Analytical Redundancy and the Design of Robust Failure Detection Systems

EDWARD Y. CHOW, MEMBER, IEEE, AND ALAN S. WILLSKY, SENIOR MEMBER, IEEE

Abstract—**The failure detection and identification (FDI) process is viewed as consisting of two stages: residual generation and decision making. It is argued that a robust FDI system can be achieved by designing a robust residual generation process. Analytical redundancy, the basis for residual generation, is characterized in terms of a parity space. Using the concept of parity relations, residuals can be generated in a number of ways and the design of a robust residual generation process can be formulated as a minimax optimization problem. An example is included to illustrate this design methodology.**

I. INTRODUCTION

PHYSICAL systems are often subjected to unexpected changes, such as component failures and variations in operating condition, that tend to degrade overall system performance. We will refer to such changes as "failures," although they may not represent the failing of physical components. In order to maintain a high level of performance, it is important that failures be promptly detected and identified so that appropriate remedies can be applied. Over the past decade numerous approaches to the problem of failure detection and identification (FDI) in dynamical systems have been developed [1]; detection filters [2], [3], the generalized likelihood ratio (GLR) method [4], [5], and the multiple model method [5], [6] are some examples. All of these analytical methods require that a dynamic model of some sort be given. The goal of this paper is to investigate the issue of designing FDI systems which are robust to uncertainties in the models on which they are based.

The FDI process essentially consists of two stages: residual generation and decision making. For a particular set of hypothesized failures, an FDI system has the structure shown in Fig. 1. Outputs from sensors are initially processed to enhance the effect of a failure (if present) so that it can be recognized. The processed measurements are called the *residuals*, and the enhanced failure effect on the residuals is called the *signature* of the failure. Intuitively, the residuals represent the difference between various functions of the observed sensor outputs and the expected values of these functions in the normal (no-fail) mode. In the absence of a failure residuals should be unbiased, showing agreement between observed and expected normal behavior of the system; a failure signature typically takes the form of residual biases that are characteristic of the failure. Thus, residual generation is based on knowledge of the normal behavior of the system. The actual process of residual generation can vary in complexity. For example, in voting systems [7], [8] the residuals are simply the differences of the outputs of the various like sensors, whereas in a GLR system, the residuals are the innovations generated by the Kalman filter.

In the second stage of an FDI algorithm, the decision process, the residuals are examined for the presence of failure signatures. Decision functions or statistics are calculated using the residuals, and a decision rule is then applied to the decision statistics to determine if any failure has occurred. A decision process may consist of a simple threshold test on the instantaneous values or moving averages of the residuals, or it may be based directly on methods of statistical decision theory, e.g., the sequential probability ratio test [9].

The first concern in the design of an FDI system is detection performance, i.e., the ability to detect and identify failures promptly and correctly with minimal delays and false alarms. In the literature, this issue has typically been dealt with using a given model of the normal system behavior. An equally important design issue that is necessarily examined in practice but has received little theoretical attention is robustness: minimizing the sensitivity of detection performance to model errors and uncertainties. An ideal simplistic approach to designing a robust FDI system is to include all uncertainties in the overall problem specification; then a robust design is obtained by optimizing (in some sense) the performance of the entire system with the uncertainties present. However, this generally leads to a complex mathematical problem that is too difficult to solve in practice. On the other hand, a simple approach is to ignore all model uncertainties in the performance optimization process. The resulting design is then evaluated in the presence of modeling errors. If the degradation in performance is tolerable, the design is accepted. Otherwise, it is modified and reevaluated. Although this method often yields acceptable designs, it has several drawbacks. For example, it may be unclear what parts of the design should be modified and what form the modification should take. Furthermore, each iteration may be very expensive to carry out since extensive Monte Carlo simulations are often required for performance evaluations.

In this paper we develop a systematic approach that considers uncertainties directly. Our work is motivated by the practical design effort of Deckert *et al.* for an aircraft sensor FDI system [10]. The basic idea used in this work was to identify the analytical redundancy relations of the system that were known well and those that contained substantial uncertainties. An FDI system (i.e., its residual generation process) was then designed based primarily on the well-known relationships (and only secondarily on the less well-known relations) of the system behavior. As model error directly affects residual generation, this approach suggests that robustness can be effectively achieved by designing a robust residual generation process. In our work, we have extracted and extended the practical idea underlying this application and developed a general approach to the design of robust FDI algorithm. In addition to its use in specifying residual generation procedure, our approach is also useful as it can provide a quantitative measure of the attainable level of robust-

Manuscript received October 25, 1982; revised May 12, 1983. Paper recommended by T. Basar, Past Chairman of the Large Scale Systems, Differential Games Committee. This work was supported in part by the Office of Naval Research under Contract N00014-77-C-0224 and in part by the NASA Ames Research Center under Grant NGL-22-009-124.

E. Y. Chow is with Schlumberger-Doll Research, Ridgefield, CT 06877.

A. S. Willsky is with the Laboratory for Information and Decision Systems, Massachusetts Institute of Technology, Cambridge, MA 02139.

0018-9286/84/0700-0603$01.00

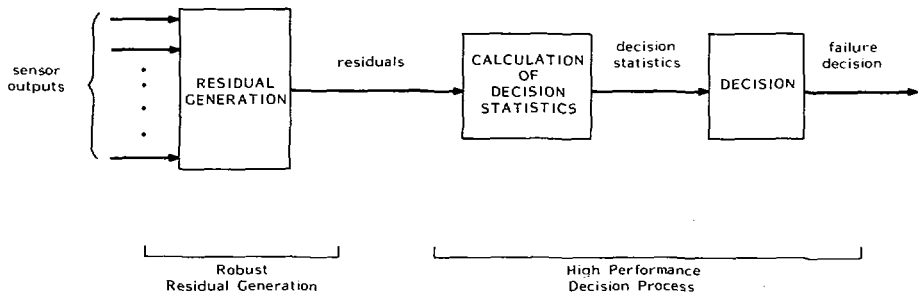

Fig. 1. Two-stage structure of the FDI process.

ness in the early states of a design. This can allow the designer to assess what he can expect in terms of overall performance.

In order to develop residual generation procedures, it is important to identify the redundancy relations of a system and to characterize them according to how they are affected by model errors and uncertainties. In this paper, we further develop the concept of analytical redundancy that is used in [10], [11], and we use this as a basis for determining redundancy relations to be used for residual generation which are least sensitive to model errors.

In Section II we describe the concept of analytical redundancy and present a mathematical characterization of redundancy in linear dynamical systems that extends ideas developed previously. We also provide for the first time a clear, general interpretation of a redundancy relation as a reduced-order auto-regressive-moving-average (ARMA) model and use this in Section III to describe the various ways that analytical redundancy can be used for residual generation and FDI. In Section IV a method of determining redundancy relations that are least sensitive to model error and noise effects is described. A numerical example illustrating some of the developed concepts is presented in Section V. Conclusions and discussions are included in Section VI.

II. Analytical Redundancy—Parity Relations

The basis for residual generation is analytical redundancy, which essentially takes two forms: 1) direct redundancy—the relationship among instantaneous outputs of sensors; and 2) temporal redundancy—the relationship among the histories of sensor outputs and actuator inputs. It is based on these relationships that outputs of (dissimilar) sensors (at different times) can be compared. The residuals resulting from these comparisons are then measures of the discrepancy between the behavior of observed sensor outputs and the behavior that should result under normal conditions. Examples where direct redundancy was exploited include [7], [8], [11], [12], [13]; explicit use of temporal redundancy was made in [10], [20].

In order to develop a clear picture of redundancy, consider the following deterministic model:

$$x(k+1) = Ax(k) + \sum_{j=1}^{q} b_j u_j(k) \tag{1a}$$

$$y_j(k) = c_j x(k), \qquad j = 1, \cdots, M \tag{1b}$$

where x is the N-dimensional state vector, A is a constant $N \times N$ matrix, b_j is a constant column N-vector, and c_j is a constant row N-vector. The scalar u_j is the known input to the jth actuator, and the scalar y_j is the output of the jth sensor.

Direct redundancy exists among sensors whose outputs are algebraically related, i.e., the sensor outputs are related in such a way that the variable one sensor measures can be determined by the instantaneous outputs of the other sensors. For the system (1), this corresponds to the situation where a number of the c_j's are linearly dependent. In this case, the value of one of the observations can be written as a linear combination of the other outputs. For example, we might have

$$y_1(k) = \sum_{i=2}^{M} \alpha_i y_i(k) \tag{2}$$

where the α_i's are constants. This indicates that under normal conditions the ideal output of sensor 1 can be calculated from those of the remaining sensors. In the absence of a failure in the sensors, the residual $y_1(k) - \sum_{i=2}^{M} \alpha_i y_i(k)$ should be zero. A deviation from this behavior provides the indication that one of the sensors has failed. This is the underlying principle used in strapdown inertial reference unit (SIRU) FDI [7], [8]. Note that while direct redundancy is useful for sensor failure detection, it is not useful for detecting actuator failures (as modeled by a change in the b_j, for instance).

