Chapter 6

Fault diagnosis of continuous-variable systems

This chapter provides solutions to the fault detection, isolation and estimation problems when the model of the supervised process is either a deterministic or a stochastic continuous-variable system. The chapter considers faults that can be modelled as additive signals acting on the process. The solution of these problems leads to a diagnostic system which is separated in two parts: a residual generation module and a residual evaluation module. Particular attention is paid to the link between these two parts when using stochastic models.

6.1 Introduction

Continuous-variable models (or analytical models) consist of sets of differential or difference equations. They can be deduced by application of the laws of physics, chemistry etc. to the supervised and/or controlled process. The external variables entering these equations are called inputs. One distinguishes control inputs, which are known and can be manipulated, from disturbances which cannot be manipulated. The disturbances that are not measured are called unknown inputs. Besides, imperfections in the model and measurement noise may be represented by stochastic processes (or sequences) appearing as additional inputs. When such random input is used, one speaks about stochastic models, as opposed to deterministic models. In this chapter, the design of fault detection, isolation and/or estimation systems for processes described by deterministic or stochastic continuous-variable models with unknown input will be solved. Such systems are made of two parts as already indicated in Chapter 1: a residual generation module and a residual evaluation module (or decision system) (Fig. 6.1).

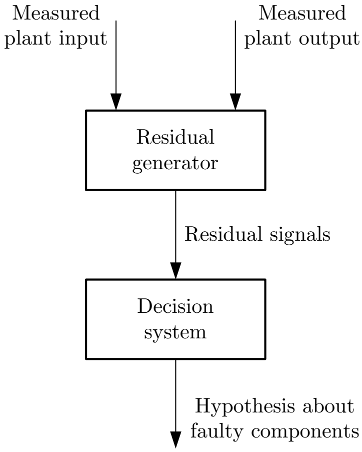


Fig. 6.1. Structure of a fault diagnosis system

The residuals are signals that, in the absence of faults, deviate from zero only due to modelling uncertainties, with nominal value being zero, or close to zero under actual working conditions. If a fault should occur, the residuals deviate from zero with a magnitude such that the new condition can be distinguished from the fault free working mode. The role of the decision system is to determine whether the residuals differ significantly from zero and, from the pattern of zero and non-zero residuals, to decide which are the most likely fault effects, and in turn, which component(s) could be the origin of a fault. When the diagnostic system is used in a fault-tolerant controller, as described in Chapters 1 and 7, details in the diagnostic task will depend on the type of faulty device and on the way the faulty condition could be treated.

Sensor faults can often be handled through estimating the faulty output signal using an estimator based on other available measurements less the one isolated as faulty. Observability of the reduced system is naturally required in this case. For this type of sensor fault, the diagnostic system needs only to perform fault detection and isolation to determine which measured signals should be disregarded. For an actuator fault which does not cause a complete loss of command, a remedial action could be to modify the control signal to the set of actuators by an increment computed in such a way that the fault is compensated. In this case, an estimate of the fault signal is needed.

The fundamental notion on which residual generation for continuous-variable systems rests is analytical redundancy. Analytical redundancy relations are equations that are deduced from an analytical model, which solely use measured variables as input. Analytical redundancy relations must be consistent in the absence of a fault, and can thus be used for residual generation. A simple example is given to introduce this notion, before considering more formal developments in subsequent sections.

Example 6.1 *Residuals for the ship autopilot*

Consider the following part of the ship autopilot example (see Section 2.2). The turn rate ω_3 and the heading angle ψ are related through

$$\dot{\psi}(t) = \omega_3(t). \tag{6.1}$$

Let us neglect the effect of waves and assume that the measurements can only be affected by a bias. Hence sensor faults are represented by additive signals and the measurement equations can be written:

$$\psi_m(t) = \psi(t) + f_\psi(t) \tag{6.2}$$
$$\omega_{3m}(t) = \omega_3(t) + f_\omega(t) \tag{6.3}$$

where the index m denotes measured quantities, and $f_\psi(t)$, $f_\omega(t)$ are the potential biases. Since most supervision systems are implemented as a software, only sampled data are available. They are linked through the following discrete model deduced from (6.1):

$$\psi(k+1) = \psi(k) + \omega_3(k)T_s, \tag{6.4}$$

where T_s stands for sampling period. By considering the equation error, r, resulting from (6.4) when the variables are substituted by their measured value, the following expression is obtained:

$$r(k) = \psi_m(k) - \psi_m(k-1) - \omega_{3m}(k-1)T_s. \tag{6.5}$$

This quantity has the properties expected for a residual. Indeed, introducing (6.2), (6.3) into (6.5) yields

$$r(k) = f_\psi(k) - f_\psi(k-1) - f_\omega(k-1)T_s.$$

This shows that, in the absence of a fault (namely when $f_\psi(k) = f_\psi(k-1) = f_\omega(k-1) = 0$), $r(k)$ is zero. Upon occurrence of a bias in the measurement of ω_3 say at time k_0, $r(k)$ takes a constant non-zero value for all $k \geq k_0$. Finally the appearance of a bias on the measurement of ψ at time instant k_0 shows up as a spike at time k_0, but has no permanent effect on r. Both faults thus affect r and this signal is zero in the absence of fault. Hence it can be named a residual signal. For decision making, it suffices to compare the residual to a specified threshold. The latter should be chosen in such a way that biases that appear to be significant for the considered application are detected.

When measurement noise is significant, comparison to a simple threshold might not be practicable, because the change in the mean of the residual due to the fault can be hidden by the effect of the noise on the residual. This noise needs to be taken into account as described in the following two discretised "noisy" versions of (6.2), (6.3):

$$\psi_m(k) = \psi(k) + f_\psi(k) + v_\psi(k) \tag{6.6}$$
$$\omega_{3m}(k) = \omega_3(k) + f_\omega(k) + v_\omega(k), \tag{6.7}$$

where $v_\psi(i), v_\omega(i), i = 1, 2, \ldots$ are mutually uncorrelated white noise sequences with variance $E(v_\psi^2(k)) = Q_\psi$ and $E(v_\omega^2(k)) = Q_{\omega_3}$ respectively.

Substituting (6.6) and (6.7) into (6.5) yields:

$$r(k) = f_\psi(k) - f_\psi(k-1) - f_\omega(k-1)T_s + v_\psi(k) - v_\psi(k-1) - v_\omega(k-1)T_s.$$

$(r(1), \ldots, r(k))$ is now a random sequence which must be evaluated by suitable algorithms. Only its mean value is equal to zero in the absence of fault. □

The difference of treatment between deterministic and stochastic models is reflected in the organisation of the chapter: Sections 6.2 to 6.5 deal with the first class of models and 6.7 to 6.8, with the second one. Analytical redundancy relations

(ARR) based on a deterministic model were already addressed in the previous chapter. Structural models were used for their determination. The link with this chapter is the object of Section 6.2 were the principle of the determination of ARR from a deterministic nonlinear state-space model is presented. Next, the particular case of deterministic linear state-space model is considered in Section 6.3, and a complete algorithm is provided for the design of parity relations (a specific type of analytical redundancy relations). A more formal presentation of parity relations for fault detection, isolation and estimation is then presented in Section 6.4 from a linear input-output model of the supervised process. The method of Sections 6.2 to 6.4 assures perfect insensitivity (or decoupling) of the residuals to an unknown input. This can only be achieved when the number of unknown input signals is lower than the number of measured output signals. When this condition does not hold, approximate decoupling of the residual with respect to the unknown input can be obtained by an optimisation approach. This is the objective of Section 6.5. A presentation of algorithms aimed at detecting changes in the mean of a stochastic random sequence is given in Section 6.7. These tools are used as parts of the systems for fault detection, fault isolation and fault estimation based on stochastic models that are described in Section 6.8.

6.2 Analytical redundancy in nonlinear deterministic systems

6.2.1 Logical background

Analytical redundancy can be seen as a tool for obtaining conditions, based on available measurements, that are necessarily fulfilled when the supervised system works in a specific operating mode. In order to illustrate the principle of analytical redundancy, consider deterministic systems described in normal operation by state and measurement equations

$$\begin{aligned} \dot{\boldsymbol{x}}(t) &= \boldsymbol{g}(\boldsymbol{x}(t), \boldsymbol{u}(t), \boldsymbol{d}(t), \theta, t) \qquad (6.8) \\ \boldsymbol{y}(t) &= \boldsymbol{h}(\boldsymbol{x}(t), \boldsymbol{u}(t), \boldsymbol{d}(t), \theta, t), \qquad (6.9) \end{aligned}$$

where $\boldsymbol{x} \in \mathrm{I\!R}^n$ is the state vector, which is not available, $\boldsymbol{u} \in \mathrm{I\!R}^m$ is the control input vector, $\boldsymbol{d} \in \mathrm{I\!R}^{n_d}$ is an uncontrolled deterministic vector (disturbance). θ is a parameter vector which is considered to be known, and $\boldsymbol{y} \in \mathrm{I\!R}^p$ is the measurement vector. Let $\mathcal{H}_0$ be the situation corresponding to normal operation, and $\mathcal{H}_1 = \neg\mathcal{H}_0$ some faulty situation. The following logical statements are true

$$\begin{aligned} \mathcal{H}_0 &\iff [\dot{\boldsymbol{x}}(t) = \boldsymbol{g}(\boldsymbol{x}(t), \boldsymbol{u}(t), \boldsymbol{d}(t), \theta, t)] \wedge [\boldsymbol{y}(t) = \boldsymbol{h}(\boldsymbol{x}(t), \boldsymbol{u}(t), \boldsymbol{d}(t), \theta, t)] \\ \mathcal{H}_1 &\iff [\dot{\boldsymbol{x}}(t) \neq \boldsymbol{g}(\boldsymbol{x}(t), \boldsymbol{u}(t), \boldsymbol{d}(t), \theta(t))] \vee [\boldsymbol{y}(t) \neq \boldsymbol{h}(\boldsymbol{x}(t), \boldsymbol{u}(t), \boldsymbol{d}(t), \theta(t))] \,. \end{aligned}$$

The violation of equality constraints that results from faults may be described in two ways:

- In the first option, faults are assumed to result from parametric variations, which is represented as

$$\theta_f(t) \neq \theta \iff \theta_f(t) = \theta + \boldsymbol{f}(t), \boldsymbol{f}(t) \neq 0,$$

where $\theta_f(t)$ stands for the parameter vector associated with the faulty system.

- In the second option, no hypothesis is made about the origin of the discrepancy, which is just represented as an additive vector

$$\begin{aligned}
&[\dot{\boldsymbol{x}}(t) \neq \boldsymbol{g}(\boldsymbol{x}(t), \boldsymbol{u}(t), \boldsymbol{d}(t), \theta(t))] \vee [\boldsymbol{y}(t) \neq \boldsymbol{h}(\boldsymbol{x}(t), \boldsymbol{u}(t), \boldsymbol{d}(t), \theta(t))] \\
&\iff \\
&\exists \left(\boldsymbol{f}_x, \boldsymbol{f}_y\right) \neq (0,0) : \\
&\qquad [\dot{\boldsymbol{x}}(t) = \boldsymbol{g}(\boldsymbol{x}(t), \boldsymbol{u}(t), \boldsymbol{d}(t), \theta(t)) + \boldsymbol{f}_x(t)] \\
&\qquad \vee \left[\boldsymbol{y}(t) = \boldsymbol{h}(\boldsymbol{x}(t), \boldsymbol{u}(t), \boldsymbol{d}(t), \theta(t)) + \boldsymbol{f}_y(t)\right].
\end{aligned}$$

In both cases, the normal and the faulty system are represented using some "fault vector" $\boldsymbol{f}(t)$ where normal operation is associated with $\boldsymbol{f}(t) = 0$. Most often, the preliminary analysis of the system has identified a set of faults that are likely to occur, and that the FDI system to be designed should detect, isolate and estimate. When such knowledge is available, it results in the logical statement

$$i \in I : \mathcal{H}_i \iff \boldsymbol{f}(t) = \boldsymbol{f}_i(\eta_i, t) \neq 0,$$

where $\mathcal{H}_i$ denotes the i^{th} fault situation, $I = \{1, 2, \cdots, n_f\}$ where n_f is the number of possible fault modes, and the knowledge available about each fault is modelled by the possible time evolution of the vector $\boldsymbol{f}$ which depends on some unknown parameters η_i (fault estimation therefore directly refers to the estimation of these parameters).

6.2.2 Analytical redundancy relations with no unknown inputs

Introducing the fault vector $\boldsymbol{f}(t)$ in the state and measurement equations, and setting $\boldsymbol{d}(t) = 0$, for all t one gets [1]

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{g}(\boldsymbol{x}(t), \boldsymbol{u}(t), \boldsymbol{f}(t)) \qquad \boldsymbol{y}(t) = \boldsymbol{h}(\boldsymbol{x}(t), \boldsymbol{u}(t), \boldsymbol{f}(t)), \tag{6.10}$$

where, since θ is known, the dependency of the state and measurement equations on the parameter is no longer made explicit, and time invariant systems are considered in order to shorten the notations. It turns out that from (6.10), it is possible to construct *residuals*, i.e. quantities which can be computed in real time from the available data, and whose behaviour is different under the different situations $\mathcal{H}_0$ and $\mathcal{H}_1$. Such residuals are obtained from a two step construction:

Step 1: Derivation of the outputs. Assuming that all functions are differentiable with respect to their arguments, it is possible to construct the derivative $\dot{\boldsymbol{y}}(t)$ of the output signal $\boldsymbol{y}(t)$:

$$\dot{\boldsymbol{y}}(t) = \frac{\partial \boldsymbol{h}}{\partial \boldsymbol{x}}(\cdot)\dot{\boldsymbol{x}}(t) + \frac{\partial \boldsymbol{h}}{\partial \boldsymbol{u}}(\cdot)\dot{\boldsymbol{u}}(t) + \frac{\partial \boldsymbol{h}}{\partial \boldsymbol{f}}(\cdot)\dot{\boldsymbol{f}}(t)$$

[1] The same symbols g and h as in (6.8), (6.9) are used by an abuse of notation

Replacing $\dot{\boldsymbol{x}}(t)$ by its value, one gets

$$\begin{aligned}\dot{\boldsymbol{y}}(t) &= \frac{\partial \boldsymbol{h}}{\partial \boldsymbol{x}}(\cdot)\boldsymbol{g}(\boldsymbol{x}(t),\boldsymbol{u}(t),\boldsymbol{f}(t)) + \frac{\partial \boldsymbol{h}}{\partial \boldsymbol{u}}(\cdot)\dot{\boldsymbol{u}}(t) + \frac{\partial \boldsymbol{h}}{\partial \boldsymbol{f}}(\cdot)\dot{\boldsymbol{f}}(t) \\ &:= \boldsymbol{h}_1(\boldsymbol{x}(t),\bar{\boldsymbol{u}}^{(1)}(t),\bar{\boldsymbol{f}}^{(1)}(t)),\end{aligned}$$

where $\bar{\boldsymbol{u}}^{(1)}(t)$ is a short notation for $\left(\boldsymbol{u}',\dot{\boldsymbol{u}}'(t)\right)'$. Iterating this process until some order of derivation q (to be determined later), and assuming the existence of all required derivatives, one obtains

$$\bar{\boldsymbol{y}}^{(q)}(t) = H^q\left(\boldsymbol{x}(t),\bar{\boldsymbol{u}}^{(q)}(t),\bar{\boldsymbol{f}}^{(q)}(t)\right) \tag{6.11}$$

which is a set of $(q+1)\,p$ equations - or constraints - (the dimension of $\bar{\boldsymbol{y}}^{(q)}(t)$), where the different variables have the following dimensions: $\boldsymbol{x}\in \mathrm{IR}^n, \bar{\boldsymbol{u}}^{(q)}(t)\in \mathrm{IR}^{(q+1)\times m}, \bar{\boldsymbol{f}}^{(q)}(t)\in \mathrm{IR}^{(q+1)\times n_f}$. The known variables are $\bar{\boldsymbol{y}}^{(q)}$ and $\bar{\boldsymbol{u}}^{(q)}$ while the unknown variables are $\boldsymbol{x}$. $\bar{\boldsymbol{f}}^{(q)}(t)$ has a particular status, since it is known (equal to zero) when $\mathcal{H}_0$ is true, while it is unknown when $\mathcal{H}_1$ is true.

Example 6.2 *Redundancy in a nonlinear system*

The variable t is omitted below. Applying the above procedure with $s=2$ to the system

$$\begin{aligned}\begin{pmatrix}\dot{x}_1\\ \dot{x}_2\end{pmatrix} &= \begin{pmatrix}-x_1+x_2^2+u+f_1\\ -2x_2+f_2\end{pmatrix}\\ y &= x_1+f_3\end{aligned}$$

gives

$$\begin{aligned}\dot{y} &= -x_1+x_2^2+u+f_1+\dot{f}_3\\ \ddot{y} &= x_1-5x_2^2-u-f_1+2x_2f_2+\dot{u}+\dot{f}_1+\ddot{f}_3.\end{aligned}$$

(6.11) is thus a system of three equations

$$\bar{y}^{(2)} = \begin{pmatrix}y\\ \dot{y}\\ \ddot{y}\end{pmatrix} = \begin{pmatrix}x_1+f_3\\ -x_1+x_2^2+u+f_1+\dot{f}_3\\ x_1-5x_2^2-u-f_1+2x_2f_2+\dot{u}+\dot{f}_1+\ddot{f}_3\end{pmatrix}. \square \tag{6.12}$$

Step 2: Elimination of the state. Assume that $(q+1)\,p > n$ and the Jacobian $\frac{\partial H^q(\cdot)}{\partial \boldsymbol{x}}$ is of rank n. Note that the first condition gives a lower bound on the order of derivation that is necessary in establishing (6.11). It follows that (6.11) can be decomposed into

$$\begin{pmatrix}\bar{\boldsymbol{y}}_m^{(q)}(t)\\ \bar{\boldsymbol{y}}_{nm}^{(q)}(t)\end{pmatrix} - \begin{pmatrix}H_m^q\left(\boldsymbol{x}(t),\bar{\boldsymbol{u}}^{(q)}(t),\bar{\boldsymbol{f}}^{(q)}(t)\right)\\ H_{nm}^q\left(\boldsymbol{x}(t),\bar{\boldsymbol{u}}^{(q)}(t),\bar{\boldsymbol{f}}^{(q)}(t)\right)\end{pmatrix} = 0 \tag{6.13}$$

where the first subsystem is of dimension n and allows to compute $x(t)$ (at least locally) as a function of the other variables

$$\boldsymbol{x}(t) = \phi(\bar{\boldsymbol{y}}_m^{(q)}(t), \bar{\boldsymbol{u}}^{(q)}(t), \bar{\boldsymbol{f}}^{(q)}(t))$$

(this results from the implicit function theorem). Replacing $\boldsymbol{x}(t)$ by its value in the second subsystem, which is of dimension $(q+1)\,p-n$, one obtains a system that is equivalent to (6.11)

$$\boldsymbol{x}(t) = \phi(\bar{\boldsymbol{y}}_m^{(q)}(t), \bar{\boldsymbol{u}}^{(q)}(t), \bar{\boldsymbol{f}}^{(q)}(t)) \tag{6.14}$$

$$0 = \boldsymbol{r}(\bar{\boldsymbol{y}}^{(q)}(t), \bar{\boldsymbol{u}}^{(q)}(t), \bar{\boldsymbol{f}}^{(q)}(t)), \tag{6.15}$$

where

$$\begin{aligned} &\boldsymbol{r}(\bar{\boldsymbol{y}}^{(q)}(t), \bar{\boldsymbol{u}}^{(q)}(t), \bar{\boldsymbol{f}}^{(q)}(t)) \\ &= \bar{\boldsymbol{y}}_{nm}^{(q)}(t) - H_{nm}^{q}(\phi(\bar{\boldsymbol{y}}_m^{q}(t)), \bar{\boldsymbol{u}}^{(q)}(t), \bar{\boldsymbol{f}}^{(q)}(t)), \bar{\boldsymbol{u}}^{(q)}(t), \bar{\boldsymbol{f}}^{(q)}(t)). \end{aligned}$$

The set of constraints (6.15) is seen to contain only inputs, outputs and fault signals (along with their derivatives). It is called an analytical redundancy relations (ARR) associated with the pair $(\boldsymbol{g}, \boldsymbol{h})$ and $\boldsymbol{r}(\bar{\boldsymbol{y}}^{(q)}, \bar{\boldsymbol{u}}^{(q)}, \bar{\boldsymbol{f}}^{(q)})$ is called the residual vector.

Remark 6.1 *Link to structural approach*
A structural condition for ARR to exist is that (6.11) is overconstrained with respect to the unknowns $\boldsymbol{x}(t)$ i.e. there is a matching which is complete with respect to $\boldsymbol{x}(t)$. Decomposing the set of constraints (6.11) into matched (index m) and non-matched ones (index nm) yields (6.13), where the matched subsystem has n constraints while the non-matched subsystem has $(q+1)\,p-n$ constraints. From the interpretation of matchings in the previous chapter, $\boldsymbol{x}(t)$ is computed in the matched subsystem, as a function of the other variables $\phi\left(\bar{\boldsymbol{y}}_m^{(q)}(t), \bar{\boldsymbol{u}}^{(q)}(t), \bar{\boldsymbol{f}}^{(q)}(t)\right)$ and replacing $\boldsymbol{x}(t)$ by its value in the non-matched subsystem gives the redundancy relations. □

Example 6.2 (cont.) *Redundancy in a nonlinear system*

Step 2 is now applied to (6.12). The state $(x_1, x_2)'$ can be computed from the first two equations of (6.12) leading to the equivalent system

$$\begin{aligned} x_1 &= y - f_3 \\ x_2 &= \pm\sqrt{\dot{y} + y - f_3 - u - f_1 - \dot{f}_3} \\ 0 &= \ddot{y} - y + f_3 + 5\left(\dot{y} + y - f_3 - u - f_1 - \dot{f}_3\right) + u + ... \\ &\quad ... + f_1 + 2\left(\sqrt{\dot{y} + y - f_3 - u - f_1 - \dot{f}_3}\right) f_2 - \dot{u} - \dot{f}_1 - \ddot{f}_3, \end{aligned} \tag{6.16}$$

where the third equation is seen to depend only on the available inputs and outputs and on the faults. □

6.2.3 Unknown inputs, exact decoupling

When unknown inputs are present, a state-space model of the system takes the form[2]

$$\begin{aligned} \dot{\boldsymbol{x}}(t) &= \boldsymbol{g}(\boldsymbol{x}(t), \boldsymbol{u}(t), \boldsymbol{d}(t), \boldsymbol{f}(t)) \\ \boldsymbol{y}(t) &= \boldsymbol{h}(\boldsymbol{x}(t), \boldsymbol{u}(t), \boldsymbol{d}(t), \boldsymbol{f}(t)). \end{aligned} \tag{6.17}$$

Applying the same technique as above leads to

$$\bar{\boldsymbol{y}}^{(q)}(t) = H^q \left(\boldsymbol{x}(t), \bar{\boldsymbol{u}}^{(q)}(t), \bar{\boldsymbol{d}}^{(q)}(t), \bar{\boldsymbol{f}}^{(q)}(t) \right). \tag{6.18}$$

Under the condition that $(q+1)\,p > n + (q+1)\,n_d$ and the Jacobian

$$\left[\begin{array}{cc} \frac{\partial H^q(\cdot)}{\partial \boldsymbol{x}} & \frac{\partial H^q(\cdot)}{\partial \bar{\boldsymbol{d}}^{(q)}} \end{array} \right]$$

is of rank $n + (q+1)\,n_d$ both the state and the unknown inputs can be eliminated, leading to the equivalent system

$$\left(\begin{array}{c} \boldsymbol{x}(t) \\ \bar{\boldsymbol{d}}^{(q)}(t) \end{array} \right) = \left(\begin{array}{c} \phi_x(\bar{\boldsymbol{y}}_m^{(q)}(t), \bar{\boldsymbol{u}}^{(q)}(t), \bar{\boldsymbol{f}}^{(q)}(t)) \\ \phi_d(\bar{\boldsymbol{y}}_m^{(q)}(t), \bar{\boldsymbol{u}}^{(q)}(t), \bar{\boldsymbol{f}}^{(q)}(t)) \end{array} \right) \tag{6.19}$$

$$0 = \boldsymbol{r}(\bar{\boldsymbol{y}}^{(q)}(t), \bar{\boldsymbol{u}}^{(q)}(t), \bar{\boldsymbol{f}}^{(q)}(t)), \tag{6.20}$$

where (6.20) are the analytical redundancy relations, which are independent of the unknown inputs, hence the name "exact decoupling" which is given to this approach. Note that exact decoupling is possible only if the structural graph of system (6.18) is overconstrained with respect to both the unknowns $\boldsymbol{x}$ and $\bar{\boldsymbol{d}}^{(q)}$.

6.2.4 How to find analytical redundancy relations

There are several procedures by which ARR can be found. They all rest on the elimination of $\boldsymbol{x}(t)$ (and $\bar{\boldsymbol{d}}^{(q)}(t)$ when unknown inputs are present), either by starting with (6.8) and (6.9) or by establishing first (6.11).

Elimination procedures fit the nature of the functions $\boldsymbol{g}$ and $\boldsymbol{h}$. When all functions are linear, projection approaches are well suited: this is the parity space approach which will be described in Section 6.3. Most often, nonlinear models involve polynomial functions (because polynomials can approximate any smooth nonlinear function). There are, basically, three elimination techniques for polynomial functions. All three require the components of the state to be eliminated according to some selected order. *Elimination theory* rests on Euclidean division and derivation. *Gröbner bases* uses Euclidean division and the computation of so called S-polynomials. *Characteristic sets* (also called Ritt's algorithm) rest on Euclidean division and derivation. The state is directly eliminated from the system (6.8) (6.9), and ARR with minimum derivative order can be obtained.

[2] Again the same functions g and h as above are used by an abuse of notation.

6.2.5 ARR-based diagnosis

Fault detection. In the absence of unknown inputs, or when exact decoupling is possible, the following logical statements hold

$$\begin{array}{llll} (6.10) & \Longleftrightarrow & (6.14),(6.15) & \Rightarrow \boldsymbol{r}(\bar{\boldsymbol{y}}^{(q)}(t),\bar{\boldsymbol{u}}^{(q)}(t),\bar{\boldsymbol{f}}^{(q)}(t))=0 \\ (6.17) & \Longleftrightarrow & (6.19) & \Rightarrow \boldsymbol{r}(\bar{\boldsymbol{y}}^{(q)}(t),\bar{\boldsymbol{u}}^{(q)}(t),\bar{\boldsymbol{f}}^{(q)}(t))=0 \end{array} \tag{6.21}$$

From (6.21) it follows that in both cases necessary conditions for normal system operation are given by

$$\mathcal{H}_0 \Rightarrow \boldsymbol{r}(\bar{\boldsymbol{y}}^{(q)}(t),\bar{\boldsymbol{u}}^{(q)}(t),0)=0$$

Therefore, fault detection immediately follows from

$$\boldsymbol{r}(\bar{\boldsymbol{y}}^{(q)}(t),\bar{\boldsymbol{u}}^{(q)}(t),0)\neq 0 \quad \Rightarrow \mathcal{H}_1.$$

Remark 6.2 *Non detectable faults*
Note that $\boldsymbol{r}(\bar{\boldsymbol{y}}^{(q)}(t),\bar{\boldsymbol{u}}^{(q)}(t),0)=0$ does not imply $\mathcal{H}_0$ since the condition expressed by the analytical redundancy relation is only necessary. In fact, $\boldsymbol{r}(\bar{\boldsymbol{y}}^{(q)}(t),\bar{\boldsymbol{u}}^{(q)}(t),0)=0$ is to be read: $\mathcal{H}_0$ is not falsified by the observations, or in other terms "it is not impossible that the system is healthy". In fact, special fault values that are not detectable through analytical redundancy could exist. They correspond to nonzero values of $\boldsymbol{f}(t)$ that yield $\boldsymbol{r}(\bar{\boldsymbol{y}}^{(q)}(t),\bar{\boldsymbol{u}}^{(q)}(t),\bar{\boldsymbol{f}}^{(q)}(t))=0$. □

Example 6.2 (cont.) *Redundancy in a nonlinear system*

The redundancy relation in (6.16) writes

$$\begin{aligned} &\ddot{y}+5\dot{y}+4y-4u-\dot{u} \\ &= \quad f_1-2\left(\sqrt{\dot{y}+y-f_3-u-f_1-\dot{f}_3}\right)f_2+4f_3+\dot{f}_1+5\dot{f}_3+\ddot{f}_3. \end{aligned}$$

Therefore, the residual is

$$r\left(\bar{y}^{(2)},\bar{u}^{(2)},0\right)=\ddot{y}+5\dot{y}+4y-4u-\dot{u}$$

and the fault detection rule is

$$\ddot{y}+5\dot{y}+4y-4u-\dot{u}\neq 0 \Rightarrow \mathcal{H}_1. \ \square$$

Fault isolation. Fault isolation is approached in a similar way, by the design of so-called structured residuals. Assume it is possible to separate the set of faults I into two subsets I_1 and I_2 such that $I=I_1\cup I_2$. Set

$$\boldsymbol{f}(t)=\left(\ \boldsymbol{f}_{I_1}(t)'\boldsymbol{f}_{I_2}(t)'\ \right)',$$

where only $\boldsymbol{f}_{I_1}(t)$ ($\boldsymbol{f}_{I_2}(t)$) is nonzero upon occurrence of a fault in I_1 (I_2). If the set of residuals can also be separated in two subsets

$$\boldsymbol{r}(\bar{y}^{(s)}(t), \bar{u}^{(s)}(t), \bar{f}^{(s)}(t)) = \begin{pmatrix} \boldsymbol{r}_1(\bar{y}^{(s)}(t), \bar{u}^{(s)}(t), \bar{f}^{(s)}(t)) \\ \boldsymbol{r}_2(\bar{y}^{(s)}(t), \bar{u}^{(s)}(t), \bar{f}^{(s)}(t)) \end{pmatrix}, \tag{6.22}$$

so that (a) $\boldsymbol{r}_1$ is insensitive to faults in I_2 but sensitive to faults in I_1 while (b) $\boldsymbol{r}_2$ is insensitive to faults in I_1 but sensitive to faults in I_2, then, as shown below, it is possible to distinguish between the occurrence of a fault from the class I_1 or I_2. The logical expressions corresponding to these assumptions are

$$(a) \quad \left.\begin{array}{l} \forall i \in I_1 : \boldsymbol{f}_{I_1}(t) = \boldsymbol{f}_i(\eta_i, t) = 0 \\ \exists i \in I_2 : \boldsymbol{f}_{I_2}(t) = \boldsymbol{f}_i(\eta_i, t) \neq 0 \end{array}\right\} \Rightarrow \begin{array}{l} \boldsymbol{r}_1(\bar{y}^{(s)}(t), \bar{u}^{(s)}(t), \bar{f}^{(s)}(t)) = 0 \\ \boldsymbol{r}_2(\bar{y}^{(s)}(t), \bar{u}^{(s)}(t), \bar{f}^{(s)}(t)) \neq 0 \end{array}$$

$$(b) \quad \left.\begin{array}{l} \exists i \in I_1 : \boldsymbol{f}_{I_1}(t) = \boldsymbol{f}_i(\eta_i, t) \neq 0 \\ \forall i \in I_2 : \boldsymbol{f}_{I_2}(t) = \boldsymbol{f}_i(\eta_i, t) = 0 \end{array}\right\} \Rightarrow \begin{array}{l} \boldsymbol{r}_1(\bar{y}^{(s)}(t), \bar{u}^{(s)}(t), \bar{f}^{(s)}(t)) \neq 0 \\ \boldsymbol{r}_2(\bar{y}^{(s)}(t), \bar{u}^{(s)}(t), \bar{f}^{(s)}(t)) = 0. \end{array}$$

There are four possible situations (logical 0 means $\boldsymbol{r} = 0$ while logical 1 means $\boldsymbol{r} \neq 0$) and the following conclusions are true.

