. Volponi, and R. Iverson, Benchmarking gas path diagnostic methods: A public approach, ASME Turbo Expo, Paper GT2008-51360, June 2008.
- [27] P. Savicky and M. Robnik-Sikonja, Learning random numbers: A Matlab anomaly, Applied Artificial Intelligence, vol. 22, no. 3, March 2008, pp. 254-265.

REPORT DOCUMENTATION PAGE			Form Approved OMB No. 0704-0188		
<p>The public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Department of Defense, Washington Headquarters Services, Directorate for Information Operations and Reports (0704-0188), 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to any penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.</p> <p>PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.</p>					
1. REPORT DATE (DD-MM-YYYY) 01-07-2009		2. REPORT TYPE Technical Memorandum		3. DATES COVERED (From - To)	
4. TITLE AND SUBTITLE Analytic Confusion Matrix Bounds for Fault Detection and Isolation Using a Sum-of-Squared-Residuals Approach			5a. CONTRACT NUMBER		
			5b. GRANT NUMBER		
			5c. PROGRAM ELEMENT NUMBER		
6. AUTHOR(S) Simon, Dan; Simon, Donald, L.			5d. PROJECT NUMBER		
			5e. TASK NUMBER		
			5f. WORK UNIT NUMBER WBS 645846.02.07.03.12		
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) National Aeronautics and Space Administration John H. Glenn Research Center at Lewis Field Cleveland, Ohio 44135-3191			8. PERFORMING ORGANIZATION REPORT NUMBER E-16998		
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) National Aeronautics and Space Administration Washington, DC 20546-0001			10. SPONSORING/MONITOR'S ACRONYM(S) NASA		
			11. SPONSORING/MONITORING REPORT NUMBER NASA/TM-2009-215655		
12. DISTRIBUTION/AVAILABILITY STATEMENT Unclassified-Unlimited Subject Category: 07 Available electronically at http://gltrs.grc.nasa.gov This publication is available from the NASA Center for AeroSpace Information, 443-757-5802					
13. SUPPLEMENTARY NOTES					
14. ABSTRACT Given a system which can fail in 1 or n different ways, a fault detection and isolation (FDI) algorithm uses sensor data in order to determine which fault is the most likely to have occurred. The effectiveness of an FDI algorithm can be quantified by a confusion matrix, which indicates the probability that each fault is isolated given that each fault has occurred. Confusion matrices are often generated with simulation data, particularly for complex systems. In this paper we perform FDI using sums of squares of sensor residuals (SSRs). We assume that the sensor residuals are Gaussian, which gives the SSRs a chi-squared distribution. We then generate analytic lower and upper bounds on the confusion matrix elements. This allows for the generation of optimal sensor sets without numerical simulations. The confusion matrix bounds are verified with simulated aircraft engine data.					
15. SUBJECT TERMS Gas turbine engines; Systems health monitoring; Fault detection; Statistical analysis					
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES 32	19a. NAME OF RESPONSIBLE PERSON
a. REPORT U	b. ABSTRACT U	c. THIS PAGE U			STI Help Desk (email: help@sti.nasa.gov)
					19b. TELEPHONE NUMBER (include area code) 443-757-5802