Because temporal redundancy relates sensor output and actuator inputs, it can potentially be used for both sensor and actuator FDI. For example, consider the relationship between velocity (v) and acceleration (a):

$$v(k+1) = v(k) + Ta(k) \tag{3}$$

where T is the sampling interval. Equation (3) prescribes a way of comparing velocity measurements and accelerometer outputs (by checking to see if the residual $v(k+1) - v(k) - Ta(k)$ is zero) that may be used in a mixed velocity–acceleration sensor voting system for the detection of both types of sensor failures. Temporal redundancy facilitates the comparison of sensors among which direct redundancy does not exist. Hence, it can lead to a reduction of hardware redundancy for sensor FDI. Viewed in a different light, the use of analytical redundancy implies that additional sensor failures can in principle be detected with the same level of hardware redundancy.

To see how temporal redundancy can be exploited for detecting actuator failures, let us consider a simplified first-order model of a vehicle in motion:

$$v(k+1) = \alpha v(k) + Tu(k) \tag{4}$$

where v denotes the vehicle's velocity, α is a scalar constant between zero and one reflecting the effect of friction and drag, T is the sampling interval, and u is the commanded engine force (actuator input) divided by the vehicle's mass. Now the velocity measurements can be compared to the actuator inputs by means of (4), i.e., through examining the residual $v(k+1) - \alpha v(k) - Tu(k)$. An actuator failure can be inferred, if the sensor is functioning normally, but the residual is nonzero.

While the additional information supplied by dissimilar sensors and actuators at different times through temporal redundancy facilitates the detection of a greater variety of failures and reduces hardware redundancy, exploitation of this additional information often results in increased computational complexity, since the dynamics of the system are used in the residual generation

process. However, the major issue in the use of analytical redundancy is the inevitable uncertainty in our knowledge of the system dynamics [e.g., of α in (4)] and the consequences of this uncertainty on the robustness of the resulting FDI algorithm. From the above discussion one approach to the design of robust residual generation in any given application is evident: first, the various redundancies that are relevant to the failures under consideration are to be determined; then residual generation is based on those relations that are least sensitive to parameter uncertainties. This is the approach we have adopted. In the remainder of this section we will present a characterization of analytical redundancy and in a subsequent section we will quantify the effect of uncertainties on a redundancy relation.

The Generalized Parity Space

Let us define

$$\boldsymbol{C}_j(k)=\begin{bmatrix} c_j \\ c_jA \\ \vdots \\ c_jA^k \end{bmatrix} \qquad k=0,1,\cdots,; \quad j=1,\cdots,M. \tag{5}$$

The well-known Cayley–Hamilton theorem [14] implies that there is an n_j, $1\leqslant n_j\leqslant N$, such that

$$\operatorname{rank}\boldsymbol{C}_j(k)=\begin{cases} k+1 & k<n_j \\ n_j & k\geqslant n_j. \end{cases} \tag{6}$$

The null space of the matrix $\boldsymbol{C}_j(n_j-1)$ is known as the unobservable subspace of the jth sensor. The rows of $\boldsymbol{C}_j(n_j-1)$ span a subspace of R^N that is the orthogonal complement of the unobservable subspace. Such a subspace will be referred to as the *observable subspace* of the jth sensor, and it has dimension n_j.

Let ω be a row vector of dimension $n=\sum_{i=1}^{M}(n_i+1)$ such that $\omega=[\omega^1,\cdots,\omega^M]$, where ω^j, $j=1,\cdots,M$, is a (n_j+1)-dimensional row vector. Consider a nonzero ω satisfying

$$[\omega^1,\cdots,\omega^M]\begin{bmatrix} \boldsymbol{C}_1(n_1) \\ \vdots \\ \boldsymbol{C}_M(n_M) \end{bmatrix}x(k)=0, \qquad x(k)\in R^N. \tag{7}$$

Note that in the above equation $\boldsymbol{C}_j(n_j)$ has n_j+1 rows while it is only of rank n_j. The reason for this will become clear when we discuss the temporal redundancy for a single sensor. Assuming that the system (1) is observable, there are only $n-N$ linearly independent ω's satisfying (7). We let Ω be an $(n-N)\times N$ matrix with a set of such independent ω's as its rows. (The matrix Ω is not unique.) Assuming all the inputs are zero for the moment, we have

$$P(k)=\Omega\begin{bmatrix} Y_1(k,n_1) \\ \vdots \\ Y_M(k,n_M) \end{bmatrix} \tag{8}$$

where

$$Y_j(k,n_j)=\begin{bmatrix} y_j(k) \\ \vdots \\ y_j(k+n_j) \end{bmatrix}, \qquad j=1,\cdots,M.$$

Note that (8) is independent of the state $x(k)$. The $(n-N)$-vector $P(k)$ is called the *parity vector*. In the absence of noise and failures, $P(k)=0$. In the noisy no-fail case, $P(k)$ is a zero-mean random vector. Under noise and failures, $P(k)$ will become biased. Moreover, different failures will produce different (biases in the) $P(k)$'s. Thus, the parity vector may be used as the signature-carrying residual for FDI. We will further discuss residual generation based on parity equations in Section III.

The matrix Ω may be generated by making direct use of (7). Let

$$T=\begin{bmatrix} \boldsymbol{C}_1(n_1) \\ \vdots \\ \boldsymbol{C}_M(n_M) \end{bmatrix}.$$

From (7), we see that the rows of Ω span the orthogonal complement of the range space of T. This suggests that Ω can be generated by subtracting the orthogonal projection onto T from the identity operator. That is, Ω can be chosen to consist of the $(n-N)$ independent rows of $I-T(T'T)^{-1}T'$. Alternatively, by viewing parity checks as finite impulse response (FIR) filters, [17] gives a method for constructing Ω in terms of null spaces of polynomial matrices.

When the actuator inputs are not zero, (8) must be modified to take into account this effect. In this case

$$P(k)=\Omega\left\{\begin{bmatrix} Y_1(k,n_1) \\ \vdots \\ Y_M(k,n_M) \end{bmatrix}-\begin{bmatrix} \boldsymbol{B}_1(n_1) \\ \vdots \\ \boldsymbol{B}_M(n_M) \end{bmatrix}U(k,n_0)\right\} \tag{9}$$

where

$$\boldsymbol{B}_j(n_j)=\begin{bmatrix} 0 & 0 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & 0 \\ c_jB & 0 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & c_jB & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ c_jA^{n_j-1}B & c_jA^{n_j-2}B & \cdot & \cdot & c_jB & 0 & \cdot & \cdot & 0 \end{bmatrix}$$

$$B=\begin{bmatrix} b_1,\cdots,b_q \end{bmatrix}$$

$$u(k)=\begin{bmatrix} u_1(k),\cdots,u_q(k) \end{bmatrix}'$$

$$n_0=\max(n_1,\cdots,n_M)$$

$$U(k,n_0)=\begin{bmatrix} u'(k),\cdots,u'(k+n_0) \end{bmatrix}'$$

$\boldsymbol{B}_j(n_j)$ is an $(n_j+1)\times n_0q$ matrix (q is the number of actuators). Note that (9) only involves the measurable inputs and outputs of the system, and it does not depend on the state $x(k)$ which is not directly measured.

The quantity $P(k)$ is known as the *generalized parity vector*, which is nonzero (or nonzero mean if noise is present) only if a failure is present. The $(n-N)$ dimensional space of all such vectors is called the *generalized parity space*. Under the no-fail situation ($P(k)=0$), (9) characterizes all the analytical redundancies for the system (1) because it specifies all the possible relationships among the actuator inputs and sensor outputs. Any linear combination of the rows of (9) is called a *parity equation* or a *parity relation*; any linear combination of the right-hand side (RHS) of (9) is called a *parity function*. Equation (6) implies that we do not need to consider a higher dimensional parity space than is defined by (9) with n_j replaced by $l_j>n_j$, $j=1,\cdots,M$, although it is possible to do so. We note that the generalized parity space we have just defined here is an extension of the parity space considered by Potter and Suman [11] to include sensor outputs and actuator inputs at different times. In [11], Potter and Suman studied exclusively (9) with $n_1=n_2=\cdots=0$.

A useful notion in describing analytical redundancy is the *order* of a redundancy relation. Consider a parity relation (under the no-fail condition) defined by

$$\sum_{j=1}^{M} \omega^j \left[Y_j(k, n_j) - \boldsymbol{B}_j(n_j) U(k, n_0) \right] = 0. \tag{10}$$

We can define the order ρ of such a relation as follows. Since some of the elements of ω may be zero, there is a largest index $\hat{n}$ such that the $\hat{n}$th element of ω^i for some i is nonzero but the $(\hat{n}+1)$st through the (n_j+1)st elements of each ω^j are zero. Then ρ is defined to be $\hat{n}-1$. The order ρ describes the "memory span" of the redundancy relation. For example, when $\rho = 0$, instantaneous outputs of sensors are involved. When $\rho > 0$, a time window of size $\rho+1$ of sensor outputs and actuator inputs are considered in the parity equation. For example, (3) is a first-order parity relation.

To provide more insights into the nature of parity relations, it is useful to examine several examples.

1) Direct Redundancy: Suppose there are ω^j's of the form

$$\omega^j = \left[\omega_0^j, 0, \cdots, 0 \right]$$

where at least two of the ω_0^j's are nonzero, and they satisfy (7). Then we have the following direct redundancy relation:

$$\left[\omega_0^1, \cdots, \omega_0^M \right] \begin{bmatrix} y_1(k) \\ \vdots \\ y_M(k) \end{bmatrix} = 0.$$

Note that the above expression represents a zeroth order parity equation.