$\boldsymbol{r}_1$	$\boldsymbol{r}_2$	Conclusion
0	0	$\mathcal{H}_0$ is not falsified (no fault is detected)
0	1	$\mathcal{H}_0$ is falsified by a fault $i \in I_2$
1	0	$\mathcal{H}_0$ is falsified by a fault $i \in I_1$
1	1	$\mathcal{H}_0$ is falsified by a fault $i \in I_1$ and a fault $j \in I_2$

Therefore, under (6.22), it is possible to isolate a fault within the subset I_1 or within the subset I_2. By designing several partitions of the set of faults into two classes it is obviously possible to isolate faults within smaller subsets that result from the intersections of all these partitions.

Remark 6.3 *Non isolable faults*

Only a limited number of partitions into two classes enjoying property (6.22) can be obtained for a given system. Therefore, it may happen that whatever the partition such that (6.22) holds, two given faults, say i and j are always in the same class. These faults always have the same effect on the analytical redundancy relations, and therefore they are not isolable from each other, which means that every FDI conclusion will contain "the fault is either i or j (or both)". □

Example 6.3 *Two-tank system*

In Example 5.40, the set of constraints associated with the two-tank system components wrote

Pump:	$q_P = u.f(h_1)$
Tank 1:	$\dot{h}_1 = \frac{1}{A}(q_P - q_L - q_{12})$
Tank 2:	$\dot{h}_2 = \frac{1}{A}(q_{12} - q_2)$
Pipe between tanks ($h_1 > h_2$):	$q_{12} = k_1\sqrt{h_1 - h_2}$
Output pipe:	$q_2 = k_2\sqrt{h_2}$
Outflow measurement:	$q_m = k_m.q_2.$

The state-space equations are

$$\begin{pmatrix} \dot{h}_1 \\ \dot{h}_2 \end{pmatrix} = \begin{pmatrix} -\frac{k_1}{A}\sqrt{h_1 - h_2} + \frac{f(h_1)}{A}.u - \frac{1}{A}.q_L \\ \frac{k_1}{A}\sqrt{h_1 - h_2} - \frac{k_2}{A}\sqrt{h_2} \end{pmatrix} \tag{6.23}$$

and the measurement equation is

$$q_m = k_m k_2 \sqrt{h_2}. \tag{6.24}$$

Derivating once the output gives

$$\begin{aligned} \dot{q}_m &= k_m k_2 \left(h_2\right)^{-1/2} \dot{h}_2 \\ \dot{q}_m &= k_m k_2 \left(h_2\right)^{-1/2} \left(\frac{k_1}{A}\sqrt{h_1 - h_2} - \frac{k_2}{A}\sqrt{h_2}\right). \end{aligned} \tag{6.25}$$

From (6.24) and (6.25) the two states h_1 and h_2 can be computed

$$\begin{aligned} h_2 &= \left(\frac{q_m}{k_m k_2}\right)^2 \\ h_1 &= q_m^2 \left(1 + (1 + \dot{q}_m)^2\right). \end{aligned} \tag{6.26}$$

Derivating once again gives

$$\ddot{q}_m = \frac{\left(h_1 - h_2\right)^{-1/2} \sqrt{h_2} \left(\dot{h}_1 - \dot{h}_2\right) - \left(h_2\right)^{-1/2} \dot{h}_2 \left(h_1 - h_2\right)^{1/2}}{h_2},$$

where replacing $h_1, h_2, \dot{h}_1, \dot{h}_2$ by their values taken from (6.26), (6.23) and (6.24) – (6.25) gives the redundancy relation

$$\begin{aligned} & r(q_m, \dot{q}_m, \ddot{q}_m, u, q_L) \\ &= \sqrt{h_2} \left(h_1 - h_2\right)^{1/2} \ddot{q}_m - \dot{h}_1 + \dot{h}_2 + \left(h_2\right)^{-1} \dot{h}_2 \left(h_1 - h_2\right) = 0 \end{aligned} \tag{6.27}$$

and the leakage detection rule

$$r(q_m, \dot{q}_m, \ddot{q}_m, u, 0) \neq 0 \Rightarrow q_L \neq 0. \square$$

6.3 Analytical redundancy relations for linear deterministic systems – time domain

Let us consider the following continuous-time state-space model

$$\begin{aligned} \dot{\boldsymbol{x}}(t) &= \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{B}\boldsymbol{u}(t) + \boldsymbol{E}_x \boldsymbol{d}(t) + \boldsymbol{F}_x \boldsymbol{f}(t), \quad \boldsymbol{x}(0) = \boldsymbol{x}_0, \\ \boldsymbol{y}(t) &= \boldsymbol{C}\boldsymbol{x}(t) + \boldsymbol{D}\boldsymbol{u}(t) + \boldsymbol{E}_y \boldsymbol{d}(t) + \boldsymbol{F}_y \boldsymbol{f}(t), \end{aligned} \tag{6.28}$$

where $\boldsymbol{x} \in \mathbb{R}^n$ denotes the state vector, $\boldsymbol{u} \in \mathbb{R}^m$, is the vector of measured input signals, $\boldsymbol{y} \in \mathbb{R}^p$ is the vector of measured plant output signals, $\boldsymbol{d} \in \mathbb{R}^{n_d}$ and $\boldsymbol{f} \in \mathbb{R}^{n_f}$ are vectors of unknown input signals. $\boldsymbol{f}$ represents the faults one wishes to detect, while $\boldsymbol{d}$ are unknown disturbances that should not be detected.

The aim is to solve the following problem.

Problem 6.1 (Design of linear analytical redundancy relations)
Given a model of the supervised process of the form (6.28), determine, if possible, a set of linear relations between the measured inputs and outputs and their derivatives up to a certain order, say q, such that,

- *in the absence of fault,*

$$\sum_{i=1}^{q} \boldsymbol{W}_{y,i}\boldsymbol{y}^{(i)}(t) + \sum_{i=1}^{q} \boldsymbol{W}_{u,i}\boldsymbol{u}^{(i)}(t) = 0,$$

  where $\boldsymbol{z}^{(i)}(t)$ denotes the i^{th} derivative of $\boldsymbol{z}(t)$ and $\boldsymbol{W}_{y,i}$, $\boldsymbol{W}_{u,i}$ are $n_r \times p$ and $n_r \times m$ matrices of real elements, n_r being the number of relations (to be determined),
- *in the presence of a fault,*

$$\sum_{i=1}^{q} \boldsymbol{W}_{y,i}\boldsymbol{y}^{(i)}(t) + \sum_{i=1}^{q} \boldsymbol{W}_{u,i}\boldsymbol{u}^{(i)}(t) \neq 0.$$

  Such relations are a particular kind of analytical redundancy relations called parity relations.

In order to solve this problem, let us consider the successive time derivatives of $\boldsymbol{y}$ up to order q:

$$\begin{aligned} \boldsymbol{y}(t) &= \boldsymbol{C}\boldsymbol{x}(t) + \boldsymbol{D}\boldsymbol{u}(t) + \boldsymbol{E}_y\boldsymbol{d}(t) + \boldsymbol{F}_y\boldsymbol{f}(t) \\ \dot{\boldsymbol{y}}(t) &= \boldsymbol{C}\dot{\boldsymbol{x}}(t) \\ &= \boldsymbol{C}\boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{C}\boldsymbol{B}\boldsymbol{u}(t) + D\dot{\boldsymbol{u}}(t) + \boldsymbol{C}\boldsymbol{E}_x\,\boldsymbol{d}(t) \\ &\quad + \boldsymbol{E}_y\;\dot{\boldsymbol{d}}(t) + \boldsymbol{C}\boldsymbol{F}_x\,\boldsymbol{f}(t) + \boldsymbol{F}_y\,\dot{\boldsymbol{f}}(t), \end{aligned} \tag{6.29}$$

where the last equality is deduced by substitution of (6.28) for $\dot{\boldsymbol{x}}(t)$. By iterating this process, the following expression for the q^{th} derivative of $\boldsymbol{y}$ is obtained:

$$\begin{aligned} \boldsymbol{y}^{(q)}(t) \;=\; & \boldsymbol{C}\boldsymbol{A}^q\boldsymbol{x}(t) + \boldsymbol{C}\boldsymbol{A}^{(q-1)}\boldsymbol{B}\boldsymbol{u}(t) + \ldots + \boldsymbol{C}\boldsymbol{B}\boldsymbol{u}^{(q-1)}(t) + \boldsymbol{D}\boldsymbol{u}^{(q)}(t) + \\ & + \boldsymbol{C}\boldsymbol{A}^{(q-1)}\boldsymbol{E}_x\boldsymbol{d}(t) + \ldots + \boldsymbol{C}\boldsymbol{E}_x\,\boldsymbol{d}^{(q-1)}(t) + \boldsymbol{E}_y\,\boldsymbol{d}^{(q)}(t) + \\ & + \boldsymbol{C}\boldsymbol{A}^{(q-1)}\boldsymbol{F}_x\boldsymbol{f}(t) + \ldots + \boldsymbol{C}\boldsymbol{F}_x\,\boldsymbol{f}^{(q-1)}(t) + \boldsymbol{F}_y\,\boldsymbol{f}^{(q)}(t). \end{aligned} \tag{6.30}$$

The above set of equations can be concatenated into the expression

$$\bar{\boldsymbol{y}}^{(q)}(t) = \mathcal{O}\boldsymbol{x}(t) + \boldsymbol{T}_{u,q}\bar{\boldsymbol{u}}^{(q)}(t) + \boldsymbol{T}_{d,q}\bar{\boldsymbol{d}}^{(q)}(t) + \boldsymbol{T}_{f,q}\bar{\boldsymbol{f}}^{(q)}, \tag{6.31}$$

where $\bar{y}^{(q)}(t) = \left(\boldsymbol{y}(t)'\dot{\boldsymbol{y}}(t)' \ldots \boldsymbol{y}^{(q)}(t)'\right)'$, and $\bar{\boldsymbol{u}}^{(q)}(t)$, $\bar{\boldsymbol{d}}^{(q)}(t)$, $\bar{\boldsymbol{f}}^{(q)}(t)$ have a similar form with $\boldsymbol{u}(t)$, $\boldsymbol{d}(t)$ and $\boldsymbol{f}(t)$ substituted for $\boldsymbol{y}(t)$,

$$\mathcal{O} = \begin{pmatrix} \boldsymbol{C} \\ \boldsymbol{C}\boldsymbol{A} \\ \vdots \\ \boldsymbol{C}\boldsymbol{A}^q \end{pmatrix}, \qquad \boldsymbol{T}_{u,q} = \begin{pmatrix} \boldsymbol{D} & 0 & 0 & \ldots & 0 \\ \boldsymbol{C}\boldsymbol{B} & \boldsymbol{D} & 0 & \ldots & 0 \\ \vdots & \ddots & \ddots & & \\ \boldsymbol{C}\boldsymbol{A}^{q-1}\boldsymbol{B} & \ldots & \ldots & \boldsymbol{C}\boldsymbol{B} & \boldsymbol{D} \end{pmatrix}$$

and a similar definition holds for the block Toeplitz matrices $\boldsymbol{T}_{d,q}, \boldsymbol{T}_{f,q}$ with respectively $\boldsymbol{E}_y$ and $\boldsymbol{E}_x$ or $\boldsymbol{F}_y$ and $\boldsymbol{F}_x$ substituted for $\boldsymbol{D}$ and $\boldsymbol{B}$.

If there exists a value of q such that

$$\text{rank}\begin{pmatrix} \mathcal{O} & \boldsymbol{T}_{d,q} \end{pmatrix} < (q+1)p,$$

the left nullspace of $(\mathcal{O}\ \boldsymbol{T}_{d,q})$ is not empty. The dimension of this subspace, say n_r, is given as $n_r = (q+1)p - \text{rank}\begin{pmatrix} \mathcal{O} & \boldsymbol{T}_{d,q} \end{pmatrix}$. Let $\boldsymbol{W}$ be a $n_r \times (q+1)p$ matrix of which each row is a basis vector for this subspace. Multiplying (6.31) on the left by $\boldsymbol{W}$ results in the following equality

$$\boldsymbol{W}\bar{\boldsymbol{y}}^{(q)}(t) - \boldsymbol{W}\boldsymbol{T}_{u,q}\bar{\boldsymbol{u}}^{(q)}(t) = \boldsymbol{W}\boldsymbol{T}_{f,q}\bar{\boldsymbol{f}}^{(q)}(t), \tag{6.32}$$

since $\boldsymbol{W}$ has been specifically computed to eliminate the terms in $\boldsymbol{x}(t)$ and $\bar{\boldsymbol{d}}^{(q)}(t)$. Equation (6.32) describes n_r analytical redundancy relations. Indeed, in the absence of fault, the right hand side is equal to zero, and it is normally different from zero in the presence of a fault.

In order to implement such relations, and thus to compute the quantity

$$\boldsymbol{r}(t) = \boldsymbol{W}\bar{\boldsymbol{y}}^{(q)}(t) - \boldsymbol{W}\boldsymbol{T}_{u,q}\bar{\boldsymbol{u}}^{(q)}(t), \tag{6.33}$$

it is necessary to evaluate the derivatives that appear in the above relation. Such signals are highly sensitive to noise, so that filtered estimates of the derivatives have to be used. One approach is to resort to a so-called state variable filter, which amounts to implementing the scheme of Fig. 6.2. Such a filter is used for each component of $\boldsymbol{y}(t)$ and $\boldsymbol{u}(t)$. Letting $z(t)$ denote the input of such a filter, the i^{th} integrator output provides the i^{th} filtered derivative of z, $z_f^{(i)}$. This filter corresponds to the analog simulation of the observability canonical state-space representation for the relation

$$z_f(s) = \frac{1}{s^q + a_1 s^{(q-1)} + \ldots + a_q} z(s).$$

By taking this filter into account, (6.33) can be rewritten in the frequency domain as

$$\boldsymbol{r}_f(s) = (\boldsymbol{W}_y(s)\boldsymbol{y}(s) + \boldsymbol{W}_u(s)\boldsymbol{u}(s))/p_f(s), \tag{6.34}$$

where

$$\boldsymbol{W}_y(s) = \sum_{i=0}^{q} \boldsymbol{W}_i s^i$$

with $\boldsymbol{W}_i$, the matrix made of columns $i\,p+1$ to $(i+1)p$ of $\boldsymbol{W}$, $\boldsymbol{W}_u(s)$ is defined similarly with $\boldsymbol{W}\boldsymbol{T}_{u,q}$ substituted for $\boldsymbol{W}$ and

$$p_f(s) = s^q + a_1 s^{(q-1)} + \ldots + a_q.$$

Vector $\boldsymbol{r}$ is called a parity vector. It has generally different directions and magnitudes in the presence of the different fault modes. The n_r dimensional space of all such vectors is called the parity space, and any linear combination of the rows of (6.33) is called a parity relation.

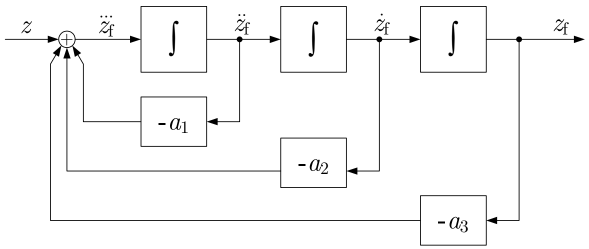


Fig. 6.2. Block diagram of a third-order state variable filter

The procedure for designing and implementing parity relations is now summarised.

Algorithm 6.1 *Parity relations for deterministic linear systems*

Given: A linear state-space model of the form (6.28) and a suitable order of derivation q

Compute off-line:
1. Matrices $\mathcal{O}, \boldsymbol{T}_{d,q}, \boldsymbol{T}_{u,q}$
2. A basis $\boldsymbol{W}$ for the left null space of $(\mathcal{O} \;\; \boldsymbol{T}_{d,q})$
3. State space filters for the estimation of the derivatives of $\boldsymbol{y}$ and $\boldsymbol{u}$ up to order q.

At each time instant:
1. Acquire the new data $\boldsymbol{y}(t)$, $\boldsymbol{u}(t)$.
2. Compute $\boldsymbol{r}_f(t)$ from (6.34).

Result: A residual vector $\boldsymbol{r}_f(t)$ for an increasing time horizon.

An alternative approach to determine analytical redundancy relations can be deduced from the input-output model of the supervised process, namely in the frequency domain. It directly results in relations involving the filtered derivatives of the measured signals. By extension this method is called the (generalised) parity space approach. It is the object of the next section. Fault isolation can be handled in the linear case in a similar way as for the nonlinear case. The detailed treatment of this issue is deferred to Section 6.4.3.

Example 6.4 *Parity relations for the ship*

A linearised model of the ship example can be written as

$$\begin{pmatrix} \dot{\omega}_3 \\ \dot{\psi} \end{pmatrix} = \begin{pmatrix} b\eta_1 & 0 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} \omega_3 \\ \psi \end{pmatrix} + \begin{pmatrix} b \\ 0 \end{pmatrix} \delta + \begin{pmatrix} 0 \\ 1 \end{pmatrix} \omega_w \tag{6.35}$$

$$\begin{pmatrix} \omega_{3m} \\ \psi_m \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} \omega_3 \\ \psi \end{pmatrix} + \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} f_\omega \\ f_\psi \end{pmatrix} + \begin{pmatrix} 1 \\ 0 \end{pmatrix} \omega_w, \tag{6.36}$$

when linearisation around $\omega_3 = 0$ is considered. Here δ, the rudder angle, is a known input, while ω_w, the wave disturbance, is an unknown input.

Straightforward computations yield the following expression for (6.31) with $q = 1$:

$$\begin{pmatrix} \omega_{3m} \\ \psi_m \\ \dot{\omega}_{3m} \\ \dot{\psi}_m \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ b\eta_1 & 0 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} \omega_3 \\ \psi \end{pmatrix} + \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ b & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} \delta \\ \dot{\delta} \end{pmatrix}$$
$$+ \begin{pmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} \omega_w \\ \dot{\omega}_w \end{pmatrix} + \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} f_\omega \\ f_\psi \\ \dot{f}_\omega \\ \dot{f}_\psi \end{pmatrix} \quad (6.37)$$

The block matrix $(\mathcal{O}\ \ \boldsymbol{T}_{d,1})$ takes the form:

$$(\mathcal{O}\ \ \boldsymbol{T}_{d,1}) = \begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ b\eta_1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{pmatrix}.$$

A basis vector for the one-dimensional left nullspace of this matrix can be written

$$\boldsymbol{W} = \begin{pmatrix} 1 & 0 & 0 & -1 \end{pmatrix}.$$

Expression (6.32) then yields

$$\omega_{3m} - \dot{\psi}_m = f_\omega - \dot{f}_\psi.$$

Hence a residual can be computed according as:

$$r_f(s) = (\omega_{3m}(s) - s\psi_m(s))/(s + a), \quad (6.38)$$

where a is a design parameter to be adjusted according to the noise level. This expression is a particulat case of the more general form (6.52) for a residual for the ship example. Further discussion of the proposed residual is provided in Section 6.4. □

6.4 Analytical redundancy relations for linear deterministic systems – frequency domain

In this section, the problems of fault detection, fault isolation and fault estimation are solved using the parity space approach to residual generation, from an input-output model of the supervised system. An alternative method would be to design observer-based residual generators, which yields similar filters, as indicated in the bibliographical notes. The observer-based approach will be used in the section on diagnosis systems design from a stochastic model, so that the reader will be acquainted with both methods.

6.4.1 Fault detection

Consider again a system described by a linear continuous-time state-space model of the form

$$\begin{aligned} \dot{\boldsymbol{x}}(t) &= \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{B}\boldsymbol{u}(t) + \boldsymbol{E}_x \boldsymbol{d}(t) + \boldsymbol{F}_x \boldsymbol{f}(t), \qquad \boldsymbol{x}(0) = \boldsymbol{x}_0 \\ \boldsymbol{y}(t) &= \boldsymbol{C}\boldsymbol{x}(t) + \boldsymbol{D}\boldsymbol{u}(t) + \boldsymbol{E}_y \boldsymbol{d}(t) + \boldsymbol{F}_y \boldsymbol{f}(t), \end{aligned} \tag{6.39}$$

where $\boldsymbol{x} \in \mathbb{R}^n$ denotes the state vector, $\boldsymbol{u} \in \mathbb{R}^m$, is the vector of measured input signals, $\boldsymbol{y} \in \mathbb{R}^p$ is the vector of measured plant output signals, $\boldsymbol{d} \in \mathbb{R}^{n_d}$ and $\boldsymbol{f} \in \mathbb{R}^{n_f}$ are vectors of unknown input signals. $\boldsymbol{f}$ represents the faults one wishes to detect, while $\boldsymbol{d}$ are unknown disturbances that should not be detected.

Such a model can also be written in terms of transfer functions:

$$\boldsymbol{y}(s) = \boldsymbol{H}_{yu}(s)\boldsymbol{u}(s) + \boldsymbol{H}_{yd}(s)\boldsymbol{d}(s) + \boldsymbol{H}_{yx}(s)\boldsymbol{x}(0) + \boldsymbol{H}_{yf}(s)\boldsymbol{f}(s), \tag{6.40}$$

where

$$\begin{aligned} \boldsymbol{H}_{yu}(s) &= \boldsymbol{C}(s\boldsymbol{I} - \boldsymbol{A})^{-1}\boldsymbol{B} + \boldsymbol{D} \\ \boldsymbol{H}_{yx}(s) &= \boldsymbol{C}(s\boldsymbol{I} - \boldsymbol{A})^{-1} \\ \boldsymbol{H}_{yd}(s) &= \boldsymbol{C}(s\boldsymbol{I} - \boldsymbol{A})^{-1}\boldsymbol{E}_x + \boldsymbol{E}_y \\ \boldsymbol{H}_{yf}(s) &= \boldsymbol{C}(s\boldsymbol{I} - \boldsymbol{A})^{-1}\boldsymbol{F}_x + \boldsymbol{F}_y. \end{aligned}$$

As indicated in Fig. 6.1, a residual generator is a filter with input $\boldsymbol{u}$ and $\boldsymbol{y}$. As supervision of linear time-invariant systems is addressed here, the class of considered filters will be restricted to linear time-invariant systems of the following form

$$\begin{aligned} \dot{\boldsymbol{z}}(t) &= \boldsymbol{A}_z \boldsymbol{z}(t) + \boldsymbol{B}_{zu}\boldsymbol{u}(t) + \boldsymbol{B}_{zy}\boldsymbol{y}(t), \qquad \boldsymbol{z}(0) = \boldsymbol{z}_0 \\ \boldsymbol{r}(t) &= \boldsymbol{C}_{rz}\boldsymbol{z}(t) + \boldsymbol{D}_{ru}\boldsymbol{u}(t) + \boldsymbol{D}_{ry}\boldsymbol{y}(t) \end{aligned} \tag{6.41}$$

or, in transfer function form, assuming zero initial conditions:

$$\boldsymbol{r}(s) = \boldsymbol{V}_{ru}(s)\boldsymbol{u}(s) + \boldsymbol{V}_{ry}(s)\boldsymbol{y}(s) = \begin{pmatrix} \boldsymbol{V}_{ru}(s) & \boldsymbol{V}_{ry}(s) \end{pmatrix} \begin{pmatrix} \boldsymbol{u}(s) \\ \boldsymbol{y}(s) \end{pmatrix}. \tag{6.42}$$

The problem of residual generator design for fault detection based on a deterministic model can be stated as follows:

Problem 6.2 (Residual generator design for fault detection based on a deterministic model)
Given a model of the supervised process of the form (6.39) or (6.40) determine a stable linear time-invariant system (6.41) or (6.42) such that:

- *In the absence of fault ($\boldsymbol{f}(t) = 0$ for all t), the output signal $\boldsymbol{r}(t), t > 0$ asymptotically decays to zero for any input $\boldsymbol{u}(t), \boldsymbol{d}(t), t > 0$ and any initial conditions $\boldsymbol{x}(0), \boldsymbol{z}(0)$.*
- *$\boldsymbol{r}(t)$ is affected by $\boldsymbol{f}(t)$.*

The first condition assures that, after a transient due to the effect of initial conditions, the residual is almost equal to zero. The second condition is a fault detectability[3] condition. The output $\boldsymbol{r}(t)$ is affected by $\boldsymbol{f}(t)$ when the transfer matrix between $\boldsymbol{f}(s)$ and $\boldsymbol{r}(s)$ obtained by combining (6.40) and (6.42) is non-zero. A time domain definition of this notion is somewhat more cumbersome, hence we defer it to Section 6.8.2.

Quite often, each component of the vector $\boldsymbol{f}(t)$ corresponds to a different fault. The detectability condition is then defined component-wise. One distinguishes the following notions:

Definition 6.1 (Weak detectability)
The i^{th} fault ($f_i(t) \neq 0$ for all $t \geq t_0$) is weakly detectable if there exists a stable residual generator such that $\boldsymbol{r}(t)$ is affected by $f_i(t)$.

In the literature weak detectability is also referred to as detectability.

Definition 6.2 (Strong detectability)
A fault f_i is strongly detectable if there exists a stable residual generator such that $\boldsymbol{r}(t)$ reaches a non-zero steady-state value for a fault signal that has a bounded final value different from zero.

6.4.2 Solution by the parity space approach

In order to determine the conditions to be fulfilled by $\boldsymbol{V}_{ru}(s)$ and $\boldsymbol{V}_{ry}(s)$ for (6.42) to be a residual generator, (6.40) is substituted for $\boldsymbol{y}(s)$ in (6.42):

$$\begin{aligned} \boldsymbol{r}(s) &= \boldsymbol{V}_{ru}(s)\boldsymbol{u}(s) + \boldsymbol{V}_{ry}(s)\left(\boldsymbol{H}_{yu}(s)\boldsymbol{u}(s) + \boldsymbol{H}_{yx}(s)\boldsymbol{x}(0)\right. \\ &\quad \left. + \boldsymbol{H}_{yd}(s)\boldsymbol{d}(s) + \boldsymbol{H}_{yf}(s)\boldsymbol{f}(s)\right) \\ &= \left(\boldsymbol{V}_{ru}(s) + \boldsymbol{V}_{ry}(s)\boldsymbol{H}_{yu}(s) \quad \boldsymbol{V}_{ry}(s)\boldsymbol{H}_{yd}(s)\right)\begin{pmatrix} \boldsymbol{u}(s) \\ \boldsymbol{d}(s) \end{pmatrix} \\ &\quad + \boldsymbol{V}_{ry}(s)\boldsymbol{H}_{yx}(s)\boldsymbol{x}(0) + \boldsymbol{V}_{ry}(s)\boldsymbol{H}_{yf}(s)\boldsymbol{f}(s) \end{aligned} \tag{6.43}$$

Figure 6.3 illustrates this residual generator.

Fulfilment of the first condition in Problem 6.2 requires:

$$\left(\boldsymbol{V}_{ru}(s) + \boldsymbol{V}_{ry}(s)\boldsymbol{H}_{yu}(s) \quad \boldsymbol{V}_{ry}(s)\boldsymbol{H}_{yd}(s)\right) = \boldsymbol{O} \tag{6.44}$$

together with the asymptotic stability of $\boldsymbol{V}_{ry}(s)\boldsymbol{H}_{yx}(s)$. Since, in healthy working mode, the plant is normally stabilised by an appropriate controller, the latter condition amounts to requiring the stability of the filter. This can be guaranteed by an

[3] This notion should not be confused with the detectability of a linear system or a pair $(C,\ A)$; indeed, the latter notion depends on the map from state to measured output, while the fault detectability is an input(i.e. fault)/output(i.e. residual) property.

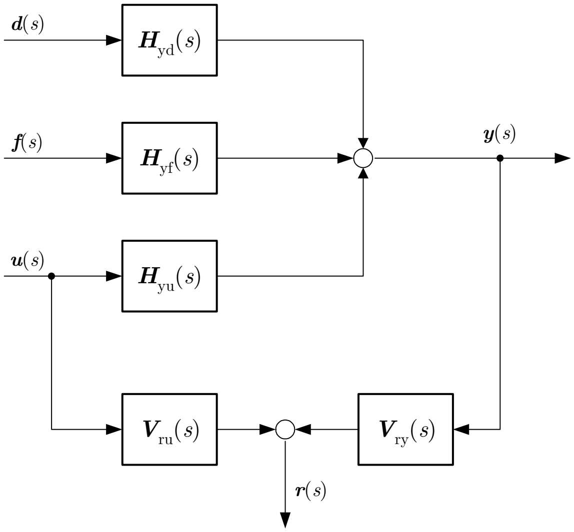


Fig. 6.3. Structure of residual generator in the parity space formulation

appropriate choice of the denominator of $\boldsymbol{V}_{ru}(s)$ and $\boldsymbol{V}_{ry}(s)$. Therefore, concentrate now on the way to achieve (6.44). The question of fault detectability will be addressed once the class of all filters that fulfil (6.44) is characterised.