2) A Single Sensor: Equation (6) implies that it is always possible to find a nonzero ω^j such that

$$\omega^j \left[Y_j(k, n_j) - \boldsymbol{B}_j(n_j) U(k, n_0) \right] = 0. \tag{11}$$

Note that equation (11) is of order n_j, and it is a special case of (10). (That is why we have used n_j instead of $n_j - 1$ in (7) in order to include this type of temporal redundancy.) Since this redundancy relation involves only one sensor the parity function defined by the left-hand side of (11) may be used as the residual for a self-test for sensor j, if $\boldsymbol{B}_j(n_j) = 0$ or if the actuators can be verified (by other means) to be functioning properly. Similarly, it can be used to detect actuator failures if sensor j can be verified to be normal. Equation (4) (in which $v(k)$ is directly measured) represents an example of this type.

Alternatively, (11) can be rewitten as

$$y_j(k) = -\left(\omega_{n_j}^j \right)^{-1} \left[\sum_{t=1}^{n_j} \omega_{n_j - t}^j y_j(k-t) - \sum_{t=1}^{n_j} \sigma_{n_j - t}^j u(k-t) \right] \tag{12}$$

where

$$\left[\sigma_0^j, \cdots, \sigma_{n_j-1}^j, 0, \cdots, 0 \right] = \omega^j \boldsymbol{B}_j(n_j)$$

σ_t^j, $t = 0, \cdots, n_j - 1$, is a q-dimensional row vector, and ω_t^j, $t = 0, \cdots, n_j - 1$ is the $(t+1)$st component of ω^j. Equation (12) represents a reduced-order ARMA model for the jth sensor alone. That is to say, the output of sensor j can be predicted from its past outputs and past actuator inputs according to (12). Based on the ARMA model several methods of residual generation are possible. We will discuss this further in Section III.

3) Temporal Redundancy Between Two Sensors: A temporal redundancy exists between sensor i and sensor j if there are

$$\omega^i = \left[\omega_0^i, \cdots, \omega_{n_i-1}^i, 0 \right]$$

$$\omega^j = \left[\omega_0^j, \cdots, \omega_{n_j-1}^j, 0 \right]$$

satisfying the redundancy relation

$$\left[\omega^i, \omega^j \right] \left\{ \begin{bmatrix} Y_i(k_1 n_1) \\ Y_j(k, n_j) \end{bmatrix} - \begin{bmatrix} \boldsymbol{B}_i(k, n_j) \\ \boldsymbol{B}_j(k, n_j) \end{bmatrix} U(k, n_0) \right\} = 0. \tag{13}$$

Equation (13) is a special case of the general form of parity equation (10) in the no-fail situation with $\omega^s = 0$ for $s \neq i$, $s \neq j$. The relation (13) is of order $\rho < \max(n_i, n_j)$. Clearly, (13) holds if and only if

$$\left[\omega_0^i, \cdots, \omega_{n_j-1}^i \right] \boldsymbol{C}_i(n_i - 1) = \left[\omega_0^j, \cdots, \omega_{n_j-1}^j \right] \boldsymbol{C}_j(n_j - 1)$$

and, this implies that a redundancy relation exists between two sensors if their observable subspaces overlap. Furthermore, when the overlap subspace is of dimension $\tilde{n}$, there are $\tilde{n}$ linearly independent vectors of the form $[\omega^i, \omega^j]$ that will satisfy (13). Note that (3) (with both $v(k)$ and $a(k)$ measured) represents an example of this type.

Because the order of (13) is ρ, either ω_ρ^i or ω_ρ^j must be nonzero. Assuming $\omega_\rho^j \neq 0$, we can rewrite (13) in an ARMA representation for sensor j as in (12)

$$y_j(k) = -\left(\omega_\rho^j \right)^{-1} \left[\sum_{t=1}^{\rho} \omega_{\rho-t}^j y_j(k-t) + \sum_{t=0}^{\rho} \omega_{\rho-t}^i y_i(k-t) - \sum_{t=0}^{\rho} \left(\sigma_{\rho-t}^i + \sigma_{\rho-t}^j \right) u(k-t) \right]. \tag{14}$$

That is, the parity relation (13) specifies an ARMA model for the jth sensor, with the original system input u *and* the ith sensor output acting as inputs to this reduced-order model. This clearly has a close relationship to the nature of the input–output structure of the original state space model. In [17] this connection is examined more explicitly in terms of representing parity relations as dynamic systems which, when cascaded with (1), produce exactly zero output under normal operation. In general, any parity relation specifies an ARMA model for some sensor driven by u and by possibly all of the other sensor outputs.

III. Residual Generation for FDI

In the first part of this section we discuss alternative residual generation procedures, and in the latter half of the section we discuss how such residuals, once generated, can be used for failure detection. To avoid obscuring the simple ideas we wish to explain, we carry out our development in this section for a second-order system ($N = 2$) in the form of (1) with the following parameters:

$$A = \begin{bmatrix} a_{11} & a_{12} \\ 0 & a_{22} \end{bmatrix}, \qquad b = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$
$$c_1 = [1 \quad 0]$$
$$c_2 = [0 \quad 1]. \tag{15}$$

In this case $n_1 = 2$, $n_2 = 1$, and $n - N = 3$. Therefore, there are only three linearly independent parity equations which may be written as

$$\begin{aligned} y_1(k) - (a_{11} + a_{22}) y_1(k-1) + a_{11} a_{22} y_1(k-2) \\ - a_{12} u(k-2) = 0 \\ y_1(k) - a_{11} y_1(k-1) - a_{12} y_2(k-1) = 0 \\ y_2(k) - a_{22} y_2(k-1) - u(k-1) = 0. \end{aligned} \tag{16}$$

Note that these represent temporal redundancies.

Residual Generation Based on Parity Relations

For a zeroth order parity relation (i.e., a direct redundancy relation) the residual is the corresponding parity function. For a higher order parity relation (temporal redundancy), there are three possible methods for the residual generation. We will illustrate these using the second parity equation of (16).

1) Parity Function as Residuals: Just as with direct redundancy relations, the parity function itself can be taken as a residual. For our specific example, this would be

$$r_1(k) = y_1(k) - a_{11}y_1(k-1) - a_{12}y_2(k-1). \quad (17)$$

Such a residual is a moving average process, i.e., it is a function of a sliding window of the most recent sensor output and (possibly) actuator input values. It is useful to note the effect of noise and failures on the residual. Specifically, if the sensor outputs are corrupted by white noise, the parity function values will be correlated over the length of the window. In our example, $r_1(k)$ is correlated with $r_1(k-1)$ and $r_1(k+1)$ but not with any of its values removed by more than one time step.

The effect of a failure on a parity function depends, of course, on the nature of the failure. To illustrate what typically occurs, consider the case in which one sensor develops a bias. Since the parity function is a moving average process it also develops a bias, taking at most ρ steps to reach the steady-state value. In our example, if $y_2(k)$ develops a bias of size β at time θ, $r_1(k)$ will have a bias of size $-a_{12}\beta$ from time $\theta+1$ on.

For a more general case, e.g., (14), $r_1(k)$ is simply the difference between the two sides of the equation. The form of a failure signature will depend on the precise nature of the RHS of (14), but it can be analyzed for each individual case. Note that extension to cases where the RHS of (14) contains more than two components of y is straightforward.

2) Open-Loop Residuals: As discussed in the preceding section, any temporal redundancy relation specifies an ARMA model. In our example we have the model

$$y_1(k) = a_{11}y_1(k-1) + a_{12}y_2(k-1). \quad (18)$$

This equation leads naturally to a second residual generation procedure: solve (18) recursively using as initial condition the actual initial value of the first sensor output and then using the actual value of the second sensor in the recursion; compare the result at each instant of time to the actual output of sensor 1. That is, we compute

$$\tilde{y}_1(k) = a_{11}\tilde{y}_1(k-1) + a_{12}y_2(k-1) \quad (19)$$

with

$$\tilde{y}_1(0) = y_1(0)$$

and the resulting residual is

$$r_2(k) = y_1(k) - \tilde{y}_1(k).$$

The behavior of this residual is decidedly different from that of $r_1(k)$. In particular, $r_2(k)$ is not a moving average of previous values as it involves the integration of $y_2(k)$. Thus, if $y_1(k)$ and $y_2(k)$ are corrupted by white noise, $r_2(k)$ will in general be correlated with all of its preceding and future values. For example, if $a_{11}=1$, $r_2(k)$ is nothing but a random walk.

The effect of failure is also different in $r_2(k)$. For example, if $y_2(k)$ develops a bias, this bias will be integrated in (19). In particular, if $a_{11}=1$, $r_2(k)$ will develop a ramp of slope $-a_{12}\beta$ at time $\theta+1$ if sensor 2 develops a bias of size β at time θ.

This method of residual generation can be used with a more general parity relation. For example, (14) may be integrated using previous actual values of y_j as initial conditions and the actual input u and the measured values of y_i as inputs. Then this resulting open-loop prediction of y_j is subtracted from the actual value of y_j to produce the residual. Again, the effect of particular failures on the residual can be computed in a straightforward manner.