Notice that (6.44) can be rewritten:

$$\left(\boldsymbol{V}_{ry}(s) \quad \boldsymbol{V}_{ru}(s)\right) \begin{pmatrix} \boldsymbol{H}_{yu}(s) & \boldsymbol{H}_{yd}(s) \\ \boldsymbol{I} & \boldsymbol{O} \end{pmatrix} = 0. \tag{6.45}$$

For any filter, the least common multiple of the denominators of the entries of $\boldsymbol{V}_{ry}(s)$ and $\boldsymbol{V}_{ru}(s)$, $p(s)$ can be determined. Using $p(s)$, the left most matrix in (6.45) can be written:

$$\left(\boldsymbol{V}_{ry}(s) \quad \boldsymbol{V}_{ru}(s)\right) = \frac{\left(\bar{\boldsymbol{V}}_{ry}(s) \quad \bar{\boldsymbol{V}}_{ru}(s)\right)}{p(s)}, \tag{6.46}$$

where $\bar{\boldsymbol{V}}_{ry}(s)$ and $\bar{\boldsymbol{V}}_{ru}(s)$ are suitable polynomial matrices. Hence, the whole class of filters that meet (6.45) can be obtained by characterising the set of polynomial matrices $\left(\bar{\boldsymbol{V}}_{ry}(s) \quad \bar{\boldsymbol{V}}_{ru}(s)\right)$ that fulfil:

$$\left(\bar{\boldsymbol{V}}_{ry}(s) \quad \bar{\boldsymbol{V}}_{ru}(s)\right) \begin{pmatrix} \boldsymbol{H}_{yu}(s) & \boldsymbol{H}_{yd}(s) \\ \boldsymbol{I} & \boldsymbol{O} \end{pmatrix} = 0. \tag{6.47}$$

This is the set of polynomial matrices that lie in the left nullspace of

$$\boldsymbol{H}(s) = \begin{pmatrix} \boldsymbol{H}_{yu}(s) & \boldsymbol{H}_{yd}(s) \\ \boldsymbol{I} & \boldsymbol{O} \end{pmatrix}. \tag{6.48}$$

This space is denoted $\mathcal{N}_L(\boldsymbol{H}(s))$. Its dimension is equal to the difference between the number of rows of $\boldsymbol{H}(s)$ and its rank, namely

$$\dim(\mathcal{N}_L(\boldsymbol{H}(s))) = m + p - \text{ rank } \boldsymbol{H}(s) = m + p - (m + n_d) = p - n_d,$$

where m is the number of inputs, p the number of outputs, and n_d the number of unknown inputs (disturbances). It has been assumed that $\boldsymbol{H}_{yu}(s)$ and $\boldsymbol{H}_{yd}(s)$ have full column rank[4]. Notice that the number of plant output signals must be larger than the number of disturbances for the left nullspace to be non-zero.

One way to characterise the set of polynomial matrices $\left(\bar{\boldsymbol{V}}_{ry}(s) \quad \bar{\boldsymbol{V}}_{ru}(s)\right)$ that meet (6.47) is to determine an irreducible polynomial basis, for the rational vector space $\mathcal{N}_L(\boldsymbol{H}(s))$. Further, let $\boldsymbol{F}(s)$ be a matrix of which the rows make such an irreducible polynomial basis, then any suitable matrix $\left(\bar{\boldsymbol{V}}_{ry}(s) \quad \bar{\boldsymbol{V}}_{ru}(s)\right)$ can be obtained by combinations of the rows of $\boldsymbol{F}(s)$, namely

$$\left(\bar{\boldsymbol{V}}_{ry}(s) \quad \bar{\boldsymbol{V}}_{ru}(s)\right) = \boldsymbol{Q}(s)\boldsymbol{F}(s), \tag{6.49}$$

where $\boldsymbol{Q}(s)$ is an arbitrary polynomial matrix with appropriate number of columns.

A general parametrisation of the family of residual generators is obtained from (6.49). Substitution of (6.49) into (6.46) yields

$$\left(\boldsymbol{V}_{ry}(s) \quad \boldsymbol{V}_{ru}(s)\right) = \frac{\boldsymbol{Q}(s)\boldsymbol{F}(s)}{p(s)}.$$

Introducing this expression into (6.42) finally results in

$$r(s) = \frac{\boldsymbol{Q}(s)\boldsymbol{F}(s)}{p(s)} \begin{pmatrix} \boldsymbol{y}(s) \\ \boldsymbol{u}(s) \end{pmatrix}. \tag{6.50}$$

The choice of the matrix $\boldsymbol{Q}(s)$ and the polynomial $p(s)$ depends on the specification of the diagnosis problem. Typically the residual generator should ensure filtering of high frequency disturbances which always exist, even though they were not considered in the model, and adequate properties at low frequencies. Sometimes, precise information on the frequency range of the fault is available, and $\boldsymbol{Q}(s)/p(s)$ can be designed to perform appropriate filtering.

Remark 6.4 *Link with parity relations deduced from the state-space model*
Equation (6.34) clearly has the same form as (6.42) and, by construction, it fulfils the first condition of problem 6.2 provided $p_f(s)$ has all its roots in the open left-half plane. Hence there exist a matrix $\boldsymbol{Q}(s)$ and a polynomial $p(s)$ for which (6.34) and (6.50) are identical. □

Modelling uncertainty. Although modelling uncertainties have not been introduced here, they can be accounted for a posteriori when $\boldsymbol{F}(s)$ has several rows. $\boldsymbol{Q}(s)$ is then used to select appropriate rows in $\boldsymbol{F}(s)$. To explain the idea, let $\boldsymbol{F}_i(s), i = 1, \cdots, n_r$ denote the i^{th} row of $\boldsymbol{F}(s)$, and consider the scalar residuals

$$r_i(s) = \frac{\boldsymbol{F}_i(s)}{p(s)} \begin{pmatrix} \boldsymbol{y}(s) \\ \boldsymbol{u}(s) \end{pmatrix} \qquad i = 1, \cdots, n_r.$$

[4] The notion of rank considered here is the normal rank computed as $\max_s \text{ rank } H(s)$ where the maximum is taken over all complex values of s.

By performing a simulation of all these filters with actual plant measurements as input, one may compare how significantly the actual residuals $r_i(t)$, $i = 1, ..., n_r$ deviate from zero in the absence of fault, once the transient due to initial conditions has vanished. This reflects the effect of modelling errors on the residuals. Besides, by using faulty data obtained with a simulation or corresponding to actual plant measurements it is also possible to compare the actual sensitivities to faults. A kind of "signal to noise ratio" could be defined for each residual as

$$SNR_i = \frac{\int_{t_0}^{t_0+T} r_i^F(t)^2 dt}{\int_{t_1}^{t_1+T} r_i^{FF}(t)^2 dt}, \tag{6.51}$$

where $r_i^F(t)$ denotes the residual obtained with the measurement associated to the faulty mode, and r_i^{FF} corresponds to the fault free situation. T is a user defined horizon, t_0 and t_1 are time instants associated to faulty and fault free data sequences. Matrix $\boldsymbol{Q}(s)$ should then be chosen to select the components of $\boldsymbol{r}(s)$ for which the "signal-to-noise ratio" is significantly larger than 1.

Computational aspects. The problem of finding an irreducible polynomial basis for $\mathcal{N}_L(\boldsymbol{H}(s))$ can be transformed into the determination of a similar basis for a polynomial matrix instead of the rational matrix $\boldsymbol{H}(s)$. It suffices to notice that

$$\boldsymbol{H}(s) = \bar{\boldsymbol{H}}(s)/h(s),$$

where $h(s)$ is the least common multiple of all denominators. An irreducible polynomial basis for $\bar{\boldsymbol{H}}(s)$ is also an irreducible polynomial basis for $\boldsymbol{H}(s)$, and vice-versa. Numerically stable algorithms for the computation of an irreducible polynomial basis are available in the literature, and they have been programmed in the polynomial toolbox of MATLAB.

The symbolic tools Maple and Mathematica can calculate a basis for the left nullspace of $\boldsymbol{H}(s)$. The Maple command *nullspace basis* applied to the matrix $\boldsymbol{H}'(s)$ will provide the row basis given in analytical form. Calculation of a basis is not unique, so the result can be expanded or reduced by a polynomial fraction as desired. The result is not necessarily irreducible, either, but the reduction to an irreducible basis is usually straightforward once a factorisation is made of the entries in the nullspace basis[5].

Example 6.4 (cont.) *Parity relations for the ship*

A model of the form (6.40) can be easily deduced from the linear state-space model for the ship example. The following transfer matrices are obtained when sensor faults are considered, and when state and sensor noise are neglected:

$$\boldsymbol{H}_{yu}(s) = \begin{pmatrix} \frac{b}{s-b\eta_1} \\ \frac{b}{(s-b\eta_1)s} \end{pmatrix}, \quad \boldsymbol{H}_{yd}(s) = \begin{pmatrix} 1 \\ \frac{1}{s} \end{pmatrix}, \quad \boldsymbol{H}_{yx}(s) = \begin{pmatrix} \frac{1}{s-b\eta_1} & 0 \\ \frac{1}{s(s-b\eta_1)} & \frac{1}{s} \end{pmatrix}$$

[5] The Maple symbolic mathematics engine is a stand-alone product. It is also a part of the MATLAB Symbolic Toolbox. MATLAB®, Maple® and Mathematica® are registered trademarks of their respective owners.

and $\boldsymbol{H}_{yf} = \boldsymbol{I}_2$. In the above expressions,

$$\begin{aligned}
\boldsymbol{x}(t) &= (\omega_3(t) \quad \psi(t))' \\
\boldsymbol{y}(t) &= (\omega_{3m}(t) \quad \psi_m(t))' \\
d(t) &= \omega_w(t) \\
u(t) &= \delta(t) \\
\boldsymbol{f}(t) &= (f_\omega(t) \quad f_\psi(t))'
\end{aligned}$$

hold. It is assumed that η_1 is negative, so that the ship is stable. $\boldsymbol{H}_{yx}(s)$ is not asymptotically stable however, due to the integrator linking speed and position. We shall see below what slight modification must be introduced in the theory to handle the pole at the origin.

The matrix $\boldsymbol{H}(s)$ takes the form:

$$\boldsymbol{H}(s) = \begin{pmatrix} \frac{b}{s-b\eta_1} & 1 \\ \frac{b}{s(s-b\eta_1)} & \frac{1}{s} \\ 1 & 0 \end{pmatrix} = \frac{1}{s(s-b\eta_1)} \begin{pmatrix} bs & s(s-b\eta_1) \\ b & (s-b\eta_1) \\ s(s-b\eta_1) & 0 \end{pmatrix}.$$

The last matrix corresponds to $\bar{\boldsymbol{H}}(s)$. An irreducible basis for its left nullspace can be calculated, or found by inspection, to be

$$F(s) = (1 \quad -s \quad 0)\,.$$

Thus, any vector of rational functions of the form

$$\begin{pmatrix} \frac{q(s)}{p(s)} & \frac{-sq(s)}{p(s)} & 0 \end{pmatrix},$$

where $p(s)$ is an arbitrary polynomial with roots in the left half plane and $q(s)$ is an arbitrary polynomial with degree less than $p(s)$, fulfils condition (6.45). Candidate residual generators have the form:

$$r(s) = \frac{q(s)}{p(s)}\omega_{3m}(s) - \frac{sq(s)}{p(s)}\psi_m(s). \tag{6.52}$$

Notice that, by setting $q(s) = 1$, one recovers (6.38) with $p(s) = s + a$.

Substituting the model equations for $\omega_3(s)$ and $\psi(s)$ yields

$$\begin{aligned}
r(s) &= \frac{q(s)}{p(s)}\left(\frac{b}{s-b\eta_1}\delta(s) + \omega_w(s) + \frac{1}{s-b\eta_1}\omega_3(0) + f_\omega(s)\right) \\
&\quad -\frac{sq(s)}{p(s)}\left(\frac{b}{s(s-b\eta_1)}\delta(s) + \frac{\omega_w(s)}{s} + \frac{1}{s(s-b\eta_1)}\omega_3(0) + \frac{1}{s}\psi(0) + f_\psi(s)\right) \\
&= -\frac{q(s)}{p(s)}\psi(0) + \frac{q(s)}{p(s)}f_\omega(s) - \frac{sq(s)}{p(s)}f_\psi(s).
\end{aligned}$$

In order to assure that the residual asymptotically vanishes, two solutions are possible:

- Introduction of a derivative action in $q(s)$, so that $q(s) = s\bar{q}(s)$ and the term involving $\psi(0)$ in the above equation is null at steady state.
- Modification of (6.52) by adding a correction term associated with the initial position (supposed to be measured correctly). This yields

  $$r(s) = \frac{q(s)}{p(s)}\psi(0) + \frac{q(s)}{p(s)}\omega_{3m}(s) - \frac{sq(s)}{p(s)}\psi_m(s)$$

  or, after substitution of $\omega_{3m}(s)$ and $\psi_m(s)$ in terms of the model equations:

$$r(s) = \frac{q(s)}{p(s)} f_\omega(s) - \frac{sq(s)}{p(s)} f_\psi(s). \tag{6.53}$$

The first solution also introduces a derivative action in the transfer functions between $f_\omega(s)$ and $r(s)$, and between $f_\psi(s)$ and $r(s)$. Hence step like faults do not have any steady state effect on the residual. On the other hand, in (6.53), $q(s)$ can be chosen so that a step-like fault f_ω has a steady state effect on r, but a step like fault in f_ψ can only influence temporarily r due to the zero at the origin in $\frac{sq(s)}{p(s)}$. Application of the theory below will indicate that, indeed, fault f_ω is strongly detectable, but f_ψ is only weakly detectable. □

Example 6.5 *Parity relations – ship with three output measurements*

Some useful observations can be made later from the above example but using an additional instrument to measure the ship heading. This third instrument is taken to be independent of the other two. This is a realistic case since redundant heading instruments are required for most merchant ships.

With two independent heading angle measurements

$$y_2(s) = \psi_m^{(1)}(s)$$

and

$$y_3(s) = \psi_m^{(2)}(s),$$

the matrix $\boldsymbol{H}(s)$ takes the form:

$$\boldsymbol{H}(s) = \frac{1}{s(s-b\eta_1)} \begin{pmatrix} bs & s(s-b\eta_1) \\ b & (s-b\eta_1) \\ b & (s-b\eta_1) \\ s(s-b\eta_1) & 0 \end{pmatrix}.$$

The nullspace basis for $\boldsymbol{H}(s)$ is computed to be

$$\begin{pmatrix} (\frac{-1}{s} & 1 & 0 & 0) \\ (\frac{-1}{s} & 0 & 1 & 0) \end{pmatrix}.$$

This means a family of candidate residual generators exist, which have the form

$$\boldsymbol{r}(s) = \frac{q(s)}{p(s)} \begin{pmatrix} \frac{-1}{s} & 1 & 0 & 0 \\ \frac{-1}{s} & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} \omega_{3m}(s) \\ \psi_m^{(1)}(s) \\ \psi_m^{(2)}(\delta) \\ \delta(s) \end{pmatrix}.$$

The relation between components of the residual vector $\boldsymbol{r}(s)$ to faults $\boldsymbol{f}(s)$ and wave disturbance ω_w is

$$\begin{aligned} r_1(s) &= \frac{q(s)}{p(s)} \left(-\tfrac{1}{s} f_\omega(s) + f_\psi^{(1)}(s)\right) \\ r_2(s) &= \frac{q(s)}{p(s)} \left(-\tfrac{1}{s} f_\omega(s) + f_\psi^{(2)}(s)\right). \end{aligned}$$

It is evident that all elements of the residual are decoupled from the wave disturbance, which was the intention.

Forming a third residual using the plain difference between heading angle measurements

$$r_3(s) = \psi_m^{(1)}(s) - \psi_m^{(2)}(s),$$

which would be a straightforward choice as an output parity equation, is indeed possible, but since this would be a linear relation between the two residuals already defined, this would not add to the information contained in the residual vector. □

Fault detectability. To deduce theoretical results on fault detectability, the expression of the residual in the presence of faults must be determined. Substituting (6.45) into (6.43) yields

$$\begin{aligned} \boldsymbol{r}(s) &= \boldsymbol{V}_{ry}(s)\boldsymbol{H}_{yx}(s)\boldsymbol{x}(0) + \boldsymbol{V}_{ry}(s)\boldsymbol{H}_{yf}(s)\boldsymbol{f}(s) \\ &= \boldsymbol{V}_{ry}(s)\boldsymbol{H}_{yx}(s)\boldsymbol{x}(0) + \sum_{i=1}^{n_f}\boldsymbol{V}_{ry}(s)\boldsymbol{H}^i_{yf}(s)f_i(s), \end{aligned} \tag{6.54}$$

where $\boldsymbol{H}^i_{yf}(s)$ denotes the i^{th} column of $\boldsymbol{H}_{yf}(s)$. It can be shown that a necessary and sufficient condition for detectability of the i^{th} fault is:

$$\boldsymbol{V}_{ry}(s)\boldsymbol{H}^i_{yf}(s) \neq 0, \tag{6.55}$$

where $\boldsymbol{V}_{ry}(s)$ also fulfils

$$\boldsymbol{V}_{ry}(s)\boldsymbol{H}_{yd}(s) = 0. \tag{6.56}$$

The latter condition comes from the second entry in (6.44). For (6.55) and (6.56) to be simultaneously verified, one should not be able to express $\boldsymbol{H}^i_{yf}(s)$ as a linear combination of the columns of $\boldsymbol{H}_{yd}(s)$. In other words, there cannot exists any non-zero polynomial set $\alpha_0(s)$, $\alpha_1(s)$, $\cdots$, $\alpha_{n_d}(s)$ such that:

$$\alpha_0(s)\boldsymbol{H}^i_{yf}(s) + \alpha_1(s)\boldsymbol{H}^1_{yd}(s) + \cdots + \alpha_{n_d}(s)\boldsymbol{H}^{n_d}_{yd}(s) = 0$$

This condition is fulfilled when

$$\text{rank } \begin{pmatrix} \boldsymbol{H}_{yd}(s) & \boldsymbol{H}^i_{yf}(s) \end{pmatrix} > \text{ rank } \boldsymbol{H}_{yd}(s), \tag{6.57}$$

where

$$\text{rank } \boldsymbol{A}(s) = \max_s \text{ rank } \boldsymbol{A}(s)$$

denotes the normal rank of the rational matrix $\boldsymbol{A}(s)$. In the latter expression, the "rank"-operation in the right hand side acts on a matrix of complex numbers obtained for a specific value of s. It can thus be evaluated in the standard way. (6.57) is actually a necessary and sufficient condition for the i^{th} fault to be weakly detectable.

To determine a test for strong fault detectability, substitute the model (6.40) for $y(s)$ in the parametrisation of the class of residual generators (6.50)

$$\begin{aligned} \boldsymbol{r}(s) &= \frac{\boldsymbol{Q}(s)\boldsymbol{F}(s)}{p(s)} \begin{pmatrix} \boldsymbol{H}_{yu}(s) & \boldsymbol{H}_{yd}(s) & \boldsymbol{H}_{yf}(s) \\ \boldsymbol{I} & \boldsymbol{O} & \boldsymbol{O} \end{pmatrix} \begin{pmatrix} \boldsymbol{u}(s) \\ \boldsymbol{d}(s) \\ \boldsymbol{f}(s) \end{pmatrix} \\ &= \frac{\boldsymbol{Q}(s)\boldsymbol{F}(s)}{p(s)} \begin{pmatrix} \boldsymbol{H}_{yf}(s) \\ \boldsymbol{O} \end{pmatrix} \boldsymbol{f}(s), \end{aligned} \tag{6.58}$$

where the second equality accounts for the fact that $\boldsymbol{F}(s)$ is a basis for the left nullspace of $\boldsymbol{H}(s)$. The transient term due to $\boldsymbol{x}(0)$ was not considered as its effect vanishes when t tends to infinity. Strong detectability of the fault f_i is thus achieved if there exists some polynomial $p(s)$ and polynomial matrix $\boldsymbol{Q}(s)$ such that

$$\frac{\boldsymbol{Q}(s)\boldsymbol{F}(s)}{p(s)} \begin{pmatrix} \boldsymbol{H}^i_{yf}(s) \\ \boldsymbol{O} \end{pmatrix}\Bigg|_{s=0} \neq 0. \tag{6.59}$$

As $p(0)$ is necessarily chosen non-zero to assure asymptotic stability of the filter, and $\boldsymbol{Q}(s)$ can be chosen arbitrarily, a necessary and sufficient condition for strong fault detectability is

$$\boldsymbol{F}(s) \begin{pmatrix} \boldsymbol{H}^i_{yf}(s) \\ \boldsymbol{O} \end{pmatrix}\Bigg|_{s=0} \neq 0. \tag{6.60}$$

Notice that this expression may be different from $\boldsymbol{F}(0) \begin{pmatrix} \boldsymbol{H}^i_{yf}(0) \\ \boldsymbol{O} \end{pmatrix}'$ and, hence, substitution by $s = 0$ must be performed after computation of the matrix product.

Example 6.6 *Detectability - ship with two output measurements*

To check that fault $f_\psi^{(1)}$ is detectable, (6.57) is applied as follows

$$\text{rank} \begin{pmatrix} 1 & 1 \\ \frac{1}{s} & 0 \end{pmatrix} > \text{rank} \begin{pmatrix} 1 \\ \frac{1}{s} \end{pmatrix}.$$

Similarly, the inequality

$$\text{rank} \begin{pmatrix} 1 & 0 \\ \frac{1}{s} & 1 \end{pmatrix} > \text{rank} \begin{pmatrix} 1 \\ \frac{1}{s} \end{pmatrix}$$

ensures that $f_\psi^{(2)}$ is detectable. Condition (6.60) is now used to check strong fault detectability. For fault $f_\psi^{(1)}$ it yields

$$\begin{pmatrix} 1 & -s & 0 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}\Bigg|_{s=0} = 1.$$

Thus fault $f_\psi^{(1)}$ is strongly detectable. For fault $f_\psi^{(2)}$ one gets

$$\begin{pmatrix} 1 & -s & 0 \end{pmatrix} \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}\Bigg|_{s=0} = 0,$$

which indicates that $f_\psi^{(2)}$ is not strongly detectable, as was expected. □

The procedure for residual generator design can be summarised as follows.

Algorithm 6.2 *Residual generator design with the parity space method*

Given: A model of the supervised system in the form (6.40).

Computation:

1. Compute matrix $\boldsymbol{H}(s)$ as defined by (6.48).
2. Determine an irreducible polynomial basis for $\mathcal{N}_L(\boldsymbol{H}(s))$, and let $\boldsymbol{F}(s)$ be the matrix whose rows make such a basis. If $\boldsymbol{F}(s) = \boldsymbol{O}$, the problem has no solution.
3. Design the filter $\frac{\boldsymbol{Q}(s)}{p(s)}$ as a low-pass or a band-pass filter which possibly selects appropriate rows in $\boldsymbol{F}(s)$ according to $SNR_i, i = 1, \cdots, \beta$ (cf. (6.51)).
4. Check for weak or strong fault detectability as needed.

Result: A residual generator in the form (6.50).

6.4.3 Fault isolation

For fault-tolerant control, faults should not only be detected, but also be isolated, namely the faulty components should be determined. The problem of residual generator design for fault detection and isolation based on a deterministic model can be stated as follows.

Consider a system described by a continuous-time linear state-space model of the form

$$\begin{aligned} \dot{\boldsymbol{x}}(t) &= \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{B}\boldsymbol{u}(t) + \sum_{j=1}^{n_f} \boldsymbol{F}_x^j \boldsymbol{f}_j(t), \qquad \boldsymbol{x}(0) = \boldsymbol{x}_0 \\ \boldsymbol{y}(t) &= \boldsymbol{C}\boldsymbol{x}(t) + \boldsymbol{D}\boldsymbol{u}(t) + \sum_{j=1}^{n_f} \boldsymbol{F}_y^j \boldsymbol{f}_j(t), \end{aligned} \tag{6.61}$$

where $\boldsymbol{f}_j \in \mathbb{R}^{n_{f_j}}, j = 1, \ldots, n_f$ represent the faults that must be detected and isolated. In terms of transfer functions, (6.61) can be written as

$$\boldsymbol{y}(s) = \boldsymbol{H}_{yu}(s)\boldsymbol{u}(s) + \boldsymbol{H}_{yx}(s)\boldsymbol{x}(0) + \sum_{j=1}^{n_f} \boldsymbol{H}_{yf_j}(s)\boldsymbol{f}_j(s), \tag{6.62}$$

where

$$\boldsymbol{H}_{yu}(s) = \boldsymbol{C}(s\boldsymbol{I} - \boldsymbol{A})^{-1}\boldsymbol{B} + \boldsymbol{D},\ \boldsymbol{H}_{yx}(s) = \boldsymbol{C}(s\boldsymbol{I} - \boldsymbol{A})^{-1}$$

and

$$\boldsymbol{H}_{yf_j}(s) = \boldsymbol{C}(s\boldsymbol{I} - \boldsymbol{A})^{-1}\boldsymbol{F}_x^j + \boldsymbol{F}_y^j.$$

Problem 6.3 (Residual generator design for fault detection and isolation based on a deterministic model)

Given a model of the supervised process of the form (6.61) or (6.62), determine a set of n_f stable linear time-invariant filters described by

$$\begin{aligned} \dot{\boldsymbol{z}}_\ell(t) &= \boldsymbol{A}_{z,\ell}\boldsymbol{z}_\ell(t) + \boldsymbol{B}_{zu,\ell}\boldsymbol{u}(t) + \boldsymbol{B}_{zy,\ell}\boldsymbol{y}(t), \quad \boldsymbol{z}_\ell(0) = \boldsymbol{z}_{0,\ell} \\ \boldsymbol{r}_\ell(t) &= \boldsymbol{C}_{rz,\ell}\boldsymbol{z}_\ell(t) + \boldsymbol{D}_{ru,\ell}\boldsymbol{u}(t) + \boldsymbol{D}_{ry,\ell}\boldsymbol{y}(t), \quad \ell = 1, \ldots, n_f \end{aligned} \tag{6.63}$$

or, in transfer function form, assuming zero initial conditions,

$$\boldsymbol{r}_\ell(s) = \boldsymbol{V}_{ru,\ell}(s)\boldsymbol{u}(s) + \boldsymbol{V}_{ry,\ell}(s)\boldsymbol{y}(s), \qquad \ell = 1, \ldots, n_f, \tag{6.64}$$

such that the following conditions are met.

- $\boldsymbol{r}_\ell(t)$ *asymptotically decays to zero for any* $\boldsymbol{u}(t)$ *and any* $\boldsymbol{f}_j(t),\ j = 1,\ \ldots,\ n_f,\ j \neq \ell,\ t > 0.$
- $r_\ell(t)$ *is affected by* $f_\ell(t)$.

In this problem statement, the ℓ^{th} residual can only be affected by the ℓ^{th} fault, and not by the others. The table below represents this situation when $n_f = 3$.

Table 6.1 Effects of the faults on the residuals

↗	$\boldsymbol{f}_1$	$\boldsymbol{f}_2$	$\boldsymbol{f}_3$
$\boldsymbol{r}_1$	×	0	0
$\boldsymbol{r}_2$	0	×	0
$\boldsymbol{r}_3$	0	0	×

A "×" in Table 6.1 indicates that the fault in the corresponding column affects the residual of the corresponding row.

The faults that do not affect the ℓ^{th} residual can be seen as unknown inputs to which this residual should not be sensitive. Hence, to design a residual generator that output r_ℓ, it suffices to use the solution of the problem of residual generation for fault detection in which vector $\boldsymbol{d}$ is replaced by $(\boldsymbol{f}_1' \cdots \boldsymbol{f}_{\ell-1}'\ \boldsymbol{f}_{\ell+1}' \cdots \boldsymbol{f}_{n_f}')'$. n_f such problems should be solved for $\ell = 1, \ldots, n_f$ in order to obtain the n_f filters that make a solution to the fault isolation problem.

From the conditions for fault detectability, the following conditions can be deduced for the above scheme to work:

$$\text{rank}\ \left(\boldsymbol{H}_{yf_\ell}(s)\ \ \boldsymbol{H}_{yf_j}(s)\right) > \ \text{rank}\ \boldsymbol{H}_{yf_j}(s) \tag{6.65}$$

for all $\ell,\ j = 1,\ \ldots,\ n_f,\ \ell \neq j$.

A suficient condition for (6.65) to hold is

$$\sum_{\substack{j=1,n_f \\ j \neq \ell}} n_{f_j} < p, \tag{6.66}$$

where p is the number of measured output signals (dimension of $\boldsymbol{y}$).

When condition, (6.65) is not met, the diagonal structure of Table 6.1 cannot be obtained, and one should attempt to group the fault vectors in different classes and to generate residuals that are affected by a specific fault class and not by the others. The table below illustrates one way to perform such a grouping, in a situation where $n_f = 3$ and two residual generators are designed.

Table 6.2 Effects of the faults on the residuals – non-diagonal structure

$\nearrow$	$\boldsymbol{f}_1$	$\boldsymbol{f}_2$	$\boldsymbol{f}_3$
r_1	$\times$	$\times$	0
r_2	$\times$	0	$\times$

In the situation of Table 6.2, all three faults can be distinguished as the combination of r_1 and r_2 reacts differently to each fault. However, simultaneous faults cannot be isolated because they affect both residuals in all cases.

Example 6.7 *Isolability - ship with three output measurements*

For the ship with one rate measurement and two heading measurements (Example 6.5), a residual generator is achieved, which was decoupled from the disturbance,

$$\begin{pmatrix} r_1(s) \\ r_2(s) \end{pmatrix} = \begin{pmatrix} -\frac{1}{s} & 1 & 0 \\ -\frac{1}{s} & 0 & 1 \end{pmatrix} \begin{pmatrix} f_{\omega 3}(s) \\ f_\psi^{(1)}(s) \\ f_\psi^{(2)}(s) \end{pmatrix}. \tag{6.67}$$

This residual generator has the properties shown in Table 6.2. □

Sensor fault isolation in a fault-tolerant control setting. If it has been detected that one out of a set of faults is present, but it has not been possible to isolate which fault is actually present, and this was due to the design of the residual generator specification, alternatives are available on the fault-tolerant setting because the supervisory system has control of the input signals to the plant. Similar to system identification, where a dedicated test signal is applied to obtain the optimal information about a particular parameter, a dedicated test signal can be applied on the control input to help confirm particular hypotheses. This procedure can help reduce time to diagnose and therefore time to reconfigure a controller.

Example 6.8 *Dedicated test signal for isolation – ship steering*

If two identical rate sensors are available in the ship steering example, and the residual generator was designed to be insensitive to the wave disturbance, it is not possible to isolate faults

f_ω^1 and f_ω^2. In a fault-tolerant control setting, employ active test signal generation to isolate the fault once it has been detected that one of the rate sensor units is defect. Define a dedicated test signal

$$\delta(t) = \tilde{\delta}(t),\ t \subset [0,\ T],$$

which is applied immediately after the hypothesis of

$$\left\{\hat{f}_\omega^1(t) \vee \hat{f}_\omega^2(t)\right\} \neq 0$$

is confirmed. Observe a-priori the response in the non-faulty condition

$$\omega_3^{rec}(t) = g_\omega\left(\tilde{\delta}(t),\ U(t)\right),\ t \subset [0, T]$$

note that the function g_{ω_3} is not calculated, the angular rate is merely recorded and stored. Calculate the correlations

$$\begin{aligned} cor_{31}(t) &= \frac{1}{t}\int_0^t \omega_3^{rec}(\tau)\omega_{3m}^1(\tau)d\tau \\ cor_{21}(t) &= \frac{1}{t}\int_0^t \omega_{3m}^2(\tau)\omega_{3m}^1(\tau)d\tau \\ cor_{32}(t) &= \frac{1}{t}\int_0^t \omega_3^{rec}(\tau)\omega_{3m}^2(\tau)d\tau. \end{aligned}$$

These correlation signals with appropriate normalisation make it straightforward to determine which hypothesis is the most likely. □

6.4.4 Fault estimation

The isolation schemes signify which fault is present but do not assess the magnitude of the fault. Fault estimates are needed in certain fault accommodation approaches as was indicated in Section 6.1. This notion is defined as follows.

Definition 6.3 (Fault estimation)
Fault estimation is the ability to estimate the magnitude of a fault $\boldsymbol{f}_i(t)$ and its time history.

Combining (6.42) and (6.58), the link between the fault vector $\boldsymbol{f}(s)$ and the residual $\boldsymbol{r}(s)$ is seen to be

$$\boldsymbol{r}(s) = \boldsymbol{V}_{ru}(s)\boldsymbol{u}(s) + \boldsymbol{V}_{ry}(s)\boldsymbol{y}(s) = \frac{\boldsymbol{Q}(s)\boldsymbol{F}(s)}{p(s)}\begin{pmatrix}\boldsymbol{H}_{yf}(s)\\ \boldsymbol{O}\end{pmatrix}\boldsymbol{f}(s), \qquad (6.68)$$

where it is assumed that initial conditions have vanished. Letting

$$\boldsymbol{F}(s) = \left(\boldsymbol{F}_1(s) \quad \boldsymbol{F}_2(s)\right),$$

where $\boldsymbol{F}_1(s)$ has p columns and $\boldsymbol{F}_2(s)$ has m columns, Eq. (6.68) can be written

$$\boldsymbol{r}(s) = \boldsymbol{V}_{ru}(s)\boldsymbol{u}(s) + \boldsymbol{V}_{ry}(s)\boldsymbol{y}(s) = \frac{\boldsymbol{Q}(s)\boldsymbol{F}_1(s)}{p(s)}\boldsymbol{H}_{yf}(s)\boldsymbol{f}(s). \qquad (6.69)$$

On the other hand, Eq. (6.54) yields the following relation when the transient due to the inital conditions is neglected

$$\boldsymbol{r}(s) = \boldsymbol{V}_{ry}(s)\boldsymbol{H}_{yf}(s)\boldsymbol{f}(s),$$

hence

$$\boldsymbol{V}_{ry}(s) := \frac{\boldsymbol{Q}(s)\boldsymbol{F}_1(s)}{p(s)}. \tag{6.70}$$

As a compact notation, introduce $\boldsymbol{H}_{rf}(s)$ by

$$\boldsymbol{H}_{rf}(s) := \boldsymbol{V}_{ry}(s)\boldsymbol{H}_{yf}(s) = \frac{\boldsymbol{Q}(s)\boldsymbol{F}_1(s)}{p(s)}\boldsymbol{H}_{yf}(s).$$

If it is possible to determine a suitable left inverse to $\boldsymbol{H}_{rf}(s)$, say $\boldsymbol{G}(s)$, an estimate of $\boldsymbol{f}(s)$ would be

$$\hat{\boldsymbol{f}}(s) = \boldsymbol{G}(s)\boldsymbol{r}(s) = \boldsymbol{G}(s)(\boldsymbol{V}_{ru}(s)\boldsymbol{u}(s) + \boldsymbol{V}_{ry}(s)\boldsymbol{y}(s)). \tag{6.71}$$

Left inverse transformation. If the polynomial matrix $\boldsymbol{H}_{rf}(s)$ is square, then the estimate $\hat{\boldsymbol{f}}(s) = \boldsymbol{H}_{rf}^{-1}(s)$ where the ij-th element of $\boldsymbol{H}_{rf}^{-1}$, call it h_{ij} is the usual inverse

$$h_{ij}(s) = \frac{1}{\det(\boldsymbol{H}_{rf}(s))}(-1)^{i+j}(\boldsymbol{M}_{ji}(s)),$$

where $\boldsymbol{M}_{ji}(s)$ is the determinant of the matrix formed by $\boldsymbol{H}_{rf}(s)$ after deleting row j and column i.

If $\boldsymbol{H}_{rf}(s)$ is non-square, with l rows and n_f columns, then, there exists a left pseudo-inverse $\boldsymbol{G}(s)$ of $\boldsymbol{H}_{rf}(s)$ if and only if

$$\text{rank}\ (\boldsymbol{H}_{rf}(s)) = n_f,$$

where the normal rank is considered. $\boldsymbol{G}(s)$ is given as

$$\boldsymbol{G}(s) = (\boldsymbol{H}'_{rf}(s)\boldsymbol{H}_{rf}(s))^{-1}\boldsymbol{H}'_{rf}(s). \tag{6.72}$$

The pseudo-inverse has the property

$$\boldsymbol{G}(s)\boldsymbol{H}_{rf}(s) = \boldsymbol{I}_{n_f}$$

with $\boldsymbol{I}_{n_f}$ being the unity matrix of dimension n_f.

Remark 6.5 *Causality of solution*
To be able to implement the filter Eq. (6.71), $\boldsymbol{G}(s)\,\boldsymbol{V}_{ru}(s)$ and $\boldsymbol{G}(s)\,\boldsymbol{V}_{ry}(s)$ must be proper and stable transfer functions. This may not be true when $\boldsymbol{G}(s)$ is computed as above. A modified procedure can be found in the literature (see the bibliographical notes for this chapter).
□

Fault estimation after isolation. A necessary condition to be able to compute the above rational estimate, based on a pseudo inverse transformation, is that the rank of the $\boldsymbol{H}_{rf}$ matrix is equal to the number of faults to be estimated. As the number of faults is often larger than the number of independent residuals, it is necessary to take advantage of the results of the fault isolation to limit estimation of faults to those that the isolation algorithm found to be present in the system.

Assume the subset of the fault vector $\boldsymbol{f}_i$, $i \in [j, ..., k]$ has been determined necessary to estimate by the isolation algorithm. The above general expressions then hold for the entries of the transfer function matrices that relate to f_i, $i \in [j, ..., k]$.

Assume a single fault has been determined present. Then, a single column in $\boldsymbol{H}_{rf}(s)$ needs to be considered. The result for this simplest case can be formulated as follows.

Given the stable residual generator

$$\boldsymbol{r}(s) = \boldsymbol{V}_{ru}(s)\boldsymbol{u}(s) + \boldsymbol{V}_{ry}(s)\boldsymbol{y}(s))$$

and a transfer function model relating this residual to faults

$$\boldsymbol{r}(s) = \boldsymbol{H}_{rf}(s)\boldsymbol{f}(s).$$

Assume that the isolation procedure indicates that fault number i is present, and let the i^{th} column of $\boldsymbol{H}_{rf}(s)$ be

$$\boldsymbol{h}_i(s) = \frac{\bar{\boldsymbol{h}}_i(s)}{\boldsymbol{\eta}(s)},$$

where $\bar{\boldsymbol{h}}_i(s)$ is a polynomial vector with entries $\bar{\boldsymbol{h}}_{ji}(s)$ and $\boldsymbol{\eta}(s)$ is the least common denomination of the entries of $\boldsymbol{h}_i(s)$.

Theorem 6.1 (Single fault estimation)

On the condition that $\boldsymbol{\eta}(s)$ and $\tilde{\boldsymbol{h}}_i(s) = \sum_{j=1}^{l} \bar{\boldsymbol{h}}_{ji}(s)^2$ are stable polynomials, an estimate of $\boldsymbol{f}_i$, $\hat{\boldsymbol{f}}_i$ is given by:

$$\hat{\boldsymbol{f}}_i(s) = \left(\boldsymbol{h}_i'(s)\,\boldsymbol{h}_i(s)\right)^{-1}\ \boldsymbol{h}_i(s)'\boldsymbol{r}(s). \tag{6.73}$$

This estimator is causal when $\deg \boldsymbol{\eta}(s) = \max\ \deg\ \bar{\boldsymbol{h}}_{ji}(s)$. This is easily proved by direct computation of the pseudo inverse in Eq. (6.73):

$$\left(\boldsymbol{h}_i(s)'\,\boldsymbol{h}_i(s)\right)^{-1}\ \boldsymbol{h}_i(s)' = \frac{\eta(s)}{\sum_{i=1}^{l} \bar{\boldsymbol{h}}_{ji}^2(s)} \left(\bar{\boldsymbol{h}}_{1i}(s), ..., \bar{\boldsymbol{h}}_{li}(s)\right). \tag{6.74}$$

If the above condition on the degree is not met, a low pass approximation for the fault estimate can be obtained by multiplying the denomination of Eq. (6.74) by $(\boldsymbol{s}+\alpha)^{\beta}$, where $\alpha \in \mathbb{R}^{+}$and β is chosen so that all entries in Eq. (6.74) are causal.

Example 6.9 *Fault estimation - ship with three output measurements*

Fault estimation following isolation for the ship with three output measurements results from the residual generator obtained in Example 6.7

$$\boldsymbol{H}_{rf}(s) = \begin{pmatrix} -\frac{1}{s} & 1 & 0 \\ -\frac{1}{s} & 0 & 1 \end{pmatrix} = \frac{1}{s}\begin{pmatrix} -1 & s & 0 \\ -1 & 0 & s \end{pmatrix}.$$

Fault 1 isolated: The estimate of fault number 1 is

$$\hat{f}_1 = \frac{s}{2}\left(\boldsymbol{r}_1(s) + \boldsymbol{r}_2(s)\right).$$

Since this filter is not causal, a low-pass filtered approximation for the rate gyro fault is needed, where $\alpha \in \mathbb{R}^+$

$$\hat{f}_1 = \frac{s}{2(s+\alpha)}\left(\boldsymbol{r}_1(s) + \boldsymbol{r}_2(s)\right). \tag{6.75}$$

Fault 2 isolated: The estimate of fault number 2 is

$$\hat{\boldsymbol{f}}_2 = \boldsymbol{r}_1(s).$$

Fault 2 isolated: The estimate of fault number 3 is

$$\hat{\boldsymbol{f}}_3 = \boldsymbol{r}_2(s).$$

It should be noted that an erroneous isolation will give gross errors in the fault estimate.

In an implementation, the above fault estimators would run in parallel. Once a particular fault is isolated, the estimate can be rapidly provided. □

Alternative methods to fault estimation. In cases, where the above algebraic approach to fault estimation fails, asymptotic estimation of faults may be achievable using an observer on an augmented system, where the state is augmented by the fault(s) to be estimated (modelling faults to be constant):

$$\begin{aligned}
\frac{d}{dt}\begin{pmatrix}\boldsymbol{x}\\ \boldsymbol{f}\end{pmatrix} &= \begin{pmatrix}\boldsymbol{A} & \boldsymbol{F}_x\\ \boldsymbol{O} & \boldsymbol{O}\end{pmatrix}\begin{pmatrix}\boldsymbol{x}\\ \boldsymbol{f}\end{pmatrix} + \begin{pmatrix}\boldsymbol{B}\\ 0\end{pmatrix}\boldsymbol{u}(t)\\
\boldsymbol{y}(t) &= \begin{pmatrix}\boldsymbol{C} & \boldsymbol{F}_y\end{pmatrix}\begin{pmatrix}\boldsymbol{x}\\ \boldsymbol{f}\end{pmatrix}.
\end{aligned}$$

A necessary condition for an asymptotically stable estimator to exist is that the pair

$$\left(\begin{pmatrix}\boldsymbol{A} & \boldsymbol{F}_x\\ \boldsymbol{O} & \boldsymbol{O}\end{pmatrix}, \begin{pmatrix}\boldsymbol{C} & \boldsymbol{F}_y\end{pmatrix}\right)$$

is observable. Observer-based methods are covered extensively in the literature (see the bibliographical notes for references).

In summary the procedure for estimating the magnitude of a fault is as follows:

Algorithm 6.3 *Fault estimation*

Given: A model of the supervised process of the form (6.40) and a residual generator of the form (6.42)

Compute:

1. The transfer matrix $\boldsymbol{H}_{rf}(s)$ relating residuals to faults
2. A left inverse to $\boldsymbol{H}_{rf}(s)$
3. An estimator of the form (6.71), possibly after appropriate filtering of the left inverse in order to obtain a causal and stable estimator for all faults.

Result: A causal and stable fault estimator based on the measurements of $\boldsymbol{u}(s)$ and $\boldsymbol{y}(s)$.

6.5 Deterministic model – optimisation-based approach

6.5.1 Problem statement

The above methods were based on algebraic or polynomial manipulations, and relied on the ability to achieve exact decoupling from disturbances and from input to the residual. When this is not possible, the influence $\boldsymbol{d}(t)$ and $\boldsymbol{u}(t)$ have on the residual competes with that generated by faults $\boldsymbol{f}(t)$. If the effects of input and disturbance on the residual are non-zero, we do not obtain

$$(\boldsymbol{V}_{ru}(s) + \boldsymbol{V}_{ry}(s)\boldsymbol{H}_{yu}(s) \quad \boldsymbol{V}_{ry}(s)\boldsymbol{H}_{yd}(s)) \begin{pmatrix} \boldsymbol{u}(s) \\ \boldsymbol{d}(s) \end{pmatrix} = 0 \tag{6.76}$$

for all $\boldsymbol{u}(s)$ and $\boldsymbol{d}(s)$ and

$$\begin{aligned} \boldsymbol{r}(s) &= (\boldsymbol{V}_{ru}(s) + \boldsymbol{V}_{ry}(s)\boldsymbol{H}_{yu}(s) \quad \boldsymbol{V}_{ry}(s)\boldsymbol{H}_{yd}(s)) \begin{pmatrix} \boldsymbol{u}(s) \\ \boldsymbol{d}(s) \end{pmatrix} \\ &\quad + \boldsymbol{V}_{ry}(s)\boldsymbol{H}_{yf}(s)\boldsymbol{f}(s) + \boldsymbol{V}_{ry}(s)\boldsymbol{H}_{yx}(s)\boldsymbol{x}(0) \end{aligned} \tag{6.77}$$

is strictly speaking not a residual generator according to the definition.

The purpose of this section is to find ways to relax the requirement on exact decoupling for the residual generator. Instead, some optimal approximation should be obtained in the sense that the design shall satisfy certain criteria.

The design objectives should be to

1. Provide a sufficient suppression of disturbances $\boldsymbol{d}$ seen from the residual,

2. Maximise the sensitivity $\boldsymbol{r}$ of the residual with respect to all or a to a selected set of faults in $\boldsymbol{f}$.
3. Make the residual signal sufficiently insensitive to variations in the input signal $\boldsymbol{u}$.
4. Provide the designer with tools to enter a specification of the desired performance.

Formulating the design objectives as performance indices will enable a rigorous treatment. From the condition Eq. (6.76), perfect decoupling of disturbance requires

$$\boldsymbol{V}_{ry}(s)\boldsymbol{H}_{yd}(s) = 0.$$

Insensitivity to input requires the model to fulfil

$$\boldsymbol{V}_{ru}(s) + \boldsymbol{V}_{ry}(s)\boldsymbol{H}_{yu}(s) = 0$$

and both are subject to the constraint that sensitivity to faults is not vanishing

$$\boldsymbol{V}_{ry}(s)\boldsymbol{H}_{yf}(s) \neq 0$$

Norms and gains. In order to treat the relaxed condition, it is not required that the right-hand sides are exactly zero, but we wish to obtain minimal values subject to constraints like stable systems and causal realisation of filters. In order to formulate adequate optimisation problems, recall the definitions of the vector norm and the matrix norm induced by a vector norm: Let $\boldsymbol{x} \in \mathrm{I\!R}^n$. Then the vector p-norm of $\boldsymbol{x}$ is

$$|\boldsymbol{x}|_p = \left(\sum_{i=1}^{n} |x_i|^p \right)^{1/p}$$

In particular, when $p = 2, \infty$,

$$|\boldsymbol{x}|_2 = \sqrt{\sum_{i=1}^{n} |x_i|^2}$$

and

$$|\boldsymbol{x}|_\infty = \max_{1 \le i \le n} |x_i| \,.$$

Further, let $\boldsymbol{A} = (a_{ij}) \in \mathrm{I\!R}^{m \times n}$ and $\boldsymbol{x} \in \mathrm{I\!R}^n$. The matrix norm induced by a vector p-norm is defined as

$$|\boldsymbol{A}|_p = \sup_{x \neq 0} \frac{|\boldsymbol{A}\boldsymbol{x}|_p}{|\boldsymbol{x}|_p}$$

In particular, when $p = 2, \infty$,

$$|\boldsymbol{A}|_2 = \sqrt{\lambda_{\max}(\boldsymbol{A}'\boldsymbol{A})} = \bar{\sigma}(A)$$

and

$$|\boldsymbol{A}|_\infty = \max_{1\le i\le m} \sum_{j=1}^{n} |a_{ij}| \qquad \textit{(maximum absolute row sum)},$$

where $\lambda_{\max}$ is the largest eigenvalue, and $\bar{\sigma}$ is the largest singular value.

It is noted that an induced norm can be viewed as a mapping from a vector space $\mathbf{C}^n$ equipped with a norm $|\cdot|_p$ to a vector space $\mathbf{C}^m$ with a norm $|\cdot|_p$. The induced norms have the interpretation of input/output amplification gains.

Let $\boldsymbol{H}(j\omega) \in \mathbf{C}^{m\times n}$ be a stable transfer function, i.e. with all poles strictly in the left half plane. Then the 2-norm is

$$|\boldsymbol{H}|_2 = trace\left(\frac{1}{2\pi}\int_{-\infty}^{\infty} \boldsymbol{H}(j\omega)\,\boldsymbol{H}'(-j\omega)\,d\omega\right)^{\frac{1}{2}}$$

and the ∞ norm

$$|\boldsymbol{H}|_\infty = \max_{\omega}\ \bar{\sigma}\ (\boldsymbol{H}(j\omega))\,.$$

An important result is that

$$|\boldsymbol{H}\boldsymbol{f}|_2^2 \le |\boldsymbol{H}|_\infty^2\,|\boldsymbol{f}|_2^2 = \max_{\omega}\bar{\sigma}(\boldsymbol{H}(j\omega))^2\,|\boldsymbol{f}|_2^2$$

since

$$\begin{aligned} |\boldsymbol{H}\boldsymbol{f}|_2^2 &= \frac{1}{2\pi}\int_{-\infty}^{\infty} \boldsymbol{f}'(-j\omega)\boldsymbol{H}'(-j\omega)\boldsymbol{H}(j\omega)\boldsymbol{f}(j\omega)d\omega \\ &\le |\boldsymbol{H}|_\infty^2\,\frac{1}{2\pi}\int_{-\infty}^{\infty} \boldsymbol{f}'(-j\omega)\boldsymbol{f}(j\omega)d\omega \\ &= |\boldsymbol{H}|_\infty^2\,|\boldsymbol{f}|_2^2\,, \end{aligned}$$

which shows $|\boldsymbol{H}|_\infty^2$ is the upper bound for the signal power transmitted from input to output of the transfer function $\boldsymbol{H}(s)$.

Formulation as an optimisation problem. The first property of a relaxed residual generator should be minimisation of the effect of disturbances in the residual.

A direct minimisation of the effect the disturbance has on the residual is expressed in the induced norm

$$\min_{\boldsymbol{V}_{ry}} \boldsymbol{J}_{id} = \min_{\boldsymbol{V}_{ry}} \frac{|\boldsymbol{V}_{ry}(s)\boldsymbol{H}_{yd}(s)\boldsymbol{d}(s)|_2^2}{|\boldsymbol{d}(s)|_2^2} = \min_{\boldsymbol{V}_{ry}} |\boldsymbol{V}_{ry}(s)\boldsymbol{H}_{yd}(s)|_\infty^2$$

subject to

$$\boldsymbol{V}_{ry}(s)\boldsymbol{H}_{yf}(s) \ne 0.$$

The constraint prevents the trivial solution $\boldsymbol{V}_{ry}(s) = 0$.

The signal power comprised in the residual caused by the disturbance over the power generated by faults should be minimised, hence a feasible index could be

$$\max_{\boldsymbol{V}_{ry}} \boldsymbol{J}_2 = \max_{\boldsymbol{V}_{ry}} \left(\frac{|\boldsymbol{V}_{ry}(s)\boldsymbol{H}_{yf}(s)\boldsymbol{f}(s)|_2^2}{|\boldsymbol{V}_{ry}(s)\boldsymbol{H}_{yd}(s)\boldsymbol{d}(s)|_2^2} \right)_{|\boldsymbol{d}|\neq 0} .$$

The interpretation of this index is to maximise the signal over noise ratio in the residual, using the total power, i.e. over all frequencies. This index cannot be easily optimised. If we, however, make a slight modification to the optimisation criterion, standard tools are available.

As a general tool for optimisation, the standard setup formulation is widely used in robust and optimal control theory and is widely supported by computer aided design tools. Hence it is advantageous to describe the optimisation problem in the standard setup formulation.

Application of the standard methods require a specific formulation of the problem, which is first illustrated using manipulation on the block diagram in Fig. 6.4 for the case, where the objective is to find a polynomial matrix $\boldsymbol{F}(s)$ such that a signal $\boldsymbol{e}(s)$ is insensitive to a disturbance $\boldsymbol{d}(s)$

$$\boldsymbol{e}(s) = (\boldsymbol{H}_{zd}(s) - \boldsymbol{F}(s)\boldsymbol{H}_{yd}(s))\ d(s).$$

6.5.2 Solution using the standard setup formulation

We introduce first the basic notion of the standard estimation setup and the standard estimation problem, which have a direct bearing on design of residual generators.

Definition 6.4 (Standard estimation setup)
Let a system be given by input vector (known and unknown input) $\boldsymbol{d} \in \mathrm{IR}^{n_d}$, state vector $\boldsymbol{x} \in \mathrm{IR}^n$, an auxiliary output $\boldsymbol{z} \in \mathrm{IR}^l$ and measured output vector $\boldsymbol{y} \in \mathrm{IR}^p$ with state-space equation

$$\begin{aligned} \dot{\boldsymbol{x}}(t) &= \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{E}_x\boldsymbol{d}(t) \\ \boldsymbol{z}(t) &= \boldsymbol{C}_z\boldsymbol{x}(t) + \boldsymbol{E}_z\boldsymbol{d}(t) \\ \boldsymbol{y}(t) &= \boldsymbol{C}_y\boldsymbol{x}(t) + \boldsymbol{E}_y\boldsymbol{d}(t) \end{aligned} \tag{6.78}$$

and, ignoring initial conditions, represented in the Laplace domain by

$$\begin{aligned} \boldsymbol{z}(s) &= \boldsymbol{H}_{zd}(s)\boldsymbol{d}(s) \\ \boldsymbol{y}(s) &= \boldsymbol{H}_{yd}(s)\boldsymbol{d}(s), \end{aligned} \tag{6.79}$$

where

$$\boldsymbol{H}_{zd}(s) = \begin{pmatrix} \boldsymbol{A} & \boldsymbol{E}_x \\ \boldsymbol{C}_z & \boldsymbol{E}_z \end{pmatrix} = \boldsymbol{C}_z(s\boldsymbol{I} - \boldsymbol{A})^{-1}\boldsymbol{E}_x + \boldsymbol{E}_z \tag{6.80}$$

$$\boldsymbol{H}_{yd}(s) = \begin{pmatrix} \boldsymbol{A} & \boldsymbol{E}_x \\ \boldsymbol{C}_y & \boldsymbol{E}_y \end{pmatrix} = \boldsymbol{C}_y(s\boldsymbol{I} - \boldsymbol{A})^{-1}\boldsymbol{E}_x + \boldsymbol{E}_y. \tag{6.81}$$

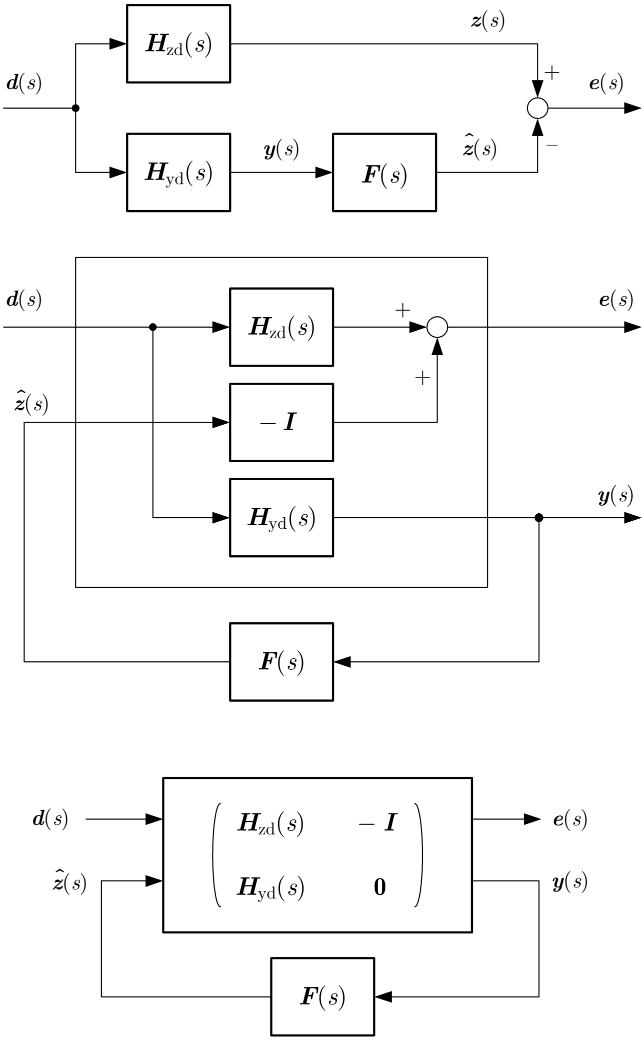


Fig. 6.4. Manipulation of the block diagram to arrive at a standard problem formulation. The upper two diagrams are equivalent, the lower is the representation used to determine $\boldsymbol{F}(s)$ by standard methods.

Problem 6.4 (Standard estimation problem)
Let $\hat{\boldsymbol{z}}(s)$ be an estimate of $\boldsymbol{z}(s)$. Denote the difference by $\boldsymbol{e}_z(s) = \boldsymbol{z}(s) - \hat{\boldsymbol{z}}(s)$. For the system defined by the standard problem setup (6.79), determine a stable transfer function matrix $\boldsymbol{F}(s)$ to provide an estimate of the auxiliary output given the measured output,

$$\hat{\boldsymbol{z}}(s) = \boldsymbol{F}(s)\boldsymbol{y}(s) \tag{6.82}$$

subject to a suitable norm of the estimation error, $|\boldsymbol{e}_z(s)|$ *being less than a chosen gain factor*

$$\sup_{\boldsymbol{F}(s)} |\boldsymbol{e}_z(s)| < \gamma \Leftrightarrow \sup_{\boldsymbol{F}(s)} |\boldsymbol{H}_{zd}(s) - \boldsymbol{F}(s)\boldsymbol{H}_{yd}(s)| < \gamma, \tag{6.83}$$

where the norm can be of types $\mathcal{H}_2$ *or* $\mathcal{H}_\infty$ *for instance.*

Equation (6.83) follows from expanding the estimation error:

$$\begin{aligned} \boldsymbol{e}_z(s) &= \boldsymbol{z}(s) - \hat{\boldsymbol{z}}(s) \\ &= \boldsymbol{z}(s) - \boldsymbol{F}(s)\boldsymbol{y}(s) \\ &= (\boldsymbol{H}_{zd}(s) - \boldsymbol{F}(s)\boldsymbol{H}_{yd}(s))\boldsymbol{d}(s). \end{aligned} \tag{6.84}$$

Remark 6.6
Different filtering and estimation problems can be easily formulated within this general estimation framework. The state can be estimated using $\boldsymbol{C}_z = \boldsymbol{I}_{n,n}$ and $\boldsymbol{E}_z = \boldsymbol{O}$. The input can be estimated using $\boldsymbol{C}_z = \boldsymbol{O}$ and $E_z = I_{l,l}$. The estimation setup will be used later for residual generation. □

Standard $\mathcal{H}_2$ and $\mathcal{H}_\infty$ methods that find the minimum of function according to the selected norm can also be applied to find a suitable $\boldsymbol{F}(s)$ transfer function matrix for the estimation problem. Use of widely available software for this purpose (for example the MATLAB μ toolbox) requires formulation in what is referred to as the *standard system setup and standard problem* in robust control.