3) Closed-Loop Residuals: A third method of residual generation is also based on the ARMA model (18), but explicitly taking noise into account. Specifically, we write each output as its noise-free value plus noise:

$$y_i(k) = y_{i0}(k) + v_i(k). \quad (20)$$

Then, from (18) we obtain the equation

$$y_{10}(k) = a_{11}y_{10}(k-1) + a_{12}y_2(k-1) - a_{12}v_2(k-1). \quad (21)$$

Note that the known driving term here is the actual sensor output, and thus the noise on this output becomes a driving noise for the model (21). Given this model and the noisy measurement $y_1(k)$ of $y_{10}(k)$ we can design a Kalman filter

$$\hat{y}_{10}(k) = a_{11}\hat{y}_{10}(k-1) + a_{12}y_2(k-1) + Hr_3(k)$$

where H is the Kalman gain and the residual is the innovations

$$r_3(k) = y_1(k) - a_{11}\hat{y}_{10}(k-1) - a_{12}y_2(k-1).$$

As in the preceding cases, this residual generation method generalizes to cases such as (14) in an obvious way.

In this case, $r_3(k)$ is an uncorrelated sequence. Also, if $y_2(k)$ develops a bias at time θ, the trend in $r_3(k)$ will be time-varying. Specifically, it will begin at time $\theta+1$ as a ramp, but will level off to a steady-state bias due to the closed-loop nature of the residual generation process.

All three of these residual generation procedures have been used in practice. For example, parity functions have found many applications, ranging from gyro failure detection [7], [8] to the validation of signals in nuclear plants [13]. The open-loop method was used in detecting sensor failures on the F-8 aircraft [10], as was the closed-loop method, which has also been used in such applications as electrocardiogram analysis [6] and maneuver detection [15]. Our contribution here is to expose the fundamental relationships among these in general.

At first glance, it might seem that the closed-loop method is the logical method to use in that, if the sensor noise is white, it produces an uncorrelated sequence of residuals rather than a correlated one that would have to be whitened in an "optimal" detection system. In fact, going one step further, it would seem decidedly suboptimal to use only one or several redundancy relations rather than *all* of them. That is, the "optimal" approach would seem to be designing a Kalman filter based on the entire model (1). This, however, is true only in the most ideal of worlds in which our knowledge of the system dynamics is perfect. When model uncertainties are taken into account it is not at all clear that this is what one should do. Rather, it would seem reasonable to identify only the most robust redundancy relations and then to structure failure detection systems based on these. This leads to two obvious questions:

1) How does one define and determine robust redundancy relations?

2) Given a set of such relations, how does one use them in concert in designing a failure detection system?

In the second part of this section we discuss the second of these questions, while the first is addressed in the next section.

In the remainder of this paper we will focus on using the parity function method of residual generation as this is the simplest analytically. It should be noted that residuals generated this way are correlated. This is because the residuals are based on a sliding window of system inputs and outputs. If one used the closed-loop method, one would obtain whitened residuals. Due to the infinite memory associated with the closed-loop and open-loop methods, the effect of model uncertainties tends to be propagated in these

procedures. Hence, the nature of the robustness problem of these residual generation processes is likely to be different from the parity function method. Despite this difference, the study of the parity function method allows us to gain considerable insight and develop some useful techniques for robust failure detection.

Use of Parity Functions in a Failure Detection System

Now we discuss how the residuals generated using parity functions can be used for failure detection. In this discussion we will not be concerned with the detailed decision process, which would involve specific statistical tests, but we will focus on the *geometry* of the failure detection problem. First we will examine (sensor) FDI using direct redundancy. This is the case that has been examined in most detail in the literature, for example, in the work of Evans and Wilcox [7], Gilmore and McKern [8], Potter and Suman [11], Daley *et al.* [12], and Desai and Ray [13]. We include this brief discussion of concepts developed by others in order to provide for a basis for our discussion of their extention to include temporal redundancy relations.

Consider a set of M sensors with output vector $y(k) = [y_1(k), \cdots, y_M(k)]'$ and a parity vector

$$P(k) = \Omega y(k) \tag{22}$$

where Ω is a matrix with M columns and a number of rows (the specification of which will be discussed later). From Section II, we see that Ω is not unique, and for any choice of Ω such that (22) is a parity vector, we know that $P(k)$ will be zero in the absence of a failure (and no noise). However, the nature of failure signatures contained in the parity vector depends heavily on the choice of Ω. Clearly Ω should be chosen so that failure signatures are easily recognizable. In the following we will describe two approaches for achieving this purpose.

One way of using the parity vector for FDI is via what we term a *voting scheme*. To implement the voting scheme, we need a set of parity relations such that each component (i.e., sensor or actuator) of interest is included in at least one parity relation and each component is excluded from at least one parity relation. When a component fails, all the parity relations involving it will be violated,[1] while those excluding it will still hold. This means that the components involved in parity relations that hold can be immediately declared as unfailed, while the component that is common to all violated parity relations is readily identified as failed. This is the basic idea of voting that is used in [7], [8]. In fact, for the detection and identification of a single failure among M components at least $M-1$ parity relations are required.[2] Therefore, the number of rows in Ω should be at least $M-1$, and the rows of Ω should be chosen to satisfy the above criterion on the set of parity relations. Furthermore, we note that at least three components are needed for voting and that it may not be possible to determine a required Ω in many applications, in which case the use of temporal redundancy is absolutely necessary.

Another method which uses more information about how failures affect the residuals has been examined by Potter and Suman [11], and Delay *et al.* [12]. This method exploits the following phenomenon. A faulty sensor, say the jth one, contains an error signal $\nu(k)$ in its output

$$y_j(k) = c_j x(k) + \nu(k). \tag{23}$$

The effect of this failure on the parity vector defined by (22) is

$$P(k) = \Omega_j \nu(k)$$

where Ω_j is the jth column of Ω. That is, no matter what $\nu(k)$ is, the effect of a sensor j failure on the residual always lies in the direction Ω_j. Thus, a sensor j failure can be identified by recognizing a residual bias in the Ω_j direction. We refer to Ω_j as the *failure direction in parity space* (FDPS) corresponding to sensor j. (In [11] Ω_j is referred to as the jth measurement axis in parity space.)

It is now clear that Ω should be chosen to have distinct columns, so that a sensor failure can be inferred from the presence of a residual bias in its corresponding FDPS. (Note that an Ω suitable for the voting scheme has M distinct columns.) In principle, an Ω with as few as two rows but M distinct columns is sufficient for detecting and identifying a failure among the M sensors. In practice, however, increasing the row dimension of Ω can help to separate the various FDPS's and increase the distinguishability of the different failures under noisy conditions.

The two FDI methods discussed above can also be used with temporal redundancy. In a voting scheme, one can see that the same logic applies. (In fact, additional self-tests may be performed for the sensors providing corroboratory information which is of great value when noise is present.) Consider next the extention of the second failure detection method to temporal redundancy relations. In this case, it is generally not possible to find an Ω to confine the effect of each component failure to a fixed direction in parity space. To see this, consider the parity relations (16). We can write the parity vector as

$$P(k) = \begin{bmatrix} 1 & -(a_{11}+a_{22}) & a_{11}a_{22} & 0 & 0 \\ 1 & -a_{12} & 0 & 0 & -a_{12} \\ 0 & 0 & 0 & 1 & -a_{22} \end{bmatrix} \begin{bmatrix} y_1(k) \\ y_1(k-1) \\ y_1(k-2) \\ y_2(k) \\ y_2(k-1) \end{bmatrix} + \begin{bmatrix} 0 & -a_{12} \\ 0 & 0 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} u(k-1) \\ u(k-2) \end{bmatrix}.$$

When sensor 2 fails [with output model (23)], the residual vector develops a bias of the form

$$P(k) = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \nu(k) + \begin{bmatrix} 0 \\ -a_{12} \\ -a_{22} \end{bmatrix} \nu(k-1). \tag{24}$$

Unless $\nu(k)$ is a constant, the effect (signature) of a sensor 2 failure is only confined to a two-dimensional subspace of the parity space. In fact, generally when temporal redundancy is used in the parity function method for residual generation, failure signatures are generally constrained to multidimensional subspace in the parity space. These subspaces may in general overlap with one another, or some may be contained in others. If no such subspace is contained in another, identification of the failure is still possible by determining which subspace the residual bias lies in. We note that the detection filters of Beard [2] and Jones [3] effectively acts, in a closed-loop fashion, to confine the signature of an actuator failure to a single direction and that of a sensor failure to a two-dimensional subspace in the residual space.

As we indicated previously, the second approach to using parity functions for FDI uses some information about the nature

[1] "Violation" can be defined in a variety of ways. Typically, one compares the residual value to a threshold determined by some means (e.g., one may use a statistical criterion to set the threshold to achieve a specified false alarm–correct detection tradeoff). Alternatively, one may use the average of the residual over a sliding window to improve the tradeoff.

[2] The logic used here has to be modified slightly. If each of the M-1 components is excluded from a different parity relation and the remaining component is involved in all parity relations, then violation of all parity relations indicates the failure of this last component, and failures in the other components can be identified using the above logic. In practice, more than $M-1$ relations are preferred for better performance in noise.

of the failure signatures. Specifically, it uses information concerning the subspaces in which the signatures evolve. In this approach no attempt is made to use any information concerning the temporal structure of this evolution. (For example, no assumption was made about the evolution of $\nu(k)$ in (24).) In some problems (e.g., in [6], [10]) one may be able to model the evolution of failures as a function of time. In this case, the temporal signature of the failure (in addition to the subspace information discussed above) can be determined. (If, for instance, $\nu(k)$ in (23) is modeled in a particular way, then one immediately obtains a model of the evolution of $P(k)$ in (24).) Such information can be of further help in distinguishing the various failures, especially in the case where temporal redundancy is used. Detection systems such as GLR [4]–[6] heavily exploit such information contained in the residual.

IV. Parity Relations for Robust Residual Generation

In this section we discuss the issue of robust failure detection in terms of the notions introduced in the previous section. The need for this development comes from the obvious fact that in any application a deterministic model such as (1) is quite idealistic. In particular, the true system will be subjected to noise and parameter uncertainty. If noise alone were present one could take this into account, as we have indicated, through the design of a statistical test based on the generated residuals (see, for example, [4], [10]). The problem of parameter uncertainty can be handled in two ways: 1) estimate the effect of such uncertainty and compensate the FDI system for it [5], [18], [19]; or 2) minimize the sensitivity of the FDI system to such uncertainty [10], [16], [17], [20]. The latter approach is the one we have adopted.

The starting point of our development is a model that has the same form as (1) but includes noise disturbance and parameter uncertainty:

$$x(k+1) = A(\gamma)x(k) + \sum_{j=1}^{q} b_j(\gamma)u_j(k) + \xi(k) \quad (25a)$$

$$y_j(k) = c_j x(k) + \eta_j(k) \quad (25b)$$

where γ is the vector of uncertain parameters taking values in a specified subset Γ of $\boldsymbol{R}^m$. This form allows the modeling of elements in the system matrices as uncertain quantities that may be functions of a common quantity. The vectors ξ and $\eta = [\eta_1, \cdots, \eta_M]'$ are independent, zero-mean, white Gaussian noise vectors with constant covariance matrices $Q(\geq 0)$ and $R(>0)$, respectively. In this section we consider the problem of determining useful parity relations that can be used for FDI for the system described by (25).

The Structure and Coefficients of a Parity Function

Before we continue with the discussion, it is useful to define the *structure* and the *coefficients* of a parity function. Recall that a parity function is essentially a weighted combination of a (time) window of sensor outputs and actuator inputs. The *structure* of a parity function defines which input and output elements are included in this window, and the *coefficients* are the (nonzero) weights corresponding to these elements. A scalar parity function $p(k)$ can be written as

$$p(k) = \alpha \boldsymbol{Y}(k) + \beta \boldsymbol{U}(k) \quad (26)$$

where $\boldsymbol{Y}(k)$ and $\boldsymbol{U}(k)$ denote the vectors containing the output and input elements in the parity function, respectively. Together, $\boldsymbol{Y}(k)$ and $\boldsymbol{U}(k)$ specify the parity structure, and the row vectors α an β contain the parity coefficients. Consider, for example, the first parity function of (16). Its corresponding $\boldsymbol{Y}(k)$, $\boldsymbol{U}(k)$, α, and β are

$$\boldsymbol{Y}(k) = [y_1(k-2), y_1(k-1), y_1(k)]'$$
$$\boldsymbol{U}(k) = u(k-2)$$
$$\alpha = [a_{11}a_{22}, -(a_{11}+a_{22}), 1]$$
$$\beta = -a_{12}.$$

Under model (25), $\boldsymbol{Y}(k)$ has the form

$$\boldsymbol{Y}(k) = \boldsymbol{C}(\gamma)x(k-\rho) + \Phi(\gamma)\tilde{\xi}(k) + \boldsymbol{B}(\gamma)\boldsymbol{U}(k) + \tilde{\eta}(k) \quad (27)$$

where ρ is the order of the parity function, and

$$\tilde{\xi}(k) = [\xi'(k-\rho), \cdots, \xi'(k-1)]'.$$

The components of $\tilde{\eta}(k)$ and $\boldsymbol{U}(k)$, and the rows of $\boldsymbol{C}(\gamma)$, $\Phi(\gamma)$, and $\boldsymbol{B}$ are determined from (25) and the structure of $\boldsymbol{Y}(k)$. If, specifically, the ith component of $\boldsymbol{Y}(k)$ is $y_s(k-\sigma)$, then the ith component of $\tilde{\eta}(k)$ is

$$\tilde{\eta}_i(k) = \eta_s(k-\sigma).$$

The vectors $\tilde{\xi}$ and $\tilde{\eta}$ are independent zero-mean Gaussian random sequences with constant covariances $\boldsymbol{Q}$ and $\boldsymbol{R}$, respectively. The matrix $\boldsymbol{Q}$ is block diagonal with Q on the diagonal; $\boldsymbol{R}_{ij} = R_{st}\delta_{\sigma\tau}$, where $\boldsymbol{R}_{ij}$ is the (i,j)th element of $\boldsymbol{R}$, $\delta_{\sigma\tau}$ is the Kronecker delta function, R_{st} is the (s,t)th element of R, and the ith element of $\boldsymbol{Y}(k)$ is $y_s(k-\sigma)$, while the jth element is $y_t(k-\tau)$. The ith row of $\boldsymbol{C}(\gamma)$, i.e., $\boldsymbol{C}(i,\gamma)$ is

$$\boldsymbol{C}(i,\gamma) = c_s A^{\rho-\sigma}.$$

The ith row, $\Phi(i,\gamma)$ of $\Phi(\gamma)$ (which has ρN columns) is

$$\Phi(i,\gamma) = \left[c_s A^{\rho-\sigma-1}, c_s A^{\rho-\sigma-2}, \cdots, c_s, 0, \cdots, 0\right].$$

Note that $x(k-\rho)$ is a random vector that is uncorrelated with $\tilde{\xi}$ and $\tilde{\eta}$, and

$$E\{x(k-\rho)\} = x_0(k-\rho)$$
$$\operatorname{cov}\{x(k-\rho)\} = \Sigma(\gamma)$$

where $\Sigma(\gamma)$ is the (steady-state) covariance of $x(k-\rho)$ and it is dependent on γ through $A(\gamma)$ and $B(\gamma)$.

The matrix $\boldsymbol{B}$ and the vector $\boldsymbol{U}(k)$ are determined as follows. First, collect into a matrix $\tilde{\boldsymbol{B}}$ all the rows in $\boldsymbol{B}_j(k,\gamma)$ [see (9)] corresponding to $\boldsymbol{C}(i,\gamma)$. Then, collect all the nonzero columns of $\tilde{\boldsymbol{B}}$ into $\boldsymbol{B}$ and the corresponding components of u in the window into $\boldsymbol{U}(k)$.

In the preceding section, we defined parity functions as linear combinations of inputs and outputs that would be identically zero in the absence of noise. When parameter uncertainties are included, however, it is not possible in general to find any parity functions in this narrow sense. In particular, with reference to the function $p(k)$ defined by (26) and (27) this condition would require that $\alpha\boldsymbol{C}(\gamma) = 0$ *for all* $\gamma \in \Gamma$. Consequently, we must modify our notion of a useful parity relation. Intuitively, any given parity structure will be useful for failure detection if we can find a set of parity coefficients that will make the resulting function $p(k)$ in (26) *close* to zero for all values of $\gamma \in \Gamma$ when no failure has occurred. When considering the use of such a function for the detection of a particular failure one would also want to guarantee that $p(k)$ deviates significantly from zero for all $\gamma \in \Gamma$ when this failure occurs. Such a parity structure–coefficient combination approximates the true parity function defined in Section II. Our approach to the robustness issue is founded on this perspective of the FDI design problem, and we will choose

parity structures and coefficients that display these properties. From this vantage point, it is not necessary to base a parity structure on a C with linearly dependent rows. Of course, the closer the rows of C are to being dependent the less the value of the state $x(k-\rho)$ will affect the value of the approximate parity function, i.e., the closer the approximate parity function is to being a true parity function.

Determination of Parity Structure and Coefficients

Clearly, there are many candidate parity structures for a given system. For a voting system, the requirements on Ω as described in Section III help to limit the number of such candidates that must be considered. In addition special features of the system under consideration typically provide additional insights into the choice of candidate parity structures. Given the set of candidate structures one is faced with the problem of finding the best coefficients for each and then with comparing the resulting candidates. In this paper we will not address the problem of defining the set of candidate structures (as this is very much a system-specific question) but will assume that we have such a set of structures,[3] and we will proceed to consider the problem of determining the coefficients for these structures and their comparison. In the following we will describe a method for choosing robust parity functions. Although this approach represents only one method of solving the problem, it serves well to illustrate the basic ideas of a useful design methodology.

The parity function design problem is approached in two steps: 1) coefficients that will make the candidate parity functions close to zero under the no-fail situation are determined; 2) the resulting parity functions that provide the most prominent failure signatures for a specified failure will be chosen. We will consider the coefficient design problem first.