Definition 6.5 (Standard system setup)
Let a system be described in the Laplace domain by the transfer function matrix $P(s)$*, and four vectors, input* $\boldsymbol{u}(s) \in \mathbf{C}^m$*, auxiliary input* $\boldsymbol{d}(s) \in \mathbf{C}^{n_d}$*, auxiliary output* $\boldsymbol{e}(s) \in \mathbf{C}^{m_e}$ *and measured output* $\boldsymbol{y}(s) \in \mathbf{C}^p$*. Input and output are related through* $\boldsymbol{P}(s) \in \mathbf{C}^{(p+m_e)\times(m+n_d)}$ *as*

$$\begin{pmatrix} \boldsymbol{e}(s) \\ \boldsymbol{y}(s) \end{pmatrix} = \boldsymbol{P}(s) \begin{pmatrix} \boldsymbol{d}(s) \\ \boldsymbol{u}(s) \end{pmatrix} = \begin{pmatrix} \boldsymbol{P}_{ed}(s) & \boldsymbol{P}_{eu}(s) \\ \boldsymbol{P}_{yd}(s) & \boldsymbol{P}_{yu}(s) \end{pmatrix} \begin{pmatrix} \boldsymbol{d}(s) \\ \boldsymbol{u}(s) \end{pmatrix}$$

Let the transfer function matrix $\boldsymbol{F}(s) \in \mathbf{C}^{m\times p}$ *be a feedback controller for the system, between* $\boldsymbol{y}$ *and* $\boldsymbol{u}$*,*

$$\boldsymbol{u}(s) = \boldsymbol{F}(s)\boldsymbol{y}(s)$$

Using this setup and utilising solutions for two fundamental optimisation problems in the design of residual generators, the $\mathcal{H}_\infty$ sub-optimal control problem and the $\mathcal{H}_2$ sub-optimal control problem.

Problem 6.5 ($\mathcal{H}_\infty$ suboptimal control)
Given a system in form of the standard system setup of Definition 6.5. Design a stabilising controller $\boldsymbol{F}(s)$ *such that the norm of the closed-loop transfer function* $\boldsymbol{T}_{ed}(s)$ *from auxiliary input* $\boldsymbol{d}(s)$ *to auxiliary output* $\boldsymbol{e}(s)$ *is lower than a specified bound* γ*:*

$$\sup_{\boldsymbol{F}} |T_{ed}|_\infty < \gamma \Leftrightarrow \sup_{\boldsymbol{F}(j\omega)} \bar{\sigma}(T_{ed}(j\omega)) < \gamma,$$

where $\bar{\sigma}$ denotes the largest singular value.

The $\mathcal{H}_\infty$ norm gives the maximum sinusoidal gain of the system (energy gain or induced L_2 system gain).

Problem 6.6 **($\mathcal{H}_2$ sub-optimal control problem)**
Given a system in form of the standard system setup in Definition 6.5. Design a stabilizing controller $\boldsymbol{F}(s)$ such that the $\mathcal{H}_2$ norm of the closed-loop transfer function $\boldsymbol{T}_{ed}(s)$ from auxiliary input $\boldsymbol{d}(s)$ to auxiliary output $\boldsymbol{e}(s)$ is minimised.

The standard estimation problem of Fig. 6.4 can be formulated in the standard setup. The generalised system $\boldsymbol{P}(s)$ then takes the form

$$\boldsymbol{P}(s) = \begin{pmatrix} \boldsymbol{H}_{zd}(s) & -\boldsymbol{I} \\ \boldsymbol{H}_{yd}(s) & \boldsymbol{O} \end{pmatrix}.$$

Note that there is no feedback through the system since $\boldsymbol{P}_{yu}(s) = \boldsymbol{O}$.

6.5.3 Residual generation

The above result can be applied in connection with detection, isolation and estimation of faults. We aim at making $\hat{\boldsymbol{z}}(s)$ a residual signal. We investigate two problems. The first is to suppress disturbances as well as possible. The second is to make a balanced optimisation, where the fault signature is preserved in the residual while disturbances are suppressed to the extent possible. Both results follow from appropriate formulation of the standard problem. The strategy is to select an auxiliary output $\boldsymbol{z}(s)$ and give it the properties that the residual should have. This means the formulation of $\boldsymbol{z}(s)$ is directly a specification of the residual. In designing the estimate $\hat{\boldsymbol{z}}(s)$ to track $\boldsymbol{z}(s)$ as closely as possible, according to a given criterion, a sub-optimal estimator is obtained for the ideal residual. The accuracy with which the specification is met is seen in the choice of the optimisation coefficient γ.

The basic residual generator will have the form

$$\boldsymbol{r}(s) = \boldsymbol{F}(s)(\boldsymbol{y}(s) - \boldsymbol{H}_{yu}(s)\boldsymbol{u}(s)). \tag{6.85}$$

The design problem is to determine the operator $\boldsymbol{F}(s)$.

Remark 6.7 *Relation to parity space formulation*
The residual generator Eq. (6.43) had as prerequisite, following from Problem 6.2, that

$$\boldsymbol{V}_{ru} + \boldsymbol{V}_{ry}\boldsymbol{H}_{yu} = 0$$

hence

$$\begin{aligned}\boldsymbol{r}(s) &= \boldsymbol{V}_{ry}(s)\boldsymbol{y}(s) + \boldsymbol{V}_{ru}\boldsymbol{u}(s)\\ &= \boldsymbol{V}_{ry}(s)\boldsymbol{y}(s) - \boldsymbol{V}_{ry}(s)\boldsymbol{H}_{yu}\boldsymbol{u}(s)\\ &= \boldsymbol{V}_{ry}(s)(\boldsymbol{y}(s) - \boldsymbol{H}_{yu}(s)\boldsymbol{u}(s)).\end{aligned}$$

Comparison with Eq. (6.85) shows that finding the solution $\boldsymbol{F}(s)$ in the standard setup is equivalent to determining the operator $\boldsymbol{V}_{ry}(s)$. □

In the design, two requirements have to be combined.

Residual generation with specification on fault sensitivity and disturbance suppression. The goal is now to have the residual replicating a fault through a specified dynamical relation while the disturbance should be suppressed as far as possible. Therefore, we include the fault vector $\boldsymbol{f}(s)$ in the system description and define the auxiliary output $\boldsymbol{z}(s)$ to be dependent only of the fault vector:

$$\begin{aligned}\boldsymbol{y}(s) &= \boldsymbol{H}_{yd}(s)\boldsymbol{d}(s) + \boldsymbol{H}_{yf}(s)\boldsymbol{f}(s)\\ \boldsymbol{z}(s) &= \boldsymbol{H}_{zf}(s)\boldsymbol{f}(s)\\ \hat{\boldsymbol{z}}(s) &= \boldsymbol{V}_{ry}(s)\boldsymbol{y}(s)\\ \boldsymbol{e}_z(s) &= \boldsymbol{z}(s) - \hat{\boldsymbol{z}}(s)\end{aligned}$$

The selection of the auxiliary output reflects directly the properties that the residual should have. $\boldsymbol{H}_{zd} = \boldsymbol{O}$ is chosen because we wish to interpret $\boldsymbol{d}(s)$ as a disturbance and decouple it from the residual. The specification of $\boldsymbol{H}_{zf}(s)$ is a design choice. There may not exist a solution $\boldsymbol{V}_{ry}(s)$ for all arbitrary specifications, so $\boldsymbol{H}_{zf}(s)$ is the key design parameter.

The performance that should be achieved is that the residual follows $\boldsymbol{z}(s)$ as close as possible, hence the relation

$$|\boldsymbol{H}_{zf}(s) - \boldsymbol{V}_{ry}(s)\boldsymbol{H}_{yf}(s)|_{\infty} < \gamma_s$$

should hold, where γ_s characterises the desired fault sensitivity (or tracking) performance. Simultaneously, the effect of the disturbance should be below a certain level, hence

$$|\boldsymbol{V}_{ry}(s)\boldsymbol{H}_{yd}(s)|_{\infty} < \gamma_r,$$

where γ_r is a measure of robustness with respect to input effects. Combining the two, the physically motivated optimisation problem yields:

$$|(-\boldsymbol{V}_{ry}(s)\boldsymbol{H}_{yd}(s)) \quad (\boldsymbol{H}_{zf}(s) - \boldsymbol{V}_{ry}(s)\boldsymbol{H}_{yf}(s))|_{\infty} < \gamma. \tag{6.86}$$

Since

$$\begin{aligned}&\boldsymbol{z}(s) - \boldsymbol{V}_{ry}(s)\boldsymbol{y}(s)\\ &\quad= (\boldsymbol{H}_{zf}(s) - \boldsymbol{V}_{ry}(s)\boldsymbol{H}_{yf}(s))\,\boldsymbol{f}(s) - \boldsymbol{V}_{ry}(s)\boldsymbol{H}_{yd}(s)\boldsymbol{d}(s)\\ &\quad= ((-\boldsymbol{V}_{ry}(s)\boldsymbol{H}_{yd}(s)) \quad (\boldsymbol{H}_{zf}(s) - \boldsymbol{V}_{ry}(s)\boldsymbol{H}_{yf}(s)))\begin{pmatrix}\boldsymbol{d}(s)\\ \boldsymbol{f}(s)\end{pmatrix}\end{aligned}$$

Eq. (6.86) is equivalent to

$$\sup_{\begin{pmatrix} \boldsymbol{d} \\ \boldsymbol{p} \end{pmatrix} \neq \boldsymbol{0}} \frac{|\boldsymbol{z}(j\omega) - \boldsymbol{V}_{ry}(j\omega)\boldsymbol{y}(j\omega)|_2}{\left| \begin{pmatrix} \boldsymbol{d}(j\omega) \\ \boldsymbol{f}(j\omega) \end{pmatrix} \right|_2} < \gamma \Leftrightarrow \sup_{\begin{pmatrix} \boldsymbol{d} \\ \boldsymbol{p} \end{pmatrix} \neq \boldsymbol{0}} \frac{|\boldsymbol{e}_z(j\omega)|_2}{\left| \begin{pmatrix} \boldsymbol{d}(j\omega) \\ \boldsymbol{f}(j\omega) \end{pmatrix} \right|_2} < \gamma.$$

The residual generation design problem is illustrated in Fig. 6.5. The upper diagram in the Figure depicts the residual generator with both specifications $\boldsymbol{H}_{zd}$ and $\boldsymbol{H}_{zf}$ given. In formulating the requirement that disturbance feed-through to the residual should be minimal, $\boldsymbol{H}_{zd} = \boldsymbol{O}$ is specified in the setup shown in the lower part of Fig. 6.5.

In the standard setup, the residual generator design has the following form:

Problem 6.7 (Residual generation with specification on fault sensitivity and disturbance suppression)
Given an LTI system with input $\mathbf{u}(s)$, unknown input (disturbances) $\boldsymbol{d}(s)$ and faults $\boldsymbol{f}(s)$ and let input-output relations of the system be described by $(\boldsymbol{H}_{yu}, \boldsymbol{H}_{yd}, \boldsymbol{H}yf)$. Introduce an auxiliary variable $\boldsymbol{z}(s)$ and specify a transfer function matrix $\boldsymbol{H}_{zf}$ and a real number γ_s. Let $\boldsymbol{z}(s) = \boldsymbol{H}_{zf}\boldsymbol{f}(s)$. Determine $\boldsymbol{V}_{ry}$ such that the maximal deviation between $\hat{\boldsymbol{z}}(s) = \boldsymbol{V}_{ry}\boldsymbol{y}(s)$ and $\boldsymbol{z}(s)$ is bounded by γ_s:

$$|\boldsymbol{z}(s) - \hat{\boldsymbol{z}(s)}| \quad < \quad \gamma_s \tag{6.87}$$

The solution to this problem of residual generation design has the following form:

1. Define the standard problem setup:

$$\begin{aligned} \text{aux.input}: \quad & \boldsymbol{d}(s) \leftarrow \begin{pmatrix} \boldsymbol{d}(s) \\ \boldsymbol{f}(s) \end{pmatrix} \\ \text{input}: \quad & \boldsymbol{u}(s) \leftarrow \boldsymbol{r}(s) \\ \text{aux.output}: \quad & \boldsymbol{e}(s) \leftarrow \boldsymbol{e}_z(s) = \boldsymbol{z}(s) - \boldsymbol{r}(s) \\ \text{output}: \quad & \boldsymbol{y}(s) \leftarrow \boldsymbol{y}(s) \\ & \boldsymbol{P}(s) \leftarrow \begin{pmatrix} \begin{pmatrix} \boldsymbol{O} & \boldsymbol{H}_{zf}(s) \end{pmatrix} & -\boldsymbol{I} \\ \begin{pmatrix} \boldsymbol{H}_{yd}(s) & \boldsymbol{H}_{yf}(s) \end{pmatrix} & \boldsymbol{O} \end{pmatrix} \\ & \boldsymbol{F}(s) \leftarrow \boldsymbol{V}_{ry}(s) \end{aligned}$$

2. Use software that solves the standard problem to determine a solution in form of a stable transfer function $\boldsymbol{V}_{ry}(s)$ that satisfies the inequality

$$\sup_{\begin{pmatrix} \boldsymbol{d} \\ \boldsymbol{f} \end{pmatrix} \neq \boldsymbol{0}} \frac{|\boldsymbol{e}_z|_2}{\left| \begin{pmatrix} \boldsymbol{d} \\ \boldsymbol{f} \end{pmatrix} \right|_2} < \gamma$$

which is equivalent to finding a solution to

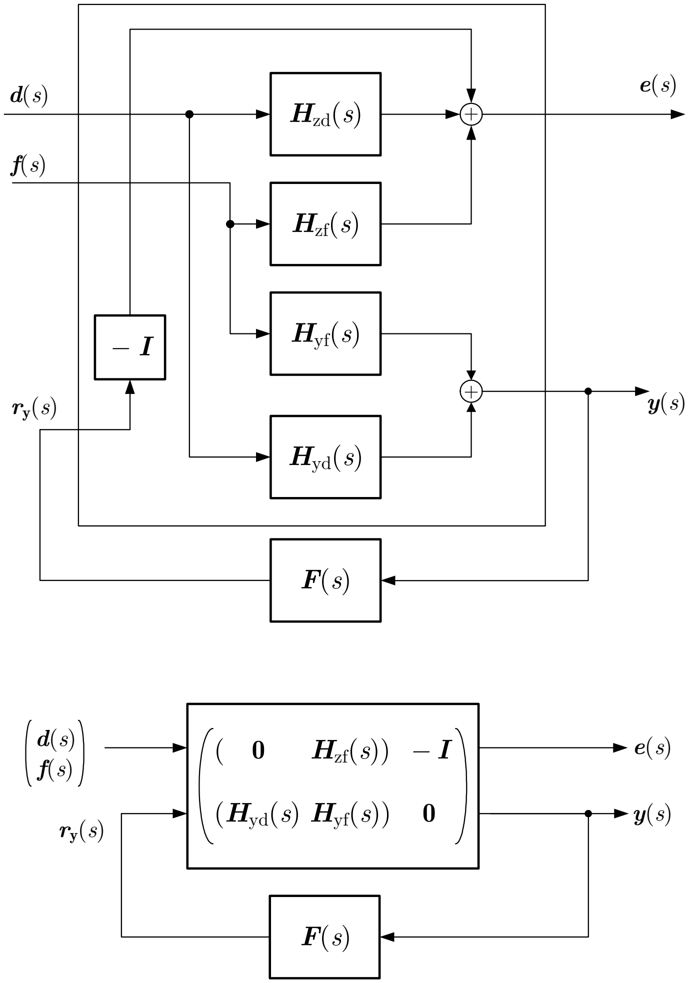


Fig. 6.5. Residual generator depicted in a standard setup formulation with specifications $\boldsymbol{H}_{zf}$ and $\boldsymbol{H}_{zd}$ in the upper part of the figure. $\boldsymbol{H}_{zd}$ is specified as zero in the design problem shown in the lower part of the figure.

$$\left\| \begin{pmatrix} -\boldsymbol{V}_{ry}\boldsymbol{H}_{yd} & \boldsymbol{H}_{zf} - \boldsymbol{V}_{ry}\boldsymbol{H}_{yf} \end{pmatrix} \right\|_{\infty} < \gamma.$$

If a result exists, which is not guaranteed, the result is strong in the sense it provides the residual generator with optimal weighting between suppression of disturbances and specified sensitivity to faults.

In practice it is worthwhile to start a design with investigating the extent to which disturbances can be suppressed using the disturbance suppression problem. When insight in the problem has been gained, continue with supplying a specification to the problem and iterate until a suitable compromise has been found between disturbance suppression and fault tracking.

Fault detection. When the purpose is to design a pure fault detection filter, a sensible way to specify $\boldsymbol{H}_{zf}(s)$ is to require that it is a row vector with nonzero causal and stable entries. When a residual vector is sought the specification becomes:

$$\forall\omega; j\omega \neq z_k : \begin{cases} \text{rank } (\boldsymbol{H}_{zf}(j\omega)) \leq 1 \\ \forall i : h_i(j\omega) \neq 0, \end{cases}$$

where $h_i(j\omega)$ stands for the i^{th} column of $\boldsymbol{H}_{zf}(j\omega)$ and z_k are the zeros of $\boldsymbol{H}_{zf}(s)$.

Fault isolation. If the number of faults to be isolated is $\boldsymbol{n}_f$ and simultaneous faults can occur, $\boldsymbol{H}_{zf}(s)$ has to fulfil the requirement:

$$\text{rank } \boldsymbol{H}_{zf}(s) = \boldsymbol{n}_f,$$

where, as usual, the normal rank is considered.

When simultaneous faults are not considered, a vector $\boldsymbol{z}$ of size l is sufficient to isolate $2^l - 1$ faults by considering suitable coding sets. This translates into the following specification for matrix $\boldsymbol{H}_{zf}(s)$. To isolate $\boldsymbol{n}_f$ faults, choose a matrix $\boldsymbol{H}_{zf}(s)$ such that:

$$\begin{aligned} \text{rank } \boldsymbol{H}_{zf}(s) &\geq \log_2(\boldsymbol{n}_f + 1) \\ \text{rank } (\boldsymbol{h}_i(s)\, \boldsymbol{h}_j(s)) &= 2 \quad \text{with} \quad i = 1, ..., \boldsymbol{n}_f i \neq j \quad j = 1, ..., \boldsymbol{n}_f. \end{aligned}$$

Fault estimation. Fault estimation can be obtained by specifying $\boldsymbol{H}_{zf}(s) = \boldsymbol{I}$. In this case,

$$\boldsymbol{z}(s) = \boldsymbol{I} f(s)$$

and

$$\boldsymbol{e}(s) = \boldsymbol{z}(s) - \hat{\boldsymbol{z}}(s).$$

In the ideal situation, where no disturbance exists, this specification aims at assuring that $\hat{\boldsymbol{z}}$ tracks the fault f by guaranteeing that

$$\sup \frac{|\hat{\boldsymbol{z}} - \boldsymbol{z}|_2}{|f|_2} < \gamma.$$

When disturbances do exist, a trade-off is made between fault tracking and insensitivity of $\hat{\boldsymbol{z}}$ to the disturbance. The block diagram to specify fault estimation from the solution to a standard problem is shown in Fig. 6.6.

Design considerations. In connection with using $\mathcal{H}_2$ or $\mathcal{H}_\infty$ optimisation to design the residual generator, a weight function can further be included in the setup to some

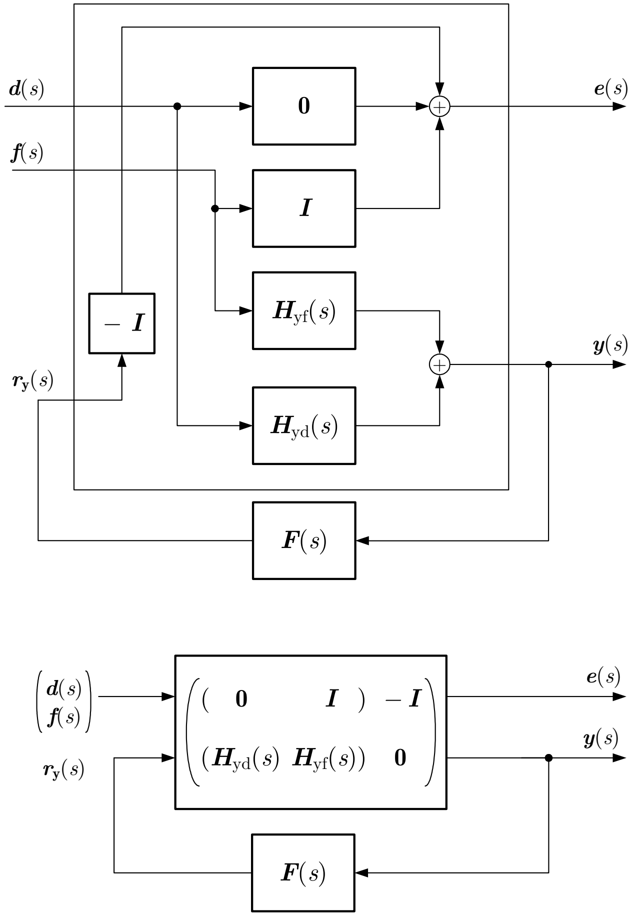


Fig. 6.6. If a solution $\boldsymbol{F}(s)$ exists, fault estimation is obtained by solving the standard problem using the specification $\boldsymbol{H}_{zd} = \boldsymbol{O}$, $\boldsymbol{H}_{zf} = \boldsymbol{I}$.

advantage of the designer. The weight function $\boldsymbol{W}(s)$ can be applied to specify the frequency range(s), where detection, isolation or estimation should be obtained most effectively. The way to include a weight matrix in the design is to modify the $\boldsymbol{P}(s)$ system matrix to

$$\boldsymbol{P}(s) = \begin{pmatrix} \begin{pmatrix} \boldsymbol{O} & \boldsymbol{W}\boldsymbol{H}_{zf} \end{pmatrix} & -\boldsymbol{W} \\ \begin{pmatrix} \boldsymbol{H}_{yd} & \boldsymbol{H}_{yf} \end{pmatrix} & \boldsymbol{O} \end{pmatrix}.$$

The weighting matrix specifies in which frequency ranges a designer emphasises to meet the bound γ and where it can be relaxed.

The main issue in using the standard setup to obtain sub-optimal residual generators of the different classes described above is the selection of a proper specification

$\boldsymbol{H}_{zf}(s)$. Whereas the optimisation itself is left to the software tools available, good results are only obtained if a good specification is provided. An iterative design method has proved useful in practice.

Algorithm 6.4 *Residual generator design*

1. **Formulate problem:** Formulate the relevant version of the standard setup for the problem.
2. **Design specification:** Specify an initial qualified guess on the specification $\boldsymbol{H}_{zf}(s)$. The specification needs to be bounded from below, otherwise, the optimal solution will be $\boldsymbol{F} = \boldsymbol{O}$ and $\boldsymbol{H}_{zf} = \boldsymbol{O}$.
3. **Solve problem:** Find the function $\boldsymbol{F}(s)$ in the residual generator

$$\boldsymbol{r}(s) = \boldsymbol{F}(s)(\boldsymbol{y}(s) - \boldsymbol{H}_{yu}(s)\boldsymbol{u}(s)),$$

   where $\boldsymbol{F}(s)$ is the best obtainable solution to the problem given the specification $\boldsymbol{H}_{zf}(s)$.
4. **Iterate until converged:** Continue until the value of γ obtained has converged.
5. **Iterate in specification:** Based on this residual generator, specify a new $\boldsymbol{H}_{zf}(s)$ and repeat the design.

The procedure usually requires very few iterations.

6.6 Residual evaluation

Given a residual generator for the deterministic case, i.e. there are only insignificant random disturbances or measurement noise, the purpose of this section is to find a method for residual evaluation that will determine whether a fault is present.

Residual - general case. Let a set of residuals obtained from structural analysis have the form $\boldsymbol{r} = (r_1, r_2, ..., r_n)'$. Consider one of these residuals

$$r_j(t) = p_j(k_i, c_i, t) \qquad k_i \in K^{(j)}, c_i \in C^{(j)}, \quad j = 1, ..., n, \tag{6.88}$$

where p_j is of the form in which the constraints in $C^{(j)}$ were formulated: linear, nonlinear, tabular, quantised, logical or hybrid. As it is useful to categorise known variables into the natural categories input u, measured y, and parameters θ, the parity relation is written as

$$r_j(t) = p_j(u_i, y_i, \theta_i, c_i, t) \quad u_i, y_i, \theta_i \in K^{(j)}, c_i \in C^{(j)}, \quad j = 1, ..., n. \tag{6.89}$$

The parity relations implemented for residual generation would not be the true system constraints c_i nor the true parameters θ_i but would be estimates of those, $\hat{c}_i, \hat{\theta}_i$, respectively.

In order to shape the signatures of faults in the residuals or suppress noise, filtering of the raw parity relation Eq. (6.89) will usually take place, and also the filtered version is a residual,

$$r_j(t) = \int_0^t w_j(t-\tau)p_j(u_i, y_i, \hat{\theta}_i, \hat{c}_i, \tau)d\tau, \quad j = 1, ..., n, \tag{6.90}$$

where $w_j(t-\tau)$ is the impulse response of the filter applied to parity relation j.

Further, the vector of residuals could be constructed as a linear combination of the elements from the above residuals, Eq. (6.90),

$$\boldsymbol{r}(t) = \boldsymbol{W} \begin{pmatrix} r_1(t) \\ \vdots \\ r_n(t) \end{pmatrix}, \tag{6.91}$$

where $\mathbf{W} \in \mathrm{IR}^{n\times n}, \det(\mathbf{W}) \neq 0$.

Uncertainty. In real life, $\hat{c}_i \neq c_i$, and $\hat{\theta}_i \neq \theta_i$, hence $r_j(t)$ could be nonzero even though there was no violation of a constraint in relation j, $\forall c_i \in C^{(j)} : c_i = 0$. In particular, actuator demand and disturbances could drive the residual away from zero when parameters and constraints are not exactly equal to those of the real object. In order to make residual evaluation under such uncertainty, it is necessary to accept that a residual can have some deviation from zero even in the no-fault case. However, the effect on $r_j(t)$ has to be bounded, hence $p_j(u_i, y_i, \hat{\theta}_i, \hat{c}_i, t)$ is bounded,

$$\left\| p_j(u_i, y_i, \hat{\theta}_i, \hat{c}_i, t) \right\| \leq \alpha_j(u_i, y_i, t) \wedge 0 < \alpha(u_i, y_i, t) < \infty. \tag{6.92}$$

LTI case. If the object for diagnosis is linear and time-invariant (LTI), the residual generator could be LTI with a frequency representation

$$\mathbf{r}(s) = \mathbf{H}_{ru}(s)\boldsymbol{u}(s) + \mathbf{H}_{rd}(s)\mathbf{d}(s) + \mathbf{H}_{rf}(s)\mathbf{f}(s) \tag{6.93}$$

being an explicit function of input, disturbances and faults.

In an ideal case, residual generation is perfect and we have $\mathbf{H}_{ru}(s) = \boldsymbol{O}$ and $\mathbf{H}_{rd}(s) = \boldsymbol{O}$. Residual evaluation then reduces to investigating the properties of

$$\mathbf{r}(s) = \mathbf{H}_{rf}(s)\mathbf{f}(s). \tag{6.94}$$

In the general case, still with an LTI system, model uncertainty and unmodelled dynamics will give rise to $\mathbf{H}_{ru}(s) \neq \boldsymbol{O}$ and $\mathbf{H}_{rd}(s) \neq \boldsymbol{O}$. Residual evaluation need then be made such that false alarms are avoided from control input $\boldsymbol{u}(t)$ and disturbances $\mathbf{d}(t)$ within the normal range.

6.6.1 Evaluation against a threshold

Validating that no fault is present is equivalent with checking that the residual vector is zero. Validating the presence of a fault means checking whether the residual is or has been different from zero. The two hypotheses and the associated condition on the residual vector are

$$\begin{array}{lll} \mathcal{H}_0(0,t): & \text{no fault is present} & \|\mathbf{r}\| = 0 \\ \mathcal{H}_1(f_j,t_j): & \text{fault } f_j \text{ was present since time } t_j & \|\mathbf{r}(t)\| \neq 0, t \geq t_j, \end{array} \tag{6.95}$$

where $\|\mathbf{r}\|$ is an appropriate norm of the residual.

Test function. For generality, introduce a test function $\varphi(r(t))$, which provides a measure (norm) of the residual's deviation from zero. Some common test functions are the following

- Absolute value

$$\varphi(r_j(t)) = |r_j(t)| \,. \tag{6.96}$$

- An approximation to the two-norm of the residual vector

$$\varphi(r_j(t)) = \left(\frac{1}{T} \int_{t-T}^{t} |r_j(\tau)|^2 \, d\tau \right)^{\frac{1}{2}} . \tag{6.97}$$

- Square root of filtered absolute value, squared,

$$\varphi(r_j(t)) = \left(\int_0^t w_\varphi^{(j)}(t-\tau) \, |r_j(\tau)| \, d\tau \right)^{\frac{1}{2}} . \tag{6.98}$$

- Filtered mean square value of signal

$$\varphi(r_j(t)) = \int_0^t w_\varphi^{(j)}(t-\tau) \left(r_j(\tau) - \frac{1}{T} \int_{\tau-T}^{\tau} r_j(\tau_2) d\tau_2 \right)^2 d\tau, \tag{6.99}$$

  where $w_\varphi^{(j)}(t)$ is the impulse response of a filter used particularly for evaluaton of residual j. In this context, the test function given in Eq. (6.96) is considered further.

Threshold function. The next step in residual evaluation is to determine a threshold function $\Phi(t)$ for evaluation of the test function $\varphi(t)$. $\Phi(t)$ should have the properties

no fault:	$\forall t \geq 0, \; f(t) = 0:$	$\varphi(r(t)) \leq \Phi(t)$
weakly detectable fault:	$\exists t \geq t_0 : f(t) \neq 0:$	$\varphi(r(t)) > \Phi(t)$
strongly detectable fault:	$\forall t \geq t_1 \geq t_0 : f(t) \neq 0, \; t \geq t_0:$	$\varphi(r(t)) > \Phi(t)$

LTI case. In the ideal LTI case, Eq. (6.94), $\Phi(t)$ could be chosen constant and as close to zero as allowed by practical values of bias and noise in the residual.