We are concerned with the choice of the coefficients, α and β for the parity function

$$p(k)=\alpha[C(\gamma)x(k-\rho)+\Phi(\gamma)\tilde{\xi}(k)+B(\gamma)U(k)+\tilde{\eta}(k)]-\beta U(k).$$

Note that the dependence of $p(k)$ on α, β, γ, $x(k-\rho)$, and $U(k)$. As $p(k)$ is a random variable, a convenient measure of the magnitude (squared) of $p(k)$ is its variance, $E\{p^2(k)\}$, where the expectation is taken with respect to the joint probability density of $x(k-\rho)$, $\tilde{\xi}(k)$, and $\tilde{\eta}(k)$ with the mean $x_0(k-\rho)$ and the value of $U(k)$ assumed known. As we will discuss shortly, this can be thought of as specifying a particular operating condition for the system. Note also that the statistics of $x(k-\rho)$ depend on γ. Define

$$e(\alpha,\beta)=\max_{\gamma\in\Gamma}E\{p^2(k)\}. \tag{28}$$

The quantity $e(\alpha,\beta)$ represents the worst case effect of noise and model uncertainty on the parity function $p(k)$ and is called the *parity error* for $p(k)$ with the coefficients α and β. We can attempt to achieve a conservative choice of the parity coefficients by solving

$$\min_{\alpha,\beta}e(\alpha,\beta).$$

Since it has a trivial solution ($\alpha=0,\beta=0$) this optimization problem has to be modified in order to give a meaningful solution. Recall that a parity equation primarily relates the sensor outputs, i.e., a parity equation always includes output terms but not necessarily input terms. Therefore, α must be nonzero. Without loss of generality, we can restrict α to have unit magnitude. The actuator input terms in a parity relation may be regarded as serving to make the parity function zero so that β is nominally free. In fact, β has only a single degree of freedom. Any β can be written as $\beta=\lambda U'(k)+z'$, where z is a (column) vector orthogonal to $U(k)$. The component z' in β will not produce any effect on $p(k)$. This implies for each $U(k)$ we only have to consider β of the form $\beta=\lambda U'(k)$, and we have the following minimax problem:

$$\min_{\substack{\alpha,\lambda\\ s.t.\ \alpha\alpha'=1}}\ \max_{\gamma\in\Gamma}E\{p^2(k)\} \tag{29}$$

where

$$E\{p^2(k)\}=[\alpha,\lambda]S[\alpha,\lambda]'$$

and S is the symmetric positive-definite matrix

$$S=\begin{bmatrix}S_{11}&S_{12}\\S_{21}&S_{22}\end{bmatrix}$$

$$\begin{aligned}S_{11}&=C(\gamma)[x_0(k-\rho)x_0'(k-\rho)+\Sigma(\gamma)]C'(\gamma)\\&\quad+\Phi(\gamma)Q\Phi'(\gamma)+R+B(\gamma)U(k)U'(k)B'(\gamma)\\&\quad+C(\gamma)x_0(k-\rho)U'(k)B'(\gamma)\\&\quad+B(\gamma)U(k)x_0'(k-\rho)C'(\gamma)\\S_{12}&=S_{21}'=-S_{22}^{1/2}[B(\gamma)U(k)+C(\gamma)x_0(k-\rho)]\\S_{22}&=[U'(k)U(k)]^2.\end{aligned}$$

Let α^* and λ^* denote the values of α and λ that solve (29), with $\beta^*=\lambda^*U'(k)$. Let e^* be the minimax parity error of (29), i.e., $e^*=e(\alpha^*,\beta^*)$. Then e^* is the parity error corresponding to the parity function $p^*(k)=\alpha^*Y(k)+\beta^*U(k)$. The quantity e^* measures the usefulness of $p^*(k)$ as a parity function around the operating point specified by $x_0(k-\rho)$ and $U(k)$.

Although the objective function of (29) is quadratic in α and λ, (29) is generally very difficult to solve, because S may depend on γ arbitrarily. (See [16] and the next section for a discussion of the solution to some special cases.) The dependence on γ can be simplified somewhat by the following approximation. Recall that the role of a parity equation is to relate the outputs and inputs at different points in time. The matrices C, Φ, and B, which specify the dynamics of the system, thus have the dominant effect on the choice of a parity equation. From this vantage point the primary effect of the uncertainty in γ is typically manifested through the direct influence of these matrices on the matrix S, rather than through the indirect effect they have on $\Sigma(\gamma)$. Said another way, the variation in S as a function of γ is dominated by the terms involving C, Φ, and B, and in this case one introduces only a minor approximation by replacing $\Sigma(\gamma)$ by a constant Σ. This is equivalent to assuming the likely variations in the state do not change as a function of γ. With this approximation the S matrix shown above can be simplified, and we will use this approximation throughout the remainder of the paper.

Note that the dependence of $e(\alpha,\beta)$ on $x_0(k-\rho)$ and $U(k)$ indicates that the coefficients in principle should be computed at each time step if $x_0(k-\rho)$ and $U(k)$ are changing with time. This is clearly an undesirable requirement. Typically, a set of coefficients will work well for a range of values of $x_0(k-\rho)$ and $U(k)$. Therefore, a practical approach is to schedule the coefficients according to the operating condition. Each operating condition may be treated as a set-point, which is characterized by some nominal state x_0 and input U_0 that are independent of time. Parity coefficients can be precomputed [by solving (29) with x_0 and U_0 in place of $x_0(k-\rho)$ and $U(k)$] and stored. Then the appropriate coefficients can be retrieved for use at the corresponding set-point. When the state and the input are varying slowly, this scheme of scheduling coefficients is likely to deliver performance close to the optimum.

[3] This set could be *all* structures up to a specified order, which is a finite set.

If a more accurate approximation is desired, the coefficients scheduling scheme described above can be modified to account for variations in the input due, for example, to regulation of the system at the set point x_0. In particular, one can consider modeling $\boldsymbol{U}(k) = \boldsymbol{U}_0 + \delta\boldsymbol{U}(k)$, where $\delta\boldsymbol{U}(k)$ is a (stationary) zero-mean random process that models the deviation of the input from the nominal $\boldsymbol{U}_0$. With this modification, the expectation of $p^2(k)$ has to be taken with respect to the joint probability density of $x(k-\rho)$, $\tilde{\xi}(k)$, $\tilde{\eta}(k)$, and $\delta\boldsymbol{U}(k)$ with x_0 and $\boldsymbol{U}_0$ fixed. This will lead to a more complex S matrix. Furthermore, the vector β will no longer be constrained but completely free. However, the general form of the optimization problem remains unchanged.

One approach to circumvent the requirement of solving the coefficient design problem for many values of x_0 and $\boldsymbol{U}_0$ is to modify (29) to be

$$\min_{\substack{\alpha,\lambda \\ s.t.\ \alpha\alpha'=1}} \max_{\substack{\gamma\in\Gamma \\ x_0(k-\rho)\in X \\ \boldsymbol{U}(k)\in\Gamma}} E\{p^2(k)\} \tag{30}$$

where X and Γ denote the ranges of values that $x_0(k)$ and $\boldsymbol{U}(k)$ may take, respectively. This formulation yields approximations with simpler solution procedures than (29). Moreover, it leads to a single parity function over all operating conditions. We will not explore this approach here, but refer to the reader to [17]. Whether this alternative approach or our coefficient-scheduling method is more appropriate depends on the problem. If the state and control are likely to vary significantly and if $e(\alpha,\beta)$ is not that strong a function of x_0 and $\boldsymbol{U}_0$, the alternative approach would be appropriate. If, however, the state and control are likely to be near specific set points for periods of time, then (30) would give an overly conservative design. In this case, scheduling the parity coefficients (function) according to the operating point would yield superior performance.

With the coefficients and the associated parity errors determined for the candidate parity structures we can proceed to choose the parity functions for residual generation using the parity function method. As the squared magnitude of the coefficients $[\alpha,\beta]$ scales the parity error, the parity errors of different parity functions can be compared if they are normalized. We define the *normalized parity error* $\bar{e}^*$, the *normalized parity coefficients*, and the *normalized parity function* $\bar{p}^*(k)$, as follows:

$$\bar{e}^* = e^*/\theta$$
$$\bar{\alpha}^* = \alpha^*/\theta$$
$$\bar{\beta}^* = \beta^*/\theta$$
$$\bar{p}^*(k) = \bar{\alpha}^*\boldsymbol{Y}(k) - \bar{\beta}^*\boldsymbol{U}(k)$$

where

$$\theta^2 = [\alpha^*,\beta^*][\alpha^*,\beta^*]' = 1+\beta^*\beta^{*\prime}.$$

The parity functions with the smallest normalized parity errors are preferred as they are closer to being true parity functions under noise and model uncertainty, i.e., they are least sensitive to these adverse effects.

An additional consideration required for choosing parity functions for residual generation is that the chosen parity functions should provide the largest failure signatures in the residuals relative to the inherent parity errors resulting from noise and parameter uncertainty. A useful index for comparing parity functions for this purpose is the *signature-to-parity error ratio* π, which is the ratio between the magnitudes of the failure signature and the parity error. Using g to denote the effect of a failure on the parity function, π can be defined as

$$\pi = |g|/\bar{e}^*.$$

For the detection and identification of a particular failure, the parity function that produces the largest π should be used for residual generation. We give an example of this procedure in the next section.

Discussions

Since a large signature-to-parity error ratio is desirable, a logical alternative approach to the choice of parity structure and coefficients is to consider the signature-to-parity error ratio as the objective function in the minimax design. Although this is a more direct way to achieve the design goal, it requires solving a more difficult optimization problem than (29). The method described above and the example in the next section take advantage of the comparatively simple optimization problem to illustrate the essential idea of how to determine redundancy relations that are least vulnerable to noise and model errors. For different residual generation methods the measures of usefulness of parity functions, such as e and π in the above, may be different, but the basic design concept illustrated here still applies.

The minimax problem (29) can be replaced by a maximization if a probability density for the parameter γ can be postulated. That is, the design problem now takes the form

$$\max_{\substack{\alpha,\lambda \\ s.t.\ \alpha\alpha'=1}} E\{p^2(k)\}$$

where the expectation of $p^2(k)$ is taken with respect to the joint density of x, $\tilde{\xi}$, $\tilde{\eta}$, and γ. This formulation will give a much simpler optimization to be solved practically than the minimax problem (29). By treating γ as a random variable, (30), also simplified as the maximization over γ, is eliminated.

V. A Numerical Example

In this section we consider the problem of choosing parity functions and parity coefficients for a four-dimensional system operating at a set-point with two actuators and three sensors. The system matrices are shown in Table I. Except for two elements in the A matrix all parameters are known exactly. These two elements are assumed to be independent parameters denoted by γ_1 and γ_2.

Suppose we want to design a voting system for detecting a sensor failure. Three candidate parity structures are

$$p_1(k) = \alpha_1 \begin{bmatrix} y_2(k-1) \\ y_2(k) \\ y_1(k-1) \end{bmatrix},$$

$$p_2(k) = \alpha_2 \begin{bmatrix} y_2(k-2) \\ y_1(k-2) \\ y_1(k-1) \\ y_1(k) \end{bmatrix},$$

$$p_3(k) = \alpha_3 \begin{bmatrix} y_3(k-1) \\ y_3(k) \\ y_1(k-1) \end{bmatrix}$$

where the α_i's are row vectors (of parity coefficients) of appropriate dimensions. The corresponding Φ and $\boldsymbol{C}$ matrices are shown in Table II. Note that each $\boldsymbol{C}$ and Φ matrix depends linearly on either γ_1 or γ_2 and that the rows of $\boldsymbol{C}_2$ are not linearly dependent for any value of γ_2. The parity structures under consideration do not contain any actuator terms due to the fact that c_1B, c_2B, c_2AB, and c_3B are all zero. This will simplify the solution of the minimax problem without severely restricting

TABLE I
SYSTEM PARAMETERS

$$A = \begin{bmatrix} .5 & -.7 & .7 & 0 \\ 0 & .8 & \gamma_1 & 0 \\ -1 & 0 & 0 & .1 \\ 0 & 0 & \gamma_2 & .4 \end{bmatrix}$$

$$B = \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix}$$

$$c_1 = [0\ 0\ 1\ 0]'$$
$$c_2 = [0\ 1\ 0\ 0]'$$
$$c_3 = [0\ 0\ 0\ 1]'$$

$\gamma_1 \in [.02, .2]$ nominal $\gamma_1 = .1$

$\gamma_2 \in [-.2, -.1]$ nominal $\gamma_2 = -.15$

TABLE II
THE C AND Φ MATRICES

$$C_1 = \begin{vmatrix} 0 & 1 & 0 & 0 \\ 0 & .8 & \gamma_1 & 0 \\ 0 & 0 & 1 & 0 \end{vmatrix} \quad \Phi_1 = \begin{vmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{vmatrix}$$

$$C_2 = \begin{vmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -1 & 0 & 0 & 1 \\ -.5 & -.7 & -.7+.1\gamma_2 & .04 \end{vmatrix} \quad \Phi_2 = \begin{vmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & .1 & 0 & 0 & 1 & 0 \end{vmatrix}$$

$$C_3 = \begin{vmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & \gamma_2 & .4 \\ 0 & 0 & 1 & 0 \end{vmatrix} \quad \Phi_3 = \begin{vmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{vmatrix}$$

TABLE III
TEST CONDITIONS

TEST COND.		PARAMETERS
a	x_0 =	[0 0 0 0]'
	Q_1	DIAG R = [1 1 1]
b	x_0 =	[−4.16 7.03 4.06 −1.01]'
	Q_1	DIAG R = [1 1 1]
c	x_0 =	[4.06 2.90 5.80 −1.45]'
	Q_1	DIAG R = [1 1 1]
d	x_0 =	[4.06 2.90 5.80 −1.45]'
	Q_1	DIAG R = [1 2 2]
e	x_0 =	[4.06 2.90 5.80 −1.45]'
	Q_1	DIAG R = [2 1 1]
f	x_0 =	[4.06 2.90 5.80 −1.45]'
	Q_2	DIAG R = [1 1 1]

$$Q_1 = \begin{vmatrix} .25 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & .25 \end{vmatrix} \quad Q_2 = \begin{vmatrix} .25 & 0 & -.325 & 0 \\ 0 & .5 & 0 & 0 \\ -.325 & 0 & .625 & 0 \\ 0 & 0 & 0 & .25 \end{vmatrix}$$

TABLE IV
NOMINAL Σ

$$\Sigma_1 = \begin{vmatrix} .5580 & .0342 & -.1508 & -.0552 \\ .0342 & .0102 & -.0129 & -.0097 \\ -.1508 & -.0129 & .5772 & .0117 \\ -.0552 & -.0097 & .0117 & .3113 \end{vmatrix}$$

$$\Sigma_2 = \begin{vmatrix} 1.9580 & -.8434 & -1.1140 & -.1049 \\ -.8434 & 1.8030 & .7691 & -.1996 \\ -1.1140 & .7691 & 2.6080 & -.1081 \\ -.1040 & -.1996 & -.1081 & .3829 \end{vmatrix}$$

the discussion. Assuming a single sensor may fail, only p_3 plus p_1 or p_2 need to be used for residual generation (because both p_1 and p_2 include sensors 1 and 2). Therefore, in addition to the coefficient design problem, we have to rank the two parity structures p_1 and p_2 in order to determine which will give more robust residuals.

The minimax design problem has been solved for a set of six test conditions consisting of different set-points and different plant and sensor noise intensities. These test conditions are described in Table III. (The two set-points are obtained by applying $u_1 = 1$ or $u_2 = 10$ to the nominal system model.) The nominal state covariances Σ_1 and Σ_2 due to the two different plant noise intensities Q_1 and Q_2 are listed in Table IV. Due to the simple dependence of the parity functions on the γ's an efficient solution procedure is possible [16]. The resulting parity coefficients and the corresponding (normalized) parity errors are summarized in Table V.

It is evident that the parity coefficients in this example are strongly dependent on the test condition (i.e., the values of x_0, Q, and R). Although this dependence is very complex, some insights may be obtained from the numerical results. Consider, for instance, p_1 under conditions b and c. For condition b the parity function is

$$p_{1b}(k) = 0.6411y_2(k-1) - 0.7666y_2(k) + 0.0378y_1(k-1)$$

and for condition c it is

$$p_{1c}(k) = 0.8947y_2(k-1) - 0.3667y_2(k) - 0.2551y_1(k-1).$$

The only difference between these conditions lies in the value of

TABLE V
MINIMAX PARITY COEFFICIENTS AND PARITY ERRORS

TEST COND.	PAR. FUNC.	$\bar{e}^*$	$\bar{\alpha}^*$			
a	1	1.002	.7282	-.6808	.0791	
	2	1.008	.9983	.0223	.0483	.0219
	3	1.118	.6833	-.7208	-.1167	
b	1	1.082	.6411	-.7666	.0378	
	2	1.101	.4462	.5079	-.4356	.5942
	3	1.210	.7027	-.7115	-.0640	
c	1	1.096	.8947	-.3667	-.2551	
	2	1.055	.9599	-.1484	.1992	.1296
	3	1.230	.7592	-.6504	.0249	
d	1	1.908	.7865	.3023	-.5385	
	2	1.123	.7345	-.5931	.4697	-.6559
	3	2.228	.7981	-.6007	.0684	
e	1	1.124	.8058	-.5832	-.1025	
	2	1.122	.9669	-.1204	.1242	.1875
	3	1.230	.7441	-.6678	.1692	
f	1	1.427	.7327	-.6803	-.0166	
	2	1.311	.5146	.4404	-.3312	.6570
	3	1.254	.6385	-.7687	.0375	

x_0. Since the first and fourth columns of C_1 are zero, only the second and third elements of x_0 (x_{02} and x_{03}) will play a role in the coefficient optimization problem. The parity function p_1 can be written in the form

$$p_1 = \alpha_{11}x_{02} + \alpha_{12}(x_{02} + \gamma_1 x_{03}) + \alpha_{13}x_{03} + \zeta(\gamma_1, \alpha_1)$$

where α_{1i}, $i=1,2,3$ denote the elements of α_1 corresponding to $y_2(k-1)$, $y_2(k)$, and $y_1(k-1)$, respectively; ζ denotes the remaining noise terms. It is clear that x_{03} and α_{12} modulate the effect of γ_1 on p_1. Qualitatively, as $|x_{03}|$ becomes large relative to $|x_{02}|$ (with all noise covariances the same), the optimal α_{12} will reduce in size (relative to α_{11} and α_{13}) in order to keep the effect of γ_1 small. As $|x_{03}|$ increases, the signal-to-noise ratio of $y_1(k)$ also increases. Therefore, we expect $|\alpha_{13}|$ to become large to take advantage of the information provided by $y_1(k)$. Under condition b, $x_{02} > x_{03}$, and under condition c the reverse is true. An inspection of p_1 under these conditions as listed above shows that this reasoning holds. Therefore, built into the minimax problem is a systematic way of handling the tradeoff between uncertainty effects due to noise and error in system parameters.