In the non-ideal case, Eq. (6.93) applies and input and disturbances have some feed-through to the residual. With the test function $\varphi(t) = \|r_j(t)\|_2$, the threshold need be determined such that

$$\Phi_j(t) \geq \sup_{f=0,\|u,d\|<\varepsilon} (\varphi(r_j(t)))$$

is achieved in the time domain. The fact that total power calculated in the time domain and in the frequency domain are equal is used to determine the theshold function,

$$||r(j\omega)||^2 = \frac{1}{2\pi}\int_0^\infty r(j\omega)r(-j\omega)d\omega = \lim_{T\to\infty}\int_0^{t'} |r(t)|^2 dt = ||r(t)||^2.$$

From Eq. (6.93), the residual is given in the frequency domain. Component j of the residual is

$$r_j(s) = (\mathbf{H}_{ru}(s)\boldsymbol{u}(s))^{(j)} + (\mathbf{H}_{rd}(s)\mathbf{d}(s))^{(j)} + (\mathbf{H}_{rf}(s)\mathbf{f}(s))^{(j)}.$$

With k control inputs

$$\begin{aligned} \|r_j(j\omega)\|_2 &\leq \|\mathbf{H}_{ru}(j\omega)\boldsymbol{u}(j\omega)\|_2^{(j)} + \|\mathbf{H}_{rd}(j\omega)\mathbf{d}(j\omega)\|_2^{(j)} \\ &\leq \sum_{i=1}^{k} \|\mathbf{H}_{ru}(j\omega)\|_\infty^{(ji)} \|u_i(j\omega)\|_2 + \|\mathbf{H}_{rd}(j\omega)\mathbf{d}(j\omega)\|_\infty^{(j)} \end{aligned} \tag{6.100}$$

for all admissible $\boldsymbol{u}$ and $\mathbf{d}$. The first term in the right hand side is a gain times input power. The second is the maximal contribution to the residual from disturbances.

Let the effect of disturbances on the residual be bounded by

$$\|\mathbf{H}_{rd}(j\omega)\mathbf{d}(j\omega)\|_\infty^{(j)} < \beta_d^{(j)}, \tag{6.101}$$

then $\Phi(t)$ should be chosen as the time-varying function

$$\begin{aligned} \Phi_j(t) &= \sum_{i=1}^{k} \beta_i \|u_i(t)\|_2 + \beta_d^{(j)} \\ \beta_i &= \|\mathbf{H}_{ru}(j\omega)\|_\infty^{(ji)}. \end{aligned} \tag{6.102}$$

This threshold is a function of maximal gains from control inputs to residual and of the maximum gain from disturbances to residual. It is often referred to as a time-varying threshold in the literature. The term adaptive threshold has also been used.

If the time-varying threshold Eq. (6.102) is too conservative, a dynamic bound could be specified as

$$\Phi_j(t) = \sum_{i=1}^{k}\left(\int_0^t \hat{h}_{ru}^{(ji)}(t-\tau)u_i(\tau)d\tau\right) + \varepsilon_d, \tag{6.103}$$

where $\hat{h}_{ru}^{(ji)}$ is an estimate of the maximum (envelope) of impulse response functions from input i to residual j for a given model uncertainty.

Example 6.10 *Ship example (LTI case)*

Assume the ship was LTI,

$$\begin{aligned} y_1(s) &= \omega_3(s) + \omega_w(s) + f_\omega(s) = \frac{b}{s-b\eta_1}\delta(s) + \omega_w(s) + f_\omega(s) \\ y_2(s) &= \psi(s) + f_\psi(s) = \frac{1}{s}(\omega_3(s) + \omega_w(s)) + f_\psi(s) \end{aligned} \tag{6.104}$$

the design model was

$$\begin{aligned}\hat{\omega}_3(s) &= \frac{\hat{b}}{s-\hat{b}\hat{\eta}_1}\delta \\ \hat{\psi}(s) &= \frac{1}{s}\hat{\omega}_3(s)\end{aligned}$$

and a residual generator is chosen as

$$\begin{aligned}r_1(s) &= y_1(s)-\hat{\omega}_3(s) = \left(\frac{b}{s-b\eta_1}-\frac{\hat{b}}{s-\hat{b}\hat{\eta}_1}\right)\delta(s)+\omega_w(s)+f_\omega(s) \\ r_2(s) &= \frac{\tau}{1+s\tau}\left(sy_2(s)-y_1(s)\right) = \frac{s\tau}{1+s\tau}f_\psi(s)-\frac{\tau}{1+s\tau}f_\omega(s).\end{aligned}$$

With no faults

$$\|r_1(j\omega)\|_2 \leq \left\|\frac{b}{j\omega-b\eta_1}-\frac{\hat{b}}{j\omega-\hat{b}\hat{\eta}_1}\right\|_\infty \|\delta(j\omega)\|_2+\|\omega_w(j\omega)\|_\infty$$

and

$$\begin{aligned}\|r_1(j\omega)\|_2 &\leq \left\|\frac{b}{j\omega+b\eta_1}-\frac{\hat{b}}{j\omega+\hat{b}\hat{\eta}_1}\right\|_\infty \|\delta(t)\|_2+\|\omega_w(j\omega)\|_\infty \quad (6.105)\\ \Phi_1(t) &= \left\|\frac{b}{j\omega-b\eta_1}-\frac{\hat{j\omega}}{j\omega-\hat{b}\hat{\eta}_1}\right\|_\infty \|\delta(t)\|_2+\|\omega_w(j\omega)\|_\infty \quad (6.106)\\ &\quad \text{with } \|\omega_w(j\omega)\|_\infty \leq \beta_d, \\ \Phi_1(t) &= \left|\frac{\hat{\eta}_1-\eta_1}{\eta_1\hat{\eta}_1}\right| \|\delta(t)\|_2+\beta_d = \beta_u\|\delta(t)\|_2+\beta_d. \quad (6.107)\end{aligned}$$

In real time, we evaluate

$$\|r(t)\|_2 \leq \beta_u\|\delta(t)\|_2+\beta_d \quad (6.108)$$

using Eq. (6.97) as an approximation to the two norm. □

General case. In the general case, if the parity relation is bounded by

$$\varphi(p_j(u_i,y_i,\hat{\theta}_i,\hat{c}_i,t)) \leq \alpha_j(u_i,y_i,t) \wedge 0 < \alpha(u_i,y_i,t) < \infty. \quad (6.109)$$

The threshold function can obviously be chosen as

$$\Phi_j(t) \geq \alpha_j(u_i,y_i,t) \quad (6.110)$$

If more detailed information is available, e.g.

$$\alpha_j(u_i,y_i,t) \leq \beta_0+\sum_{i=1}^{k}\beta_{ji}|u_i| \quad (6.111)$$

such information should be utilised when specifying the threshold, in this case as

$$\Phi_j(t) = \beta_0+\sum_{i=1}^{k}\beta_{ji}|u_i|. \quad (6.112)$$

Return to normal. The above procedure tested for the change H_0 to H_1. When a fault has been detected, $\varphi(r_j(t)) \geq \Phi_j(t) \Longrightarrow H^{(j)} = H_1$, change to normal is usually made with a hysteresis, $\gamma : \varphi(r_j(t)) < \gamma\Phi_j(t) \Longrightarrow H^{(j)} = H_0$. A common choice of hysteresis is $\gamma \subset [0.5, 0.8]$.

If a fault is only weakly detectable in residual j, but strongly detectable in other residuals, $\forall j : \varphi(r_j(t)) < \gamma\Phi_j(t) \Longrightarrow H^{(j)} = H_0$ should be used.

It is obvious that simulation and tests in the real environment some engineering judgement need be employed before good choices can be made of the timevarying threshold function $\Phi_j(t)$ and of the hysteresis γ.

This leads to algorithms for deterministic change detection,

Algorithm 6.5 *Test against time-varying threshold*

Given: A residual $r_j = p_j(u_i, y_i, \hat{\theta}_i, \hat{c}_i, t)$ and the object for diagnosis assumed in the no-fault condition.

1. Determine a test function $\varphi(r(t))$ according to Eqs. (6.96) to (6.99).

2. Determine a threshold function $\Phi_j(t)$:for the LTI case according to Eq. (6.102), for the general case according to Eq. (6.109) or Eq. (6.102) when scecific information is available.

Iniitialise: $H^{(j)} = H_0$.

Do:

1. Calculate $\varphi(r_j(t))$ and $\Phi_j(t)$.

2. If $H^{(j)} = H_0, \forall j$:.

If $\varphi(r_j(t)) \geq \Phi_j(t)$ set hypothesis to $H^{(j)} = H_1$.

Else:

If $\varphi(r_j(t)) < \gamma\ \Phi_j(t)$ for $\forall j$ set hypothesis to $H^{(j)} = H_1$.

Example 6.11 *Time-varying threshold for ship*

Let the ship's true constraints be:

$$\begin{aligned} c_1 &: & \dot{\omega}_3 &= & &b\eta_1\omega_3 + b\eta_3\omega_3^3 + b\delta \\ c_2 &: & \dot{\psi} &= & &\omega_3 + \omega_w \\ m_1 &: & y_1 &= & &\dot{\psi} \\ m_2 &: & y_2 &= & &\psi \end{aligned} \tag{6.113}$$

And let a model used for design be

$$\begin{aligned} \hat{c}_1 &: & \dot{\omega}_3 &= & &\hat{b}\hat{\eta}_1\omega_3 + \hat{b}\delta \\ \hat{c}_2 &: & \dot{\psi} &= & &\omega_3 \\ m_1 &: & y_1 &= & &\dot{\psi} \\ m_2 &: & y_2 &= & &\psi \end{aligned} \tag{6.114}$$

Using the model for design, a residual generator is suggested as

$$\begin{aligned} r_1 = &\quad \tfrac{d}{dt} y_1 - \tfrac{d}{dt} \hat{y}_1 \\ r_2 = &\quad \tfrac{d}{dt} y_2 - y_1 \end{aligned} \tag{6.115}$$

then, the real residual will vary with input and

$$\begin{aligned} r_1(t) = &\quad (b\eta_1 - \hat{b}\hat{\eta}_1)\omega_3 + b\eta_3\omega_3^3 + (b - \hat{b})\delta(t) + \tfrac{d}{dt}\omega_w(t) \\ r_2(t) = &\quad 0 \end{aligned}$$

$$\begin{aligned} |r_1(t)| &\leq \left| b\eta_1 - \hat{b}\hat{\eta}_1 \right| |y_1| + |b\eta_3| \left| y_1^3 \right| + \left| b - \hat{b} \right| |\delta| + \left| \frac{d}{dt}\omega_w(t) \right|_{\text{sup}} \\ &\leq \beta_1 |y_1| + \beta_3 \left| y_1^3 \right| + \alpha_1 |\delta| + \beta_d \leq \alpha_2 |\delta| + \beta_d. \ \square \end{aligned}$$

6.7 Stochastic model – change detection algorithms

6.7.1 Introduction

To be able to solve the problem of fault detection, isolation and/or estimation that will be stated precisely in a stochastic framework below, a prerequisite is needed on sequential change detection algorithms. It is known that fault detection (and isolation or estimation) systems are made of two parts, a residual generator and a decision system. When the residual generator is designed on the basis of a linear stochastic model, residual evaluation reduces to the problem of detecting a change in the mean of a normally distributed random sequence, which can be achieved by sequential change detection algorithms. Therefore, this topic is considered before addressing successively fault detection, isolation and estimation in the case of additive faults.

6.7.2 Sequential change detection: the scalar case

Introduction. The sequential change detection algorithms are first derived in the simple case of processing a sequence of independent random variables with probability density function depending on a scalar parameter θ. The situation where θ is the mean of a Gaussian distribution is used to illustrate the theory, since this is the problem encountered for residual evaluation. As the sequential algorithms will be used to process residuals, the above theory has to be generalised to be able to detect changes in the mean of sequences of Gaussian vectors, which is done in a subsequent paragraph.

Problem statement. Consider a sequence of independent random variables $z(i)$, $i = 1, 2, \ldots,$ with probability density function $p_\theta(z)$ depending upon one scalar parameter θ. Before an unknown change time, k_0, θ is equal to θ_0. At time k_0, it changes to $\theta = \theta_1 \neq \theta_0$. The problem is to detect the change time, and possibly estimate the value of the change in the parameter. No a priori knowledge of the distribution of the change time is assumed. θ_0 is known by hypothesis, and two situations are considered for θ_1, namely θ_1 known and θ_1 unknown. The first case yields the so-called cumulative sum (CUSUM) algorithm, the second the so-called generalised likelihood ratio (GLR) algorithm.

Both algorithms rely on a fundamental concept, namely the log-likelihood ratio of an observation z, which is defined as:

$$s(z) = \ln \frac{p_{\theta_1}(z)}{p_{\theta_0}(z)}. \tag{6.116}$$

The name comes from the fact that the likelihood function of the observation z is by definition equal to the probability density $p_\theta(z)$ of the underlying random variable evaluated at z. The likelihood function is thus a deterministic function of θ.

The log-likelihood ratio has the following fundamental statistical property:

$$E_{\theta_0}(s) = \int_{-\infty}^{\infty} s(z)\, p_{\theta_0}(z) dz < 0, \tag{6.117}$$

$$E_{\theta_1}(s) = \int_{-\infty}^{\infty} s(z)\, p_{\theta_1}(z) dz > 0. \tag{6.118}$$

E_{θ_0} (E_{θ_1}) denotes expectation of $s(z)$ under the distribution associated to $p_{\theta_0}(z)$ ($p_{\theta_1}(z)$). This property can be easily understood from the following example. Assume that $p_\theta(z)$ is a Gaussian distribution and that the parameter θ is the mean of this distribution, which will be denoted μ.

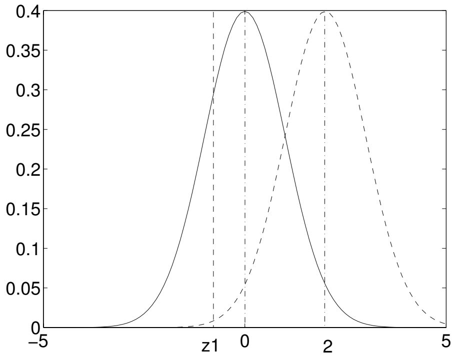


Fig. 6.7. Two Gaussian probability density functions with mean $\mu_0 = 0$ and $\mu_1 = 2$, and with the same variance $\sigma^2 = 1$

Consider Fig. 6.7. When the random variable z has $p_{\mu_0}(z)$ ($p_{\mu_1}(z)$) as probability density function, its realisations are most often in the "neighbourhood" of μ_0 (μ_1). Take the realisation z_1 for instance. Clearly $\frac{p_{\mu_1}(z_1)}{p_{\mu_0}(z_1)} < 1$. As z_1 is most probably obtained when the random variable z has $p_{\mu_0}(z)$ as probability density function, this illustrates that the logarithm of $\frac{p_{\mu_1}(z)}{p_{\mu_0}(z)}$ is on the average negative when z has

$p_{\mu_0}(z)$ as probability density function. The property described by (6.117), (6.118) is exploited in the next section to provide an intuitive derivation of the CUSUM algorithm.

Derivation of the CUSUM algorithm. The problem stated in the previous section, with θ_1 known, is addressed. Consider the cumulative sum:

$$\text{Cumulative sum:} \quad S(k) = \sum_{i=1}^{k} s(z(i)) = \sum_{i=1}^{k} \ln \frac{p_{\theta_1}(z(i))}{p_{\theta_0}(z(i))}. \tag{6.119}$$

In this expression, and in the remaining part of this section, k denotes the present time instant. From (6.117) and (6.118), $S(k)$ is expected to exhibit a negative drift before change, and a positive drift after change. This is illustrated in Fig. 6.8 and 6.9.

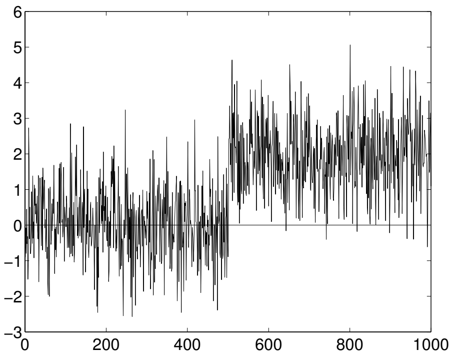


Fig. 6.8. Realisation of a sequence of independent random variables with distributions depicted in Fig. 6.7. Time on the x-axis is expressed in number of samples.

In Fig. 6.8, a realisation of a sequence of independent random variables with distribution $p_{\theta_0}(z)$ before $k = 500$ and $p_{\theta_1}(z)$ after $k = 500$ is depicted. Here $p_{\theta_0}(z)$ and $p_{\theta_1}(z)$ correspond to the Gaussian distributions $p_{\mu_0}(z)$ and $p_{\mu_1}(z)$ of Fig. 6.7. Figure 6.9 gives the evolution of $S(k)$, which behaves as expected. The difference between $S(k)$ and the minimum value of $S(j)$, $1 \leq j \leq k$ yields relevant information on the change. Hence the following decision function $g(k)$ is considered

$$g(k) = S(k) - m(k) \tag{6.120}$$

with $m(k) = \min_{1 \leq j \leq k} S(j)$. The stopping time (also called alarm time), k_a is the time instant at which $g(k)$ crosses a user defined positive threshold h. The fault occurrence time, k_0, can be estimated as the time instant $\hat{k}_0$ at which $S(k)$ has changed from negative to positive slope. It is formally expressed by

$$\hat{k}_0 = \text{argmin}_{\,1 \leq j \leq k_a} S(j).$$

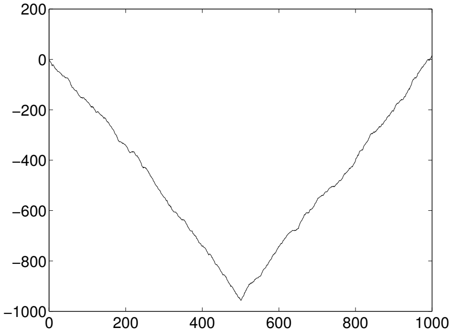


Fig. 6.9. Evolution of $S(k)$ for the sequence of Fig. 6.8, as a function of time in number of samples

The expression of the cumulative sum (6.119) can easily be computed for the distributions considered in Fig. 6.7, as shown in the example below.

Example 6.12 *Change in the mean of a Gaussian sequence*

Remember that the Gaussian probability density function for a random variable with mean μ and variance σ is

$$p_\mu(z) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(z-\mu)^2}{2\sigma^2}\right). \tag{6.121}$$

The resulting likelihood ratio for detecting a change in the mean from μ_0 to μ_1 is

$$\frac{p_{\mu_1}(z)}{p_{\mu_0}(z)} = \exp\left(-\frac{(z-\mu_1)^2}{2\sigma^2} + \frac{(z-\mu_0)^2}{2\sigma^2}\right).$$

Hence straightforward computations yield the following expression for the log-likelihood ratio $s(z)$:

$$s(z) = \frac{2(\mu_1-\mu_0)z + (\mu_0^2 - \mu_1^2)}{2\sigma^2} = \frac{\mu_1-\mu_0}{\sigma^2}\left(z - \frac{\mu_0+\mu_1}{2}\right). \tag{6.122}$$

Figure 6.9 has been obtained by substituting (6.122) (with $z = z(i)$) for $s(z(i))$ in (6.119), which yields the algorithm depicted in the block diagram of Fig. 6.10. Notice that the signal to noise ratio $\frac{\mu_1-\mu_0}{\sigma^2}$ appears in (6.122), and it is thus automatically accounted for in the testing procedure. □

Example 6.13 *Change in the mean and variance*

If both mean and variance change after a fault, the following relation

$$\frac{p_{\mu_1}(z)}{p_{\mu_0}(z)} = \frac{\sigma_0}{\sigma_1}\exp\left(-\frac{(z-\mu_1)^2}{2\sigma_1^2} + \frac{(z-\mu_0)^2}{2\sigma_0^2}\right)$$

holds and the log-likelihood ratio is

$$s(z) = \ln\frac{\sigma_0}{\sigma_1} + \frac{(z-\mu_0)^2}{2\sigma_0^2} - \frac{(z-\mu_1)^2}{2\sigma_1^2}.$$

This general case is shown in the literature but not considered further in this context. □

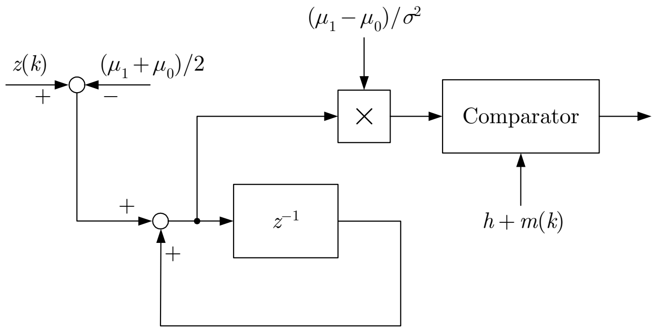


Fig. 6.10. Block diagram for the CUSUM test (6.119), (6.120), (6.122)

Remark 6.8 *Mean and variance of cumulative sum increments*
The mean μ_s, and the variance σ_s^2 of the cumulative sum increments (6.122) will be needed at a latter stage. They can be computed in a straightforward way from (6.122). Indeed,

when $\mu = \mu_0,\ E_{\mu_0}(z) = \mu_0$ and $\mu_s = E_{\mu_0}(s(z)) = -\dfrac{(\mu_1 - \mu_0)^2}{2\sigma^2}$,

when $\mu = \mu_1,\ E_{\mu_1}(z) = \mu_1$ and $\mu_s = E_{\mu_1}(s(z)) = \dfrac{(\mu_1 - \mu_0)^2}{2\sigma^2}$,

$$\sigma_s^2 = \frac{(\mu_1 - \mu_0)^2}{\sigma^2}. \quad \square$$

A more formal derivation of the CUSUM algorithm which is helpful for the subsequent description of the GLR algorithm is now presented. It is called the off-line statistical derivation, and it is based on the following re-formulation of the problem.

Problem 6.8 (Off-line statistical formulation) *Consider the sequence of independent random variables $z(1), \ldots, z(k)$ with probability density function $p_\theta(z)$ depending on one scalar parameter θ. Choose at time instant k between the hypotheses:*

$\mathcal{H}_0$: $\theta = \theta_0$ for $1 \le i \le k$.
$\mathcal{H}_1$: $\theta = \theta_0$ for $1 \le i \le k_0 - 1$ and $\theta = \theta_1$ for $k_0 \le i \le k$, where the time instant k_0 is unknown.

From classical results in hypothesis testing due to Neyman and Pearson, it is known that tests to decide between $\mathcal{H}_0$ and $\mathcal{H}_1$ that are optimal in some sense are based on the log-likelihood ratio between both hypotheses. As k_0 is unknown, let j

be an hypothetical change time. The log-likelihood ratio between $\mathcal{H}_0$ and $\mathcal{H}_1$ with $k_0 = j$ is given as

$$\Lambda_1^k(j) = \frac{\prod_{i=1}^{j-1} p_{\theta_0}(z(i)) \prod_{i=j}^{k} p_{\theta_1}(z(i))}{\prod_{i=1}^{k} p_{\theta_0}(z(i))}. \tag{6.123}$$

The independence between the random variables $z(i)$, $i = 1, \ldots k$ was used to express $\Lambda_1^k(j)$ in terms of the marginal probability density function $p_\theta\,(z(i))$. From (6.123), the following cumulative sum of log-likelihood ratios is deduced:

$$S_j^k = \sum_{i=j}^{k} \ln \frac{p_{\theta_1}(z(i))}{p_{\theta_0}(z(i))}. \tag{6.124}$$

As the change time is unknown, the standard statistical approach consists in replacing it by its maximum likelihood estimate, namely, in looking for the value of j that maximises the numerator in (6.123). This is also the value of j that maximises (6.124). The log-likelihood ratio between $\mathcal{H}_0$ and $\mathcal{H}_1$ is thus estimated by $\max_{1 \le j \le k} S_j^k$. The result due to Neyman and Pearson invoked above actually states that the optimal decision function for Problem 6.8 is

$$g(k) = \max_{1 \le j \le k} \sum_{i=j}^{k} \ln \frac{p_{\theta_1}(z(i))}{p_{\theta_0}(z(i))} = \max_{1 \le j \le k} \sum_{i=j}^{k} s(i) \tag{6.125}$$

and the optimal test consists of the following decisions.

$$\begin{array}{lll} \text{if} \quad g(k) \le h & \text{accept} & \mathcal{H}_0 \\ \text{if} \quad g(k) > h & \text{accept} & \mathcal{H}_1. \end{array} \tag{6.126}$$

The way optimality is understood here involves several concepts. The reader should consult the reference section for precisions on this topic.

When $\mathcal{H}_1$ is accepted, an estimate of the change time is provided by:

$$\hat{k}_0 = \text{argmax}_{\ 1 \le j \le k_a} S_j^k,$$

where k_a is the alarm time, namely the value of k for which $g(k)$ crosses the threshold h.

The decision functions (6.120) and (6.125) are identical. Indeed, with reference to Fig. 6.9, $\sum_{i=j}^{k} \ln \frac{p_{\theta_1}(z(i))}{p_{\theta_0}(z(i)}$ is maximum when all the successive likelihood ratios which correspond to a positive slope on average are considered. This is precisely the way (6.120) was obtained.

An efficient way to implement the CUSUM algorithm is to use its recursive form. From (6.120) and Fig. 6.9 or from (6.125), and from the fact that the threshold h is positive, it is seen that only the contributions to the cumulative sum that add up to a positive number must be taken into account in order to determine the decision function. It justifies the following recursive computation of this function:

$$g(k) = \max(0,\, g(k-1) + s(z(k))). \tag{6.127}$$

To obtain an estimate of the fault occurrence time, the number of successive observations for which the decision function remains strictly positive is computed as:

$$N(k) = N(k-1)\, 1_{\{g(k-1)>0\}} + 1, \tag{6.128}$$

where $1_{\{x\}}$ is the indicator of event x, namely $1_{\{x\}} = 1$ when x is true, and $1_{\{x\}} = 0$ otherwise. An estimate for the fault occurrence time is then given as

$$\hat{k}_0 = k_a - N(k_a), \tag{6.129}$$

where k_a is the stopping or alarm time.

Example 6.12 (cont.) *Change in the mean of a Gaussian sequence*

Considering again the detection of a change in the mean of a Gaussian sequence, (6.127) together with (6.122) yields:

$$g(k) = \max(0,\, g(k-1) + \frac{\mu_1 - \mu_0}{\sigma^2}\left((z(k) - \frac{\mu_0 + \mu_1}{2})\right)) \tag{6.130}$$

which must be introduced in the decision logic (6.126). □

Remark 6.9 *Two-sided CUSUM algorithm*

Quite often, both positive and negative changes in the mean of a Gaussian sequence with mean μ_0 and variance σ^2 have to be detected. Letting β denote the magnitude of this change, the following two-sided CUSUM algorithm can be used for this purpose.

$$g^+(k) = \max\left(0,\, g^+(k-1) + z(k) - \mu_0 - \frac{\beta}{2}\right) \tag{6.131}$$

$$g^-(k) = \max\left(0,\, g^-(k-1) - z(k) + \mu_0 - \frac{\beta}{2}\right). \tag{6.132}$$

An alarm is generated when either $g^+(k)$ or $g^-(k)$ reaches the threshold $\bar{h} = h\sigma^2/\beta$. Notice that the factor $\frac{\mu_1 - \mu_0}{\sigma^2}$ that appears in (6.130) has been cancelled from the decision functions $g^+(k)$ and $g^-(k)$ in (6.131) and 6.132). Equivalently, it is now included in the threshold $\bar{h}$. The expression for $g^-(k)$ is deduced from (6.130) by looking for a positive change in the mean of the sequence $-z(i), i = 1, 2, \ldots$. □

Parameter tuning for the CUSUM algorithm. In this section, the focus is on the case of a change in the mean, μ, of a Gaussian sequence.

Normally, the data associated to hypothesis $\mathcal{H}_0$ correspond to a fault free working mode. Hence parameter μ_0 can be estimated from a set of experimental data obtained in the absence of fault by taking the empirical mean of these data. The variance σ^2 can also be estimated in this way. The estimates are denoted $\hat{\mu}_0$ or $\hat{\sigma}^2$, respectively.

There are thus two design parameter left in the CUSUM algorithm, h and μ_1. Indeed, although the algorithm was derived under the hypothesis that μ_1 is known, this is seldom the case in practice. Nevertheless, the algorithm proved to be useful even when μ_1 is replaced by an approximate value.

The choice of the threshold h results from a compromise between the mean delay for detection and the mean time between false alarms. Exact computation of these quantities is involved, but approximate expressions and bounds are available. Both quantities can be determined from the so-called average run length (ARL) function defined as

$$L(\mu) = E_\mu (k_a)$$

for the example of the detection of a change in the mean of a Gaussian sequence with variance σ^2. The ARL function is thus the expected value of the alarm time instant when the data are distributed according to the normal probability density function with mean μ and variance σ^2. It is a function of the mean μ. When $\mu = \mu_0$ (data recorded in healthy conditions), the value of the ARL function $L(\mu_0)$ is equal to the mean time between false alarms, $\bar{T}$. On the other hand, $L(\mu_1)$ gives the mean delay for detection. An approximation for the ARL function in the situation were changes in the mean of a Gaussian sequence have to be detected is given by the following expression ([5], page 219)

$$\hat{L}(\mu_s) = \left(\exp\left[-2\left(\frac{\mu_s h}{\sigma_s^2} + 1.166\frac{\mu_s}{\sigma_s}\right)\right] - 1 + 2\left(\frac{\mu_s h}{\sigma_s^2} + 1.166\frac{\mu_s}{\sigma_s}\right)\right)\left(\frac{\sigma_s^2}{2\mu_s^2}\right) \tag{6.133}$$

for $\mu_s \neq 0$, where μ_s and σ_s are the mean or the standard deviation of the increments of the cumulative sum, respectively, as computed in Remark 6.8.

The mean time for detection, $\bar{\tau}$ can be estimated as

$$\hat{\bar{\tau}} = \hat{L}\left(\frac{(\mu_1 - \mu_0)^2}{2\sigma^2}\right) = \hat{L}\left(\frac{\beta^2}{2\sigma^2}\right), \tag{6.134}$$

where $\beta = \mu_1 - \mu_0$ and the estimated mean time between false alarms is obtained as

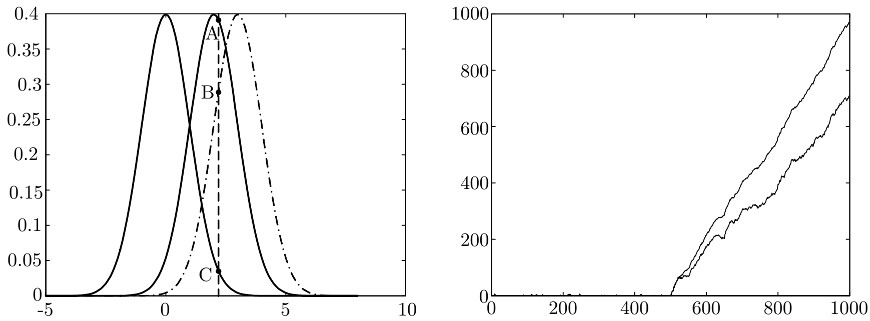
$$\hat{\bar{T}} = \hat{L}\left(-\frac{(\mu_1 - \mu_0)^2}{2\sigma^2}\right) = \hat{L}\left(-\frac{\beta^2}{2\sigma^2}\right). \tag{6.135}$$

When μ_1 (or equivalently β) is not known, it can be replaced by a user specified value such as the most likely magnitude of the change. A simple way to determine the test threshold from (6.134) or (6.135) is to plot $\hat{\bar{\tau}}$ and $\hat{\bar{T}}$ as a function of h. To this end, considering (6.134) for instance, $\frac{(\mu_1-\hat{\mu}_0)^2}{2\hat{\sigma}^2}$ is substituted for μ_s and $\frac{(\mu_1-\hat{\mu}_0)^2}{\hat{\sigma}^2}$ is substituted for σ_s in (6.133). Knowing the desired value for $\hat{\bar{\tau}}$ or $\hat{\bar{T}}$, one then determines from the plot an appropriate value for h. Alternatively, standard approaches such as the secant method can be used to solve the nonlinear equations (6.134) and (6.135).

Quite often, one can provide a minimum value of the change for which one wishes the algorithm to generate an alarm. Let $\beta_{\min}$ denote this value. It is then advisable to choose $\mu_1 = \hat{\mu}_0 + 2\beta_{\min}$. Indeed, let $p_{\hat{\mu}_0}(z)$ and $p_{\hat{\mu}_0+2\beta_{\min}}(z)$ denote the Gaussian probability density functions with respective mean $\hat{\mu}_0$ and $\hat{\mu}_0 + 2\beta_{\min}$ and with variance σ^2. It is easy to check that $p_{\hat{\mu}_0}(z) = p_{\hat{\mu}_0+2\beta_{\min}}(z)$ is achieved for $z =$

$\hat{\mu}_0 + \beta_{\min}$. Thus any sequence of values of z greater than $\hat{\mu}_0 + \beta_{\min}$ on average will yield a sequence of positive log-likelihood ratio $\ln \frac{p_{\hat{\mu}_0+2\beta_{\min}}(z)}{p_{\hat{\mu}_0}(z)}$ on average, and an alarm will be triggered after some time for such a sequence.

Remark 6.10 *Effect of an error on* μ_1
The objective of this remark is to illustrate that the CUSUM algorithm for detection of a change in the mean of a Gaussian sequence can detect changes even when μ_1 is overestimated. To this end, let us consider Fig. 6.11.



(a) (b)

Fig. 6.11. (a) Gaussian probability density functions with actual (continuous line) and overestimated means (dash-dotted line), (b) evolution of the recursive CUSUM decision functions computed with the exact (continuous line) and approximated likelihood ratios (dash-dotted line) for the data sequence of Fig. 6.8

In the left hand figure, the density functions represented by continuous lines correspond to the actual data, which have mean $\mu_0 = 0$ before the change and mean $\mu_1 = 2$ after the change. Fig. 6.8 represents a data sequence which was generated from these density functions. The evolution of the recursive CUSUM decision function tuned with $\mu_0 = 0$ and $\mu_1 = 2$ obtained by processing the data of Fig. 6.8 is plotted as a continuous line in Fig. 6.11(b). Let us now process the same data with a CUSUM algorithm tuned with a mean value after change equal to 3 (instead of 2). The resulting decision function is represented by the dash-dotted line in Fig. 6.11(b). One notices that, when the value of μ_1 in the function $g(k)$ is higher than the real μ, the decision function still increases on average upon occurrence of a change, however the slope of the decision function is lower than with the correct value of μ_1. To understand this phenomenon, let us look again at Fig. 6.11(a), where the Gaussian density function with mean equal to 3 is plotted with a dash-dotted line. Let $p_0(z)$, $p_2(z)$ and $p_3(z)$ denote the density functions with mean 0, 2 or 3, respectively. After the change in the mean, a typical data sample from the actual data sequence, says $\tilde{z}$ will have a value in the neighbourhood of 2. The associated values of the density functions are represented by the points A ($p_2(\tilde{z})$), B($p_3(\tilde{z})$) or C ($p_0(\tilde{z})$), respectively. The contribution to the CUSUM decision function associated to $\tilde{z}$ is equal to $\frac{p_2(\tilde{z})}{p_0(\tilde{z})}$ when the correct tuning is used, and to $\frac{p_3(\tilde{z})}{p_0(\tilde{z})}$ when the μ_1 is overestimated. Both values are clearly larger than one, but $\frac{p_3(\tilde{z})}{p_0(\tilde{z})} < \frac{p_2(\tilde{z})}{p_0(\tilde{z})}$ which explains the lower slope of the CUSUM decision function when μ_1 is overestimated. □

The configuration and the implementation of the CUSUM algorithm to detect changes in the mean of a Gaussian sequence can be summarised as follows:

Algorithm 6.6 *CUSUM algorithm for detection of a change in the mean of a Gaussian sequence*

Given:

1. A set of experimental data $\{z(1), \ldots, z(N)\}$ obtained in fault free working mode.
2. β, corresponding to twice the minimum magnitude of the change to be detected or to the most likely magnitude of this change.
3. A specified mean time for detection or a specified mean time between false alarms.

Initialisation:

1. Determine $\hat{\mu}_0$ and $\hat{\sigma}^2$ from $\{z(1), \ldots, z(N)\}$.
2. Choose h to meet either the specified mean time for detection or the specified mean time between false alarms from (6.134) or (6.135).

At each sample time:

1. Acquire the new data $z(k)$.
2. Compute $g(k)$ by (6.130) and $N(k)$ by (6.128).
3. If $g(k) > h$, issue an alarm, provide an estimate of the change occurrence time $\hat{k}_0$ by (6.129) and reinitialise the decision function to 0.

Result: A sequence of alarm time instants k_a and estimated change occurrence times $\hat{k}_0$, for increasing time horizon k.

The reinitialisation after an alarm allows one to check whether the change in the mean persists as time elapses. More on this issue will be said when the algorithm will be used for fault detection applications.

Example 6.12 (cont.) *Change in the mean of a Gaussian sequence*

From the first 500 data samples plotted in Fig. 6.8, the following estimates were obtained

$$\hat{\mu}_0 = 0.0445 \qquad \hat{\sigma}^2 = 0.946.$$

Letting $\beta = 2$ yields $\hat{\mu}_1 = 2.0445$. Figure 6.12 gives the mean detection delay and the mean time between false alarms as a function of the threshold h, computed from (6.134) and (6.135) were the estimated values are substituted for μ_0, μ_1 and σ^2. The threshold $h = 10$ gives an estimated mean time between false alarms larger than 10^5 while assuring an estimated mean detection delay lower than 6 samples. Figure 6.13 gives a zoom of the decision function in the vicinity of the change time (namely time 500). The alarm will be issued at time 503 which corresponds to a detection delay of three samples (of the order of magnitude of the estimated one). □

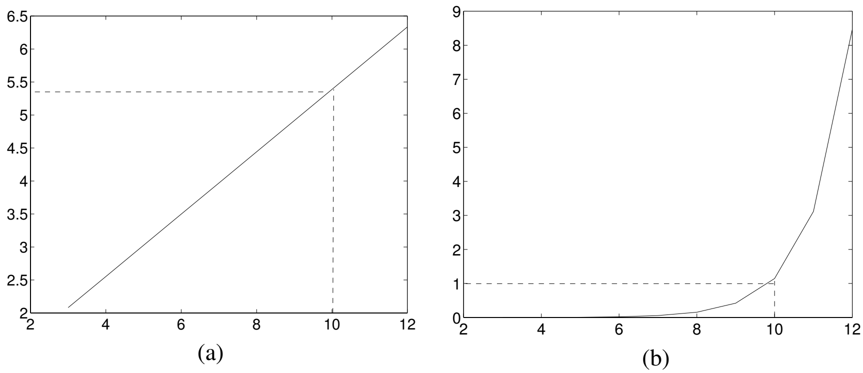


Fig. 6.12. Estimated mean detection delay in number of samples, as a function of h (a) and mean time between false alarms expressed in multiples of 10^5 samples as a function of h (b)

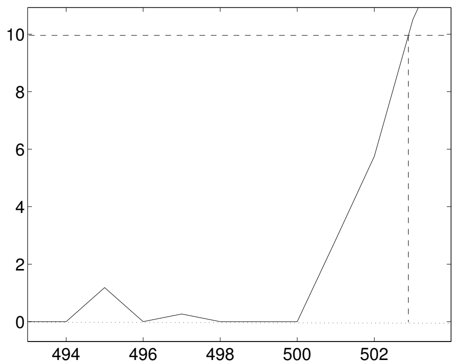


Fig. 6.13. Zoom on the decision function resulting from the recursive algorithm for the data of Fig. 6.8

Another option regarding the choice of θ_1 (the value of the parameter after change) consists in replacing it by the most likely value computed a posteriori from experimental data. This leads to the generalised likelihood ratio algorithm described in the next section.

Deriviation of the generalised likelihood ratio algorithm. The problem can be stated in a similar way as for the off-line derivation of the CUSUM algorithm (cf.

Problem 6.8), except that θ_1 is unknown. By the same reasoning as above, the log-likelihood between hypotheses $\mathcal{H}_0$ and $\mathcal{H}_1$, with an hypothetical change time j, can be computed as

$$S_j^k(\theta_1) = \sum_{i=j}^{k} \ln \frac{p_{\theta_1}(z(i))}{p_{\theta_0}(z(i))}. \tag{6.136}$$

For a given time instant k, it is a function of both j, the change time, and θ_1, the value of the parameter θ after the change. The standard statistical approach to estimate (6.136) is to replace j and θ_1 by their maximum likelihood estimates. The latter are obtained by solving the following double maximisation problem:

$$(\hat{k}_0, \hat{\theta}_1) = \arg\left\{ \max_{1\leq j\leq k} \max_{\theta_1} S_j^k(\theta_1) \right\}, \tag{6.137}$$

where $\hat{k}_0$ denotes the estimate of the change time. The GLR decision function takes the form:

$$g(k) = \max_{1\leq j\leq k} \max_{\theta_1} S_j^k(\theta_1). \tag{6.138}$$

The configuration and the implementation of the algorithm can be summarised as follows:

Algorithm 6.7 *GLR algorithm, scalar parameter*

Given:

1. A sequence of data $z(1), \ldots, z(k)$ with probability density function $p_\theta(z)$ depending on the scalar parameter θ.
2. A threshold h.

Compute: $g(k)$ using (6.136), (6.138).

Decide to

1. accept $\mathcal{H}_0$ if $g(k) \leq h$.
2. accept $\mathcal{H}_1$ if $g(k) > h$.

Remark 6.11 *Maximum and supremum*

Rigorously, the symbol $\max_{\theta_1} S_j^k(\theta_1)$ should be replaced by $\sup_{\theta_1} S_j^k(\theta_1)$ (which gives the least upper bound of $S_j^k(\theta_1)$ with respect to θ_1). The reason is that the maximum may only be reached when θ_1 tends to infinity in pathological cases. However, for engineering purpose, there is no difference between "max" and "sup"; hence only the "max" operation will be used here. □

The maximisation in (6.138) is performed over all possible past time instants. As time elapses, the considered time span increases which induces an increasing search duration for finding the optimum. To avoid that problem, the fault occurrence time is restricted to the last M time instants in practice. This amounts to assuming that the delay for detection is lower than M so that faults can be detected in M sampling periods at most. The actual decision function obtained from (6.138) is thus:

$$g(k) = \max_{k-M+1 \leq j \leq k} \max_{\theta_1} S_j^k(\theta_1). \tag{6.139}$$

Example 6.12 (cont.) *Change in the mean of a Gaussian sequence*

In this particular case, it is possible to find an explicit expression for $\hat{\mu}_1(k, j)$, the maximum likelihood estimate of μ_1 at the present time instant k, assuming that the fault occurred at time instant j. Indeed, from (6.122), $S_j^k(\mu_1)$ takes the following form:

$$S_j^k(\mu_1) = \frac{\mu_1 - \mu_0}{\sigma^2} \sum_{i=j}^{k} \left(z(i) - \frac{\mu_0 + \mu_1}{2} \right) \tag{6.140}$$

In order to maximise this expression with respect to μ_1, one has to take the derivative of $S_j^k(\mu_1)$ with respect to μ_1 and equate that expression to zero:

$$\frac{\partial S_j^k(\mu_1)}{\partial \mu_1} = \frac{1}{\sigma^2} \sum_{i=j}^{k} \left(z(i) - \frac{\mu_0 + \mu_1}{2} \right) - \frac{k-j+1}{2} \frac{(\mu_1 - \mu_0)}{\sigma^2} = 0. \tag{6.141}$$

Equation (6.141) yields:

$$\hat{\mu}_1(k, j) = \frac{1}{k-j+1} \sum_{i=j}^{k} z(i). \tag{6.142}$$

Substituting this expression for μ_1 in (6.140) results, after straightforward computations, in:

$$S_j^k(\hat{\mu}_1(k, j)) = \frac{1}{2\sigma^2} \frac{1}{k-j+1} \left[\sum_{i=j}^{k} (z(i) - \mu_0) \right]^2 . \tag{6.143}$$

Hence the GLR decision function can be written:

$$g(k) = \frac{1}{2\sigma^2} \max_{k-M+1 \leq j \leq k} \frac{1}{k-j+1} \left[\sum_{i=j}^{k} (z(i) - \mu_0) \right]^2 . \tag{6.144}$$

If $\mathcal{H}_1$ is accepted in the above GLR algorithm, at the alarm time k_a, the estimated change occurrence time is given as:

$$\hat{k}_0 = \arg\{ \frac{1}{2\sigma^2} \max_{k_a-M+1 \leq j \leq k_a} \frac{1}{k_a-j+1} \left[\sum_{i=j}^{k_a} (z(i) - \mu_0) \right]^2 . \quad \square \tag{6.145}$$

Parameter tuning for the generlised likelihood ratio algorithm. An experimental approach will be considered here to adjust these parameters. Although the method is described with reference to Example 6.12, it can be generalised easily to other types of changes than jumps in the mean of a Gaussian sequence. First

M should be chosen larger or equal to the acceptable detection delay. Next, the threshold should be determined on the basis of healthy and faulty process data. A computation of the decision function based on a set of healthy data allows one to determine the typical range of values of this function in the absence of fault, and to set the threshold in such a way that false alarms are avoided (or the time between false alarms is acceptable) . This choice can then be validated by processing data obtained in faulty working mode, and checking that detection is achieved. Should experimental data corresponding to a faulty behaviour not be available, a simulator could possibly be used to obtain data that could be used as a substitute. An iterative adjustment of the horizon M and the threshold h may be needed to obtain the right compromise between false alarm rate and detection delay. Indeed, the lower M, the lower h has to be chosen in order to achieve detection in the window $[k-M+1,\ k]$. Decreasing the detection delay thus increases the false alarm rate. For the evaluation of the GLR decision function (6.144) required in the above procedure, empirical estimates $\hat{\mu}_0$ and $\hat{\sigma}^2$ should be substituted for μ_0 and σ^2.

Reinitilization. Again the particular case of a change in the mean of a Gaussian sequence is considered here. The reinitialisation allows one to detect a new change in the mean. The mean value of the data after change is thus considered as the new value of μ_0. This reinitialisation could use $\hat{\mu}_1(k_a, \hat{k}_0)$ as an estimate of the mean after the change occurred. However, if the delay for detection is short (one or a few samples), very few data are used to compute $\hat{\mu}_1(k_a, \hat{k}_0)$, and this estimate of the mean might be poor when the noise on the data is significant. It is the reason why the reinitialisation is based on a data set of fixed length obtained by collecting additional data. Here the length of the data set is chosen equal to M, but an additional parameter different from M might be introduced. It should be determined in such a way that a reliable estimate of the mean after change is obtained. More on this can be found in the appendix on random variables and stochastic processes, where the statistics of the empirical mean is studied.

With this in mind, the global GLR algorithm to detect a change in the mean of a Gaussian sequence can be summarised as follows.

Algorithm 6.8 *GLR algorithm to detect and estimate changes in the mean of a Gaussian sequence*

Given:

1. A set of data $\{z_0(1), \ldots, z_0(N_0)\}$ under hypothesis $\mathcal{H}_0$ and a set $\{z_1(1), \ldots, z_1(N_1)\}$ under hypothesis $\mathcal{H}_1$.
2. An acceptable maximum detection delay.

Initialisation:

1. Choose M larger than or equal to the maximum detection delay.
2. Determine $\hat{\mu}_0$ and $\hat{\sigma}^2$ from $\{z_0(1), \ldots, z_0(N_0)\}$.
3. Compute the decision function $g(i), i = M+1, \ldots, N_0$ for the data set $\{z_0(1), \ldots, z_0(N_0)\}$ by using (6.144) and choose the threshold h so that $g(i) < h,\ i = M+1, \ldots, N_0$.
4. Compute the decision function $g(i),\ i = M+1, \ldots, N_1$ for the data set $\{z_1(1), \ldots, z_1(N_1)\}$ by (6.144) and the estimated change magnitude by (6.142). Check that the fault is detected and that the delay for detection is acceptable for the estimated change magnitude.
5. Possibly iterate on the choice of M and h.
6. Acquire $M-1$ data samples.

At each sampling time:

R1. Acquire the new data $z(k)$.

R2. Compute $g(k)$ from (6.144).

R3. If $g(k) > h$, generate an alarm; provide the alarm time instant $k_a = k$, the estimate of the change occurrence time $\hat{k}_0$ by (6.145) ; and compute $\hat{\mu}_1(k_a, \hat{k}_0)$ by (6.142).

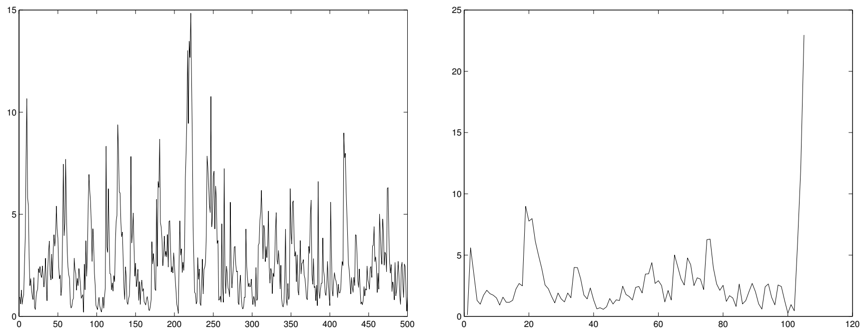
Reinitialisation:

1. Collect a set of M data from time $\hat{k}_0$ to $\hat{k}_0 + M - 1$.
2. Compute the new value of $\hat{\mu}_0$ from these data.
3. Restart the on-line algorithm from $k = \hat{k}_0 + M$ onwards (step R1).

Result: A sequence of alarm time instants k_a, estimated change occurrence times $\hat{k}_0$ and mean signal values $\hat{\mu}_1(k_a, \hat{k}_0)$, for increasing time horizon k.

Example 6.12 (cont.) *Change in the mean of a Gaussian sequence*

Consider again the data of Fig. 6.8. Let the set $z(1), \ldots, z(N_0)$ be made of the first 500 samples while $z(1), \ldots, z(N_1)$ consists of samples 400 to 1000. One gets, as before, $\hat{\mu}_0 = 0.0445$, $\hat{\sigma}^2 = 0.9455$. Figure 6.14 (left) gives the value of the GLR decision function obtained by processing the sequence $z(1), \ldots, z(N_0)$ with a window M of length 10 samples. A threshold above 15 appears to be suitable in this case. Hence, h is set to 20. Running the algorithm on the set $z(1), \ldots, z(N_1)$, one observes that an alarm is generated at time 105 (Fig. 6.14 (right)). The estimate of the change magnitude is 2.69, and the estimate of the change occurrence time is 103, while the actual change occurred at the 101^{st} sample in that set. Due to the noise on the signal, the estimate of the change magnitude is in error by 30%. This could be partly alleviated by increasing the threshold, so that more data are used to estimate the fault magnitude; this would increase the detection delay however. □



(a) (b)

Fig. 6.14. Two GLR decision functions

6.7.3 Sequential change detection: the vector case

The previous discussion dealt with detection of changes in a scalar signal. In the fault detection applications, the signal to be processed is issued by a residual generator, and it is generally a vector signal. The combined information comprised in this vector should be considered in our algorithm. Since fault detection will be reduced to the detection of changes in the mean of a Gaussian vector sequence, the solution to the following problem will be needed.

Problem 6.9 (Detection of a change in the mean of a Gaussian vector sequence) *Consider a sequence of n_z-dimensional random vectors $\boldsymbol{z}(1), \ldots, \boldsymbol{z}(k)$ that are independent and distributed as $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{Q})$, where $\boldsymbol{Q}$ is known, as well as the nominal value for $\boldsymbol{\mu}$, $\boldsymbol{\mu}_0$. Choose between the following two hypotheses:*

$\mathcal{H}_0 : \mathcal{L}(z(i)) = \mathcal{N}(\boldsymbol{\mu}_0, \boldsymbol{Q}), \qquad i = 1, \ldots, k$

$\mathcal{H}_1$: *From time instant* 1 *up to an unknown time instant* k_0, $z(i)$, $i = 1, \ldots, k_0 - 1$ *is distributed as:*

$$\mathcal{L}(\boldsymbol{z}(i)) = \mathcal{N}(\boldsymbol{\mu}_0, \boldsymbol{Q}) \tag{6.146}$$

while for time instants $i \geq k_0$:

$$\mathcal{L}(\boldsymbol{z}(i)) = \mathcal{N}(\boldsymbol{\mu}_1, \boldsymbol{Q}) \tag{6.147}$$

with $\boldsymbol{\mu}_1 \neq \boldsymbol{\mu}_0$.

Beside detecting the possible change in the mean, one should also estimate its time of occurrence, and possibly its magnitude.

When $\boldsymbol{\mu}_1$ is known, the detection algorithm is a direct generalisation of the CUSUM algorithm. In the second situation which is considered in this paragraph, the change in $\boldsymbol{\mu}$ has known direction but unknown magnitude. This yields a GLR algorithm. Finally, the situation, where $\boldsymbol{\mu}_1$ is replaced by a dynamical profile of change will be considered, as this is a result needed at a later stage.

$\boldsymbol{\mu}_1$ known - CUSUM algorithm. By using the expression of the probability density function of a n-dimensional Gaussian vector $\boldsymbol{z}$ with mean $\boldsymbol{\mu}$ and variance $\boldsymbol{Q}$

$$p_{\boldsymbol{\mu}}(\boldsymbol{z}) = \frac{1}{\sqrt{(2\pi)^n det\boldsymbol{Q}}} \exp\left(-\frac{1}{2}\,(\boldsymbol{z}-\boldsymbol{\mu})'\,\boldsymbol{Q}^{-1}(\boldsymbol{z}-\boldsymbol{\mu})\right), \tag{6.148}$$

the following expression is obtained for the log-likelihood ratio associated to the above problem.

$$\begin{aligned} s(\boldsymbol{z}(k)) = \ln \frac{p_{\boldsymbol{\mu}_1}(\boldsymbol{z}(k))}{p_{\boldsymbol{\mu}_0}(\boldsymbol{z}(k))} & \\ = \quad & -\frac{1}{2}\,(\boldsymbol{z}(k)-\boldsymbol{\mu}_1)'\,\boldsymbol{Q}^{-1}\,(\boldsymbol{z}(k)-\boldsymbol{\mu}_1) + \frac{1}{2}\,(\boldsymbol{z}(k)-\boldsymbol{\mu}_0)'\,\boldsymbol{Q}^{-1}\,(\boldsymbol{z}(k)-\boldsymbol{\mu}_0) \\ = \quad & (\boldsymbol{\mu}_1-\boldsymbol{\mu}_0)'\,\boldsymbol{Q}^{-1}\left(\boldsymbol{z}(k) - \frac{1}{2}(\boldsymbol{\mu}_0+\boldsymbol{\mu}_1)\right) \end{aligned} \tag{6.149}$$

This loglikelihood ratio is scalar, so the recursive computation of the CUSUM decision function can be performed in a similar way as for the scalar case (cf. (6.127))

$$g(k) = \max(0,\, g(k-1) + s(\boldsymbol{z}(k))). \tag{6.150}$$

The alarm or stopping time, k_a, is the smallest time instant at which $g(k)$ crosses a given threshold.

Known direction of change - GLR algorithm. Let $\boldsymbol{\mu}_1$ be of the form

$$\boldsymbol{\mu}_1 = \boldsymbol{\mu}_0 + \boldsymbol{\Gamma}\nu,$$

where $\boldsymbol{\Gamma}$ is a known vector, and ν is an unknown scalar change magnitude. Substituting this expression for $\boldsymbol{\mu}_1$ in (6.149) allows one to deduce the following expression of the cumulative sum $S_j^k(\nu)$, where j denotes an hypothetical value of the change time k_0,

$$S_j^k(\nu) = \sum_{i=j}^{k} \ln \frac{p_{\boldsymbol{\mu}_0 + \boldsymbol{\Gamma}\nu}(\boldsymbol{z}(i))}{p_{\boldsymbol{\mu}_0}(\boldsymbol{z}(i))} \tag{6.151}$$
$$= \sum_{i=j}^{k} \left(\nu \boldsymbol{\Gamma}' \boldsymbol{Q}^{-1} (\boldsymbol{z}(i) - \boldsymbol{\mu}_0) - \frac{1}{2}\nu^2 \boldsymbol{\Gamma}' \boldsymbol{Q}^{-1} \boldsymbol{\Gamma} \right).$$

Equating $\frac{\partial S_j^k(\nu)}{\partial \nu}$ to zero yields

$$\frac{\partial S_j^k(\nu)}{\partial \nu} = \sum_{i=j}^{k} \boldsymbol{\Gamma}' \boldsymbol{Q}^{-1} (\boldsymbol{z}(i) - \boldsymbol{\mu}_0) - (k-j+1)\, \boldsymbol{\Gamma}' \boldsymbol{Q}^{-1} \boldsymbol{\Gamma}\nu$$
$$= (k-j+1)\, \boldsymbol{\Gamma}' \boldsymbol{Q}^{-1} \left(\bar{\boldsymbol{Z}}_j^k - \boldsymbol{\mu}_0 \right) - (k-j+1)\, \boldsymbol{\Gamma}' \boldsymbol{Q}^{-1} \boldsymbol{\Gamma}\nu \tag{6.152}$$
$$= 0$$

with $\bar{\boldsymbol{Z}}_j^k = \frac{1}{k-j+1} \sum_{i=j}^{k} \boldsymbol{z}(i)$.

Hence, the maximum likelihood estimate of ν at time k, assuming the fault occurred at time j is obtained from (6.152) as

$$\hat{\nu}(k,\, j) = \frac{\boldsymbol{\Gamma}' \boldsymbol{Q}^{-1} (\bar{\boldsymbol{Z}}_j^k - \boldsymbol{\mu}_0)}{\boldsymbol{\Gamma}' \boldsymbol{Q}^{-1} \boldsymbol{\Gamma}}. \tag{6.153}$$

Substituting (6.153) for ν into (6.151) finally yields the GLR decision function in a similar way as (6.144) was deduced from (6.142) and (6.143)

$$g(k) = \max_{k-M+1 \le j \le k} S_j^k(\hat{\nu}(k,\, j)) = \max_{k-M+1 \le j \le k} (k-j+1) \cdot$$
$$\cdot \left(\hat{\nu}(k,\, j)\, \boldsymbol{\Gamma}' \boldsymbol{Q}^{-1} \left(\bar{\boldsymbol{Z}}_j^k - \boldsymbol{\mu}_0 \right) - \frac{1}{2} \hat{\nu}(k,\, j)^2\, \boldsymbol{\Gamma}' \boldsymbol{Q}^{-1} \boldsymbol{\Gamma} \right).$$

The estimated fault occurrence time upon acceptance of hypothesis $\mathcal{H}_1$ at time instant k_a is given as

$$\hat{k}_0 = \arg\{ \max_{k_a - M+1 \le j \le k_a} (k_a - j + 1) \cdot$$
$$\cdot \left(\hat{\nu}(k_a,\, j) \boldsymbol{\Gamma}' \boldsymbol{Q}^{-1} \left(\bar{\boldsymbol{Z}}_j^k - \boldsymbol{\mu}_0 \right) - \frac{1}{2} \hat{\nu}(k_a, j)^2\, \boldsymbol{\Gamma}' \boldsymbol{Q}^{-1} \boldsymbol{\Gamma} \right) \}.$$

Known dynamical profile of change - CUSUM algorithm. There is a need to generalise the previous result by replacing $\boldsymbol{\Gamma}\nu$ by a time-varying change direction, as this is precisely the situation which is encountered when detecting additive faults in linear systems. This leads to the following modified version of Problem 6.9.