Note that both p_1 and p_2 relate the first sensor to the second one, and p_2 is a higher order parity function than p_1. Furthermore, the rows of C_2 are not linearly dependent for any value of γ_2. However, the parity error associated with p_2 is smaller than that of p_1 in all conditions except conditions a and b. This shows that a higher order parity relation (which is more likely to contain higher order effects of γ) is not necessarily more vulnerable to model errors and noise. In addition, a parity function based on a C matrix with rows that are linearly dependent for all values of γ does not necessarily produce a smaller parity error than a parity function that is based on a C with independent rows.

In Table VI we have tabulated the signature-to-parity error ratio associated with the three parity functions for sensor failures that are modeled by a constant bias of size ν_i in the output for test conditions c and d. Here, π_i denotes the signature-to-parity error ratio for a bias failure in sensor i, and it is calculated by

TABLE VI
π VALUES FOR SELECTED TEST CONDITIONS

TEST COND.	PAR. FUNC.	π_1	π_2	π_3
c	p_1	$.243\nu_1$	$.504\nu_2$	-
	p_2	$.176\nu_1$	$.934\nu_2$	-
	p_3	$.022\nu_1$	-	$.107\nu_3$
d	p_1	$.390\nu_1$	$.788\nu_2$	-
	p_2	$.733\nu_1$	$.693\nu_2$	-
	p_3	$.046\nu_1$	-	$.126\nu_3$

substituting ν_i for y_i in the parity function (26) with the minimax coefficients. Such a table is helpful for determining the relative merits of p_1 and p_2. For instance, under condition d and assuming $\nu_1 = \nu_2$, p_2 is preferred to p_1 because it has a larger value of π_1 than p_2, while its π_2 value is comparable to that of p_2.

VI. CONCLUSIONS

In this paper we have characterized the notion of analytical redundancy in terms of a generalized parity space. We have described three methods for using parity relations to generate residuals for FDI. The problem of determining robust parity relations for residual generation using the parity function method was studied. This design task was formulated as an optimization problem, and an example was presented to illustrate the design methodology. A number of problem areas await further research. They include: a method for selecting useful parity structures for the parity coefficient problem studied in Section IV; solution procedures for the (minimax) optimization problem; and a method for determining robust parity relations for other methods of residual generation (i.e., the open-loop and the closed-loop methods).

REFERENCES

[1] A. S. Willsky, "A survey of design methods for failure detection in dynamic systems," *Automatica*, vol. 12, pp. 601–611, 1976.

[2] R. V. Beard, *Failure Accommodation in Linear Systems Through Reorganization*, Man Vehicle Lab., Mass. Inst. Technol., Cambridge, MA, Rep. MVT-71-1, Feb. 1971.

[3] H. L. Jones, "Failure detection in linear systems," Ph.D. dissertation, Dep. Aero. and Astro., Mass. Inst. Technol., Cambridge, MA, Sept. 1973.

[4] A. S. Willsky and H. L. Jones, "A generalized likelihood ratio approach to the detection and estimation of jumps in linear systems," *IEEE Trans. Automat. Contr.*, vol. AC-21, pp. 108–112, Feb. 1976.

[5] A. S. Willsky, E. Y. Chow, S. B. Gershwin, C. S. Greene, P. K. Houpt, and A. L. Kurkjian, "Dynamic model-based techniques for the detection of incidents on freeways," *IEEE Trans. Automat. Contr.*, vol. AC-25, pp. 347–360, June 1980.

[6] D. E. Gustafson, A. S. Willsky, and J. Y. Wang, "Final report: Cardiac arrhythmia detection and classification through signal analysis," The Charles Stark Draper Lab., Cambridge, MA, Rep. R-920, July 1975.

[7] F. A. Evans and J. C. Wilcox, "Experimental strapdown redundant sensor inertial navigation system," *J. Spacecr. Rockets*, vol. 7, pp. 1070–1074, Sept. 1970.

[8] J. P. Gilmore and R. A. McKern, "A redundant strapdown inertial reference unit (SIRU)," *J. Spacecr. Rockets*, vol. 9, pp. 39–47, Jan. 1972.

[9] A. Wald, *Sequential Analysis*. New York: Wiley, 1947.

[10] J. C. Deckert, M. N. Desai, J. J. Deyst, and A. S. Willsky, "F-8 DFBW sensor failure identification using analytic redundancy," *IEEE Trans. Automat. Contr.*, vol. AC-22, pp. 795–803, Oct. 1977.

[11] J. E. Potter and M. C. Suman, "Threshold redundancy management with arrays of skewed instruments," *Integrity Electron. Flight Contr. Syst.*, AGARDOGRAPH-224, pp. 15-11 to 15-25, 1977.

[12] K. C. Daley, E. Gai, and J. V. Harrison, "Generalized likelihood test for FDI in redundant sensor configurations," *J. Guidance Contr.*, vol. 2, pp. 9–17, Jan.–Feb. 1979.

[13] M. Desai and A. Ray, "A fault detection and isolation methodology," in *Proc. 20th Conf. Decision and Contr.*, Dec. 1981, pp. 1363–1369.

[14] R. W.Brockett, *Finite Dimensional Linear Systems*. New York: Wiley, 1970.

[15] R. J. McAulay and E. Denlinger, "A decision-directed adaptive tracker," *IEEE Trans. Aerosp. Electron. Syst.*, vol. AES-9, pp. 229–236, Mar. 1976.
[16] E. Y. Chow, "A failure detection system design methodology," Sc.D. thesis, Lab. Inform. Decision Syst., Mass. Inst. Technol., Cambridge, MA, 1980.
[17] X. C. Lou, "A system failure detection method—Failure projection method," S.M. thesis, Lab. Inform. Decision Syst., Mass. Inst. Technol., Cambridge, MA, 1982.
[18] G. G. Leninger, "Model degradation effects on sensor failure detection," in *Proc. 1981 Joint Automat. Contr. Conf.*, June 1981.
[19] S. R. Hall, "Parity vector compensation for FDI," S.M. thesis, Dep. of Aerosp. and Astro., Mass. Inst. Technol., Cambridge, MA, Feb. 1981.
[20] K. Watanabe and D. M. Himmelblau, "Instrument fault detection in systems with uncertainties," *Int. J. Syst. Sci.*, vol. 13, pp. 137–158, 1982.

Edward Y. Chow (S'77–M'80) was born in Hong Kong on August 5, 1952. He received the S.B., M.S., and Sc.D. degrees in electrical engineering from the Massachusetts Institute of Technology, Cambridge, in 1974, 1976, and 1981, respectively.

In 1976 he worked at The Charles Stark Draper Laboratory, Cambridge, MA. Since September 1980, he has been with Schlumberger-Doll Research, Ridgefield, CT, working on the application of estimation and signal processing theory to well-logging problems.

Dr. Chow is a member of Eta Kappa Nu and Tau Beta Pi.

Alan S. Willsky (S'70–M'73–SM'82) received both the S.B. and the Ph.D. degrees from the Massachusetts Institute of Technology, Cambridge, in 1969 and 1973, respectively.

From 1969 to 1973 he held a Fannie and John Hertz Foundation Fellowship. He joined the M.I.T. faculty in 1973 where his present position is Professor of Electrical Engineering. From 1974 to 1981 he served as Assistant Director of the M.I.T. Laboratory for Information and Decision Systems. He is also a founder and member of the Board of Directors of Alphatech, Inc., Burlington, MA. He has held visiting positions at Imperial College, London, England, and L'Université de Paris-Sud, Orsay, France. He is Editor of the M.I.T. Press series on signal processing, optimization, and control, and is the author of the research monograph *Digital Signal Processing and Control and Estimation Theory* and coauthor of the undergraduate text *Signals and Systems*. His present research interests are in problems involving abrupt changes in signals and systems, multidimensional estimation, decision-directed signal processing, and the asymptotic analysis of control and estimation systems.

Dr. Willsky was Program Chairman for the 17th IEEE Conference on Decision and Control, is Associate Editor of several journals including the IEEE TRANSACTIONS ON AUTOMATIC CONTROL, is a member of the Board of Governors of the IEEE Control Systems Society, and was Program Chairman for the 1981 Bilateral Seminar on Control Systems held in the People's Republic of China. He gave the opening plenary lecture at the 20th IEEE Conference on Decision and Control. He was awarded the 1979 Alfred Noble Prize by the ASCE and the 1980 Browder J. Thompson Memorial Prize Award by the IEEE for a paper excerpted from his monograph.