Problem 6.10 (Change detection, known dynamical profile of change)
Consider a sequence of n_z-dimensional random vectors $\boldsymbol{z}(1), \ldots, \boldsymbol{z}(k)$ that are independent and distributed as $\mathcal{N}(\boldsymbol{\mu},\, \boldsymbol{Q})$, where $\boldsymbol{Q}$ is known, as well as the nominal value for $\boldsymbol{\mu}$, $\boldsymbol{\mu}_0$. Choose between the following two hypotheses:

$\mathcal{H}_0 : \mathcal{L}(z(i)) = \mathcal{N}(\boldsymbol{\mu}_0, \boldsymbol{Q}) \qquad i = 1, \ldots, k$
$\mathcal{H}_1$: *From time instant* 1 *up to an unknown time instant* k_0, $z(i)$, $i = 1, \ldots, k_0-1$ *is distributed as*

$$\mathcal{L}(\boldsymbol{z}(i)) = \mathcal{N}(\boldsymbol{\mu}_0, \boldsymbol{Q}) \tag{6.154}$$

while for time instants $i \geq k_0$*:*

$$\mathcal{L}(\boldsymbol{z}(i)) = \mathcal{N}(\boldsymbol{\mu}_0 + \boldsymbol{\rho}(i - k_0), \boldsymbol{Q}), \tag{6.155}$$

where $\boldsymbol{\rho}(i - k_0)$ *is a known vector profile which is non-zero only for* $i \geq k_0$.

Beside detecting the possible change in the mean, one should also estimate its time of occurrence.

The cumulative sum for this problem setting, with j as an hypothetical value for k_0, is given as

$$\begin{aligned} S_j^k &= \sum_{i=j}^{k} \ln \frac{p_{\boldsymbol{\mu}_0 + \boldsymbol{\rho}(i-j)}(\boldsymbol{z}(i))}{p_{\boldsymbol{\mu}_0}(\boldsymbol{z}(i))} \\ &= \sum_{i=j}^{k} \left(-\frac{1}{2}(\boldsymbol{z}(i) - \boldsymbol{\mu}_0 - \boldsymbol{\rho}(i-j))' \boldsymbol{Q}^{-1} (\boldsymbol{z}(i) - \boldsymbol{\mu}_0 - \boldsymbol{\rho}(i-j)) \right) + \\ &\quad + \frac{1}{2} \sum_{i=j}^{k} (\boldsymbol{z}(i) - \boldsymbol{\mu}_0)' \boldsymbol{Q}^{-1} (\boldsymbol{z}(i) - \boldsymbol{\mu}_0) \\ &= \sum_{i=j}^{k} \boldsymbol{\rho}(i-j)' \boldsymbol{Q}^{-1} (\boldsymbol{z}(i) - \boldsymbol{\mu}_0) - \frac{1}{2} \sum_{i=j}^{k} \boldsymbol{\rho}(i-j)' \boldsymbol{Q}^{-1} \boldsymbol{\rho}(i-j) \end{aligned} \tag{6.156}$$

and the decision function, obtained in a similar way as for the scalar case, is given as (cf. (6.125))

$$g(k) = \max_{1 \leq j \leq k} S_j^k$$

with S_j^k as in (6.156).

This algorithm can be written in a recursive form ([5], pp. 283-284):

$$g(k) = \max(0, S(k)) \tag{6.157}$$

$$N(k) = N(k-1)\, 1_{\{g(k-1)>0\}} + 1 \tag{6.158}$$

$$\begin{aligned} S(k) &= S(k-1)\, 1_{\{g(k-1)>0\}} + \boldsymbol{\rho}(N(k)-1)' \boldsymbol{Q}^{-1} (\boldsymbol{z}(k) - \boldsymbol{\mu}_0) - \\ &\quad -0.5 \boldsymbol{\rho}(N(k)-1)' \boldsymbol{Q}^{-1} \boldsymbol{\rho}(N(k)-1). \end{aligned} \tag{6.159}$$

$N(k)$ is thus the number of observations after the last time instant for which the decision function g was null. Notice that this algorithm requires the hypothesis $\boldsymbol{\rho}(0) \neq 0$, which is included in the problem statement, otherwise the decision function would always remain equal to zero.

Example 6.14 *Data exhibiting a dynamical profile of change*

The aim of this example is to illustrate the type of vector signal on which the above algorithm can be applied. Consider a vector signal made of two components, z_1 and z_2. Suppose that the dynamical profile of the change in z_1 (z_2) can be modelled as the step response to a first order system with transfer function $\frac{0.5}{z-0.5}$ ($\frac{1.4}{z-0.3}$). In other words, the sequence $z_j(i)$, $i = 1,\ 2,\ \ldots,\ j = 1,\ 2,$ takes the form

$$z_j(i) = z_j^0(i) + \rho_{s,j}(i - k_0)\, 1_{\{i \geq k_0\}}, \tag{6.160}$$

where

$$\mathcal{L}(z_1^0\,(i)) = \mathcal{L}(z_2^0\,(i)) = \mathcal{N}(0, 0.025),$$

hold and $\rho_{s,j}(\ell)$, which is tabulated below for $\ell = 0,\ \ldots,\ 9$, $j = 1,\ 2$, are the step responses (hence the index s) associated to $\frac{0.5}{z-0.5}$ and $\frac{1.4}{z-0.3}$. Figure 6.15 gives a realisation of the sequence (6.160) for $k_0 = 20$. □

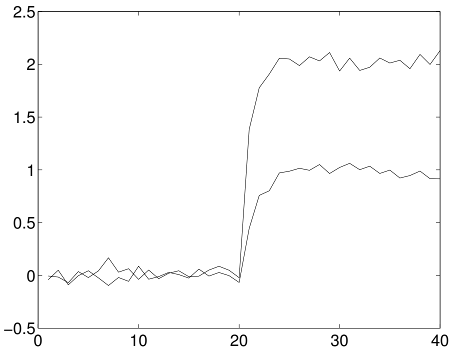


Fig. 6.15. Realisation of the vector sequence (6.160)

The superimposition of the deterministic step response and the stochastic sequence is clearly visible. To apply algorithm (6.157)-(6.159) to detect the change in the sequence, one should take $\rho_j(\ell) = \rho_{s,j}(\ell + 1)$, $\ell = 0, 1, 2\ \ldots$, $j = 1,\ 2$ in order to assure that $\boldsymbol{\rho}(0)$ be nonzero.

Table 6.3 First 10 values of the dynamical profile of the change

ℓ	$\rho_{s,1}$	$\rho_{s,2}$
0	0	0
1	0.5000	1.4000
2	0.7500	1.8200
3	0.8750	1.9460
4	0.9375	1.9838
5	0.9688	1.9951
6	0.9844	1.9985
7	0.9922	1.9996
8	0.9961	1.9999
9	0.9980	2.0000

Known dynamical profile of change up to an unknown constant - GLR algorithm. Yet a more general situation occurs when the form of the dynamical profile of change is known, but its magnitude is not known.

Problem 6.11 (Change detection, known dynamical profile of change up to an unknown constant)
Consider a sequence of n_z-dimensional random vectors $\boldsymbol{z}(1), \ldots, \boldsymbol{z}(k)$ that are independent and distributed as $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{Q})$, where $\boldsymbol{Q}$ is known, as well as the nominal value for $\boldsymbol{\mu}$, $\boldsymbol{\mu}_0$. Choose between the following two hypotheses:

$$\mathcal{H}_0 : \mathcal{L}(z(i)) = \mathcal{N}(\boldsymbol{\mu}_0, \boldsymbol{Q}) \qquad i = 1, \ldots, k$$

$\mathcal{H}_1$: *From time instant* 1 *up to an unknown time instant* k_0, $z(i)$, $i = 1, \ldots, k_0 - 1$ *is distributed as:*

$$\mathcal{L}(\boldsymbol{z}(i)) = \mathcal{N}(\boldsymbol{\mu}_0, \boldsymbol{Q}) \tag{6.161}$$

while for time instants $i \geq k_0$

$$\mathcal{L}(\boldsymbol{z}(i)) = \mathcal{N}(\boldsymbol{\mu}_0 + \tilde{\boldsymbol{\rho}}\,(i - k_0)\nu, \boldsymbol{Q}), \tag{6.162}$$

where $\tilde{\boldsymbol{\rho}}\,(i - k_0)$ is a known vector profile which is non-zero only for $i \geq k_0$, k_0 is an unknown time instant, and ν is an unknown scalar.

Beside detecting the possible change in the mean, one should also estimate its time of occurrence and its magnitude ν.

The cumulative sum for this problem setting, with j as an hypothetical value for k_0, is given as

$$S_j^k(\nu) \quad = \quad \sum_{i=j}^{k} \ln \frac{p_{\boldsymbol{\mu}_0 + \tilde{\boldsymbol{\rho}}\,(i-j)\nu}(\boldsymbol{z}(i))}{p_{\boldsymbol{\mu}_0}(\boldsymbol{z}(i))}$$

$$= \nu \sum_{i=j}^{k} \tilde{\boldsymbol{\rho}}\,(i-j)'\,\boldsymbol{Q}^{-1}\,(\boldsymbol{z}(i)-\boldsymbol{\mu}_0) - \frac{\nu^2}{2}\sum_{i=j}^{k} \tilde{\boldsymbol{\rho}}\,(i-j)'\,\boldsymbol{Q}^{-1}\tilde{\boldsymbol{\rho}}\,(i-j). \tag{6.163}$$

Similar computations as for the case of a constant direction of the parameter change yield the following maximum likelihood estimate of ν at time k, assuming the fault occurred at time j,

$$\hat{\nu}(k,\,j) = \frac{\sum_{i=j}^{k} \tilde{\boldsymbol{\rho}}\,(i-j)'\,\boldsymbol{Q}^{-1}\,(\boldsymbol{z}(i)-\boldsymbol{\mu}_0)}{\sum_{i=j}^{k} \tilde{\boldsymbol{\rho}}\,(i-j)'\,\boldsymbol{Q}^{-1}\tilde{\boldsymbol{\rho}}\,(i-j)} \tag{6.164}$$

and

$$\begin{aligned} g(k) &= \max_{k-M+1\le j\le k}\ \max_{\nu}\ S_j^k(\nu) \\ &= \max_{k-M+1\le j\le k}\ \{\hat{\nu}(k,\,j)\sum_{i=j}^{k} \tilde{\boldsymbol{\rho}}\,(i-j)'\,\boldsymbol{Q}^{-1}\,(\boldsymbol{z}(i)-\boldsymbol{\mu}_0) \\ &\qquad -\frac{\hat{\nu}(k,\,j)^2}{2}\sum_{i=j}^{k} \tilde{\boldsymbol{\rho}}\,(i-j)'\,\boldsymbol{Q}^{-1}\tilde{\boldsymbol{\rho}}\,(i-j)\}. \end{aligned} \tag{6.165}$$

The estimated fault occurrence time upon acceptance of hypothesis $\mathcal{H}_1$ at time instant k_a is given as

$$\begin{aligned} \hat{k}_0 &= \arg\left\{\max_{k_a-M+1\le j\le k_a}\ \hat{\nu}(k_a,\,j)\sum_{i=j}^{k_a} \tilde{\boldsymbol{\rho}}\,(i-j)'\,\boldsymbol{Q}^{-1}\,(\boldsymbol{z}(i)-\boldsymbol{\mu}_0) - \right. \\ &\qquad \left. -\frac{\hat{\nu}(k_a,\,j)^2}{2}\sum_{i=j}^{k_a} \tilde{\boldsymbol{\rho}}\,(i-j)'\,\boldsymbol{Q}^{-1}\tilde{\boldsymbol{\rho}}\,(i-j)\right\}. \end{aligned} \tag{6.166}$$

Parameter setting for the CUSUM algorithm. For the CUSUM algorithm associated to a known vector $\boldsymbol{\mu}_1$, the expressions of the ARL function (6.133) remains valid in the vector case. It suffices to replace μ_s and σ_s by:

$$\begin{aligned} \mu_s &= \pm\frac{(\boldsymbol{\mu}_1-\boldsymbol{\mu}_0)'\,\boldsymbol{Q}^{-1}(\boldsymbol{\mu}_1-\boldsymbol{\mu}_0)}{2} \\ \sigma_s^2 &= (\boldsymbol{\mu}_1-\boldsymbol{\mu}_0)'\,\boldsymbol{Q}^{-1}(\boldsymbol{\mu}_1-\boldsymbol{\mu}_0), \end{aligned}$$

where the plus or the minus sign are chosen according to the value of $\boldsymbol{\mu}$, the expected value of $\boldsymbol{z}(i)$ (cf. Remark 6.8). The mean time for detection and the mean time between false alarms can respectively be estimated as

$$\hat{\bar{\tau}} = \hat{L}\left(\frac{(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_0)'\, \boldsymbol{Q}^{-1} (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_0)}{2}\right) = \hat{L}\left(\frac{1}{2}\boldsymbol{\beta}'\,\boldsymbol{Q}^{-1}\boldsymbol{\beta}\right), \tag{6.167}$$

where $\boldsymbol{\beta} = \boldsymbol{\mu}_1 - \boldsymbol{\mu}_0$

$$\hat{\bar{T}} = \hat{L}\left(-\frac{(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_0)'\, \boldsymbol{Q}^{-1} (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_0)}{2}\right). \tag{6.168}$$

In order to evaluate the above expressions, given a set of experimental data recorded in fault free condition and a change magnitude $\boldsymbol{\beta}$, the empirical mean and variances $\hat{\boldsymbol{\mu}}_0$ and $\hat{\boldsymbol{Q}}$ should be substituted for $\boldsymbol{\mu}_0$ and $\boldsymbol{Q}$. These empirical estimates should also be used for the implementation of (6.149)-(6.150).

The algorithm for the vector parameter case is thus similar to the scalar case, namely

Algorithm 6.9 *CUSUM algorithm for Gaussian vector sequence, step-like change (μ_1 known)*

Given:

1. A set of experimental data $\{\boldsymbol{z}(1), \ldots, \boldsymbol{z}(N)\}$ obtained in fault free working mode.
2. $\boldsymbol{\beta}$, corresponding to twice the minimum magnitude of the change to be detected or to the most likely magnitude of this change.
3. A specified mean time for detection or a specified mean time between false alarms.

Initialisation:

1. Determine $\hat{\boldsymbol{\mu}}_0$ and $\hat{\boldsymbol{Q}}$ from $\{\boldsymbol{z}(1), \ldots, \boldsymbol{z}(N)\}$.
2. Choose h to meet either the specified mean time for detection or the specified mean time between false alarms from (6.167) or (6.168).

At each sample time:

1. Acquire the new data vector $\boldsymbol{z}(k)$.
2. Compute $g(k)$ by (6.150).
3. If $g(k) > h$, issue an alarm and reinitialise the decision function to 0.

Result: A sequence of alarm time instants k_a and estimated change occurrence times $\hat{k}_0$, for increasing time horizon k.

When the dynamical profile of the change is accounted for (as in (6.155)), the study of the properties of the algorithm such as mean delay for detection and mean time between false alarms becomes much more difficult. The difficulty stems from the fact that the increments in the cumulative sum of log-likelihood ratios are not identically distributed. Therefore, an experimental approach for setting the design parameters is proposed in the algorithm below.

Algorithm 6.10 *CUSUM algorithm for Gaussian vector sequence, known dynamical profile of change*

Given:

1. A set of data $\{\boldsymbol{z}_0(1), \ldots, \boldsymbol{z}_0(N_0)\}$ under hypothesis $\mathcal{H}_0$ and a set $\{\boldsymbol{z}_1(1), \ldots, \boldsymbol{z}_1(N_1)\}$ under hypothesis $\mathcal{H}_1$.
2. A dynamical profile of change $\boldsymbol{\rho}(i) \neq 0,\ i \geq 0$.

Initialisation:

1. Determine $\hat{\boldsymbol{\mu}}_0$ and $\hat{\boldsymbol{Q}}$ from $\{\boldsymbol{z}_0(1), \ldots, \boldsymbol{z}_0(N_0)\}$.
2. Compute the decision function $g(i), i = 1, \ldots, N_0$ for the data set $\{\boldsymbol{z}_0(1), \ldots, \boldsymbol{z}_0(N_0)\}$ by (6.157) – (6.159) and choose the threshold h so that

   $g(i) < h,\ i = 1, \ldots, N_0.$

4. Compute the decision function $g(i),\ i = 1, \ldots, N_1$ for the data set $\{\boldsymbol{z}_1(1), \ldots, \boldsymbol{z}_1(N_1)\}$ by (6.157) – (6.159); check that the fault is detected and that the delay for detection is acceptable.
6. Possibly iterate on the choice of h.

At each sampling time:

R1. Acquire the new data $\boldsymbol{z}(k)$.

R2. Compute $g(k)$ from (6.157) – (6.159).

R3. If $g(k) > h$, generate an alarm by setting $k_a = k$, and provide an estimate of the change occurrence time as $k_a - N(k_a)$ by (6.158).

Reinitialisation:

1. Reset $g(k_a)$, $N(k_a)$, and $S(k_a)$ to zero in (6.157) – (6.159).
2. Restart the recursive algorithm with (step R1).

Result: A sequence of alarm time instants k_a and estimated fault occurrence times $\hat{k}_0$, for increasing time horizon k.

Remark 6.12 *Reinitialisation procedure*
The reinitialisation may depend on what one wishes to detect. Here it is assumed that one wishes to check whether the observed change remains present. By reinitialising the algorithm

as proposed, repeated alarms will occur as long as the change is present in the signal. Another option for reinitialisation is to change the sign of the loglikelihood ratio which amounts to changing the sign of the last two terms in (6.159) and to reset all variables to zero as indicated is step 1 of the reinitialisation. In this way a return to normal will generate an alarm.

The proposed reinitilisation policies require that the dynamical profile of change does not asymptotically vanish. Should this not hold, one should resort to a GLR algorithm as illustrated in the ship example in Section 6.8.4 □

An example of application of this algorithm in the framework of a fault detection system is given in Section 6.8.2.

GLR algorithm Here also an experimental approach is used to set the design parameters. The algorithm is only presented for a change characterised by a dynamical profile with unknown magnitude.

Algorithm 6.11 *GLR algorithm, known dynamical profile but unknown change magnitude*

Given:

1. A set of data $\{\boldsymbol{z}_0(1), \ldots, \boldsymbol{z}_0(N_0)\}$ under hypothesis $\mathcal{H}_0$ and a set $\{\boldsymbol{z}_1(1), \ldots, \boldsymbol{z}_1(N_1)\}$ under hypothesis $\mathcal{H}_1$.
2. A dynamical profile of change $\tilde{\boldsymbol{\rho}}(i)$, $i \geq 0$.

Initialisation:

1. Choose M larger than or equal to largest admissible detection delay.
2. Determine $\hat{\boldsymbol{\mu}}_0$ and $\hat{\boldsymbol{Q}}$ from $\{\boldsymbol{z}_0(1), \ldots, \boldsymbol{z}_0(N_0)\}$.
3. Compute the decision function $g(i)$, $i = M+1, \ldots, N_0$ for the data set $\{\boldsymbol{z}_0(1), \ldots, \boldsymbol{z}_0(N_0)\}$ by (6.164), (6.165)), and choose the threshold h so that $g(i) < h$, $i = M+1, \ldots, N_0$.
4. Compute the decision function $g(i)$, $i = M+1, \ldots, N_1$ for the data set $\{\boldsymbol{z}_1(1), \ldots, \boldsymbol{z}_1(N_1)\}$ by (6.165) and the estimated change magnitude by (6.164); check that the fault is detected and that the delay for detection is acceptable given the estimated change magnitude.
5. Possibly iterate on the choice of M and h.
6. Acquire $M-1$ data samples.

At each sampling time:

R1. Acquire the new data $\boldsymbol{z}(k)$.

R2. Compute $g(k)$ from (6.164), (6.165)

R3. If $g(k) > h,$ generate an alarm; provide the estimate of the change occurrence time, $\hat{k}_0$, by (6.166), and the estimated change magnitude $\hat{\nu}(k, \hat{k}_0)$ computed by (6.164).

Reinitialisation:

1. Collect a set of M data from time $\hat{k}_0$ to $\hat{k}_0 + M - 1$.
2. Compute the estimated change magnitude $\hat{\boldsymbol{\nu}}(\hat{k}_0 + M - 1,\, \hat{k}_0)$ for these data by (6.164).
3. Restart the recursive algorithm from $k = \hat{k}_0 + M$ onwards with step R1. Notice that the mean value to be subtracted from $\boldsymbol{z}(i)$ in (6.164), (6.165) should now be $\hat{\boldsymbol{\mu}}_0 + \hat{\nu}(\hat{k}_0 + M - 1,\, \hat{k}_0)\tilde{\boldsymbol{\rho}}(i - \hat{k}_0)$ (cf. Remark 6.13 below).

Result: A sequence of alarm time instants k_a, estimated change occurrence times $\hat{k}_0$ and change magnitudes $\hat{\nu}(k, \hat{k}_0)$.

Remark 6.13 *Reinitialisation for GLR algorithm*

- The reason for collecting a set of M data to estimate $\hat{\nu}(\hat{k}_0 + M - 1,\, \hat{k}_0)$ is to assure a sufficient precision of the change magnitude so that updating the mean of the signal is performed properly.
- After reinitialisation, (6.164) and (6.165) should be replaced in step R2 by the following expression which accounts for the estimated mean of the signal after change

$$\hat{\nu}(k,\, j) = \frac{\sum_{i=j}^{k} \tilde{\boldsymbol{\rho}}\,(i-j)'\, \boldsymbol{Q}^{-1}\left(\boldsymbol{z}(i) - \hat{\boldsymbol{\mu}}_0 - \hat{\nu}\tilde{\rho}(i - \hat{k}_0)\right)}{\sum_{i=j}^{k} \tilde{\boldsymbol{\rho}}\,(i-j)'\, \boldsymbol{Q}^{-1}\tilde{\boldsymbol{\rho}}\,(i-j)} \tag{6.169}$$

$$g(k) = \max_{k-M+1\le j\le k} \left\{ \hat{\nu}(k,\, j)\sum_{i=j}^{k} \tilde{\boldsymbol{\rho}}\,(i-j)'\, \boldsymbol{Q}^{-1}\left(\boldsymbol{z}(i) - \hat{\boldsymbol{\mu}}_0 - \hat{\nu}\tilde{\rho}(i - \hat{k}_0)\right) - \right.$$
$$\left. - \frac{\hat{\nu}(k,\, j)^2}{2}\sum_{i=j}^{k} \tilde{\boldsymbol{\rho}}\,(i-j)'\, \boldsymbol{Q}^{-1}\tilde{\boldsymbol{\rho}}\,(i-j) \right\}, \tag{6.170}$$

where $\hat{\nu}$ stands for $\hat{\nu}(\hat{k}_0 + M - 1,\, \hat{k}_0)$. □

The method will be illustrated in Section 6.8.3, as a part of a fault detection and estimation system.

6.8 Stochastic model – Kalman filter approach

After a presentation of the model of the supervised process, the problems of detection, isolation and estimation of additive faults in a stochastic system will be successively considered in this section.

6.8.1 Model

Let us consider a system described by a linear discrete-time model of the form

$$\begin{aligned} \boldsymbol{x}(k+1) &= \boldsymbol{A}\boldsymbol{x}(k) + \boldsymbol{B}\boldsymbol{u}(k) + \boldsymbol{F}_d\,\boldsymbol{d}(k) + \boldsymbol{F}_f\,\boldsymbol{f}(k) + \boldsymbol{w}(k) \\ \boldsymbol{x}(0) &= \boldsymbol{x}_0 \\ \boldsymbol{y}(k) &= \boldsymbol{C}\boldsymbol{x}(k) + \boldsymbol{D}\boldsymbol{u}(k) + \boldsymbol{E}_d\,\boldsymbol{d}(k) + \boldsymbol{E}_f\,\boldsymbol{f}(k) + \boldsymbol{v}(k), \end{aligned} \tag{6.171}$$

where $\boldsymbol{x} \in \mathbb{R}^n$, $\boldsymbol{u} \in \mathbb{R}^m$, $\boldsymbol{y} \in \mathbb{R}^p$ are respectively the state vector, the vector of known input signals and the vector of measured output signals, $\boldsymbol{w}$ is the vector of state noise, $\boldsymbol{v}$ denotes the measurement noise. $\boldsymbol{w}(k)$ and $\boldsymbol{v}(k)$ are samples of vector white noise sequences with zero mean and covariance matrix:

$$E\left[\begin{pmatrix} \boldsymbol{w}(k) \\ \boldsymbol{v}(k) \end{pmatrix} (\boldsymbol{w}(\ell)'\boldsymbol{v}(\ell)')\right] = \begin{pmatrix} \boldsymbol{Q}_w & \boldsymbol{Q}_{wv} \\ \boldsymbol{Q}'_{wv} & \boldsymbol{Q}_v \end{pmatrix} \delta_{k\ell}.$$

$\boldsymbol{x}_0$ is a stochastic vector with mean $\boldsymbol{m}_0$ and variance $\boldsymbol{\Pi}_0$ uncorrelated with the state and measurement noise sequences. Finally, $\boldsymbol{d} \in \mathbb{R}^{n_d}$ is a vector of unknown input signals or disturbances (deterministic or stochastic with non-zero mean), and $\boldsymbol{f} \in \mathbb{R}^{n_f}$ is a vector of unknown input signals representing the faults to be detected. The faults are said to be additive, since they enter linearly in the model as additional input.

Such a model can also be written in terms of a single vector white noise sequence, with variance equal to the identity matrix by considering the factorisation

$$\begin{pmatrix} \boldsymbol{Q}_w & \boldsymbol{Q}_{wv} \\ \boldsymbol{Q}'_{wv} & \boldsymbol{Q}_v \end{pmatrix} = \begin{pmatrix} \boldsymbol{B}_\epsilon \\ \boldsymbol{D}_\epsilon \end{pmatrix} \begin{pmatrix} \boldsymbol{B}'_\epsilon & \boldsymbol{D}'_\epsilon \end{pmatrix}.$$

A sample of this sequence will be denoted $\boldsymbol{\epsilon}(k)$, hence the index in $\boldsymbol{B}_\epsilon$ and $\boldsymbol{D}_\epsilon$. It is a n_ϵ-dimensional random vector, where n_ϵ is the rank of the variance of $(\boldsymbol{w}(k)' \;\; \boldsymbol{v}(k)')'$, generally equal to $n + p$. The state-space model (6.171) can thus be rewritten as

$$\begin{aligned} \boldsymbol{x}(k+1) &= \boldsymbol{A}\boldsymbol{x}(k) + \boldsymbol{B}\boldsymbol{u}(k) + \boldsymbol{F}_d\,\boldsymbol{d}(k) + \boldsymbol{F}_f\,\boldsymbol{f}(k) + \boldsymbol{B}_\epsilon\boldsymbol{\epsilon}(k) \\ \boldsymbol{x}(0) &= \boldsymbol{x}_0 \\ \boldsymbol{y}(k) &= \boldsymbol{C}\boldsymbol{x}(k) + \boldsymbol{D}\boldsymbol{u}(k) + \boldsymbol{E}_d\,\boldsymbol{d}(k) + \boldsymbol{E}_f\,\boldsymbol{f}(k) + \boldsymbol{D}_\epsilon\boldsymbol{\epsilon}(k). \end{aligned} \tag{6.172}$$

6.8.2 Fault detection

Problem statement. Fault detection amounts to determining whether the supervised process is working in a normal (or healthy) operating mode. The problem can be stated as follows.

Problem 6.12 (Fault detection)
Given

- *a model of the process of the form (6.171) or (6.172)*
- *a sequence of measured process input and output*

$$(\boldsymbol{y}(i), \boldsymbol{u}(i))_{1 \le i \le k},$$

where k denotes the current time instant.

Choose between the following two hypotheses:

$\mathcal{H}_0$: *healthy operating condition,*
$\mathcal{H}_1$: *faulty operating condition.*

The quality of a fault detection system is measured in terms of detection delay and time between false alarms. A typical objective is to minimise the mean delay for detection of a change subject to a fixed false alarm rate before the change time.

To achieve this goal, the task is usually divided into two parts: residual generation and residual evaluation. Each of them is addressed successively in the following subsections.

Residual generation. As in the deterministic case, the residual generators are filters with input signals $\boldsymbol{u}$ and $\boldsymbol{y}$, belonging to the following class of linear time invariant systems

$$\begin{aligned} \boldsymbol{z}(k+1) &= \boldsymbol{A}_z\, \boldsymbol{z}(k) + \boldsymbol{B}_{zu}\, \boldsymbol{u}(k) + \boldsymbol{B}_{zy}\, \boldsymbol{y}(k), \qquad \boldsymbol{z}(0) = \boldsymbol{z}_0 \\ \boldsymbol{r}(k) &= \boldsymbol{C}_{rz}\, \boldsymbol{z}(k) + \boldsymbol{D}_{ru}\, \boldsymbol{u}(k) + \boldsymbol{D}_{ry}\, \boldsymbol{y}(k) \end{aligned} \tag{6.173}$$

or, in transfer function form, after taking the z-transform of the above equations and assuming zero initial conditions:

$$\begin{aligned} \boldsymbol{r}(z) &= \boldsymbol{V}_{ru}(z)\boldsymbol{u}(z) + \boldsymbol{V}_{ry}(z)\boldsymbol{y}(z) \qquad (6.174) \\ &= \left(\boldsymbol{V}_{ru}(z) \;\; \boldsymbol{V}_{ry}(z)\right) \begin{pmatrix} \boldsymbol{u}(z) \\ \boldsymbol{y}(z) \end{pmatrix}. \end{aligned}$$

The problem of designing a residual generator can be stated as follows.

Problem 6.13 (Residual generator design for fault detection)
Determine a stable linear time-invariant filter (6.173) or (6.174) such that:

1. *The sequence of output values $\boldsymbol{r}(k)$, $k = 1, 2, \ldots$ is a zero mean white noise vector sequence (which is not affected by $\boldsymbol{u}$ and $\boldsymbol{d}$), once the transient due to initial conditions has vanished.*

2. *In the presence of a fault ($\boldsymbol{f}(k) \neq 0$ for all $k \geq k_0$), the mean of $\boldsymbol{r}(k)$ is different from zero for at least some $k \geq k_0$.*

As $\boldsymbol{u}$ and $\boldsymbol{d}$ are arbitrary signals, they cannot affect $\boldsymbol{r}$ for the latter to be a white noise sequence. To define rigorously what is meant by this statement, notice that the global system made of the supervised process and the residual generator, obtained by combining Eqs. (6.172) and (6.173), is seen to have as input signals $\boldsymbol{u}$, $\boldsymbol{d}$, $\boldsymbol{f}$, $\boldsymbol{\epsilon}$, and state $(\boldsymbol{x} \;\; \boldsymbol{z})'$. Hence the residual at time k can be considered as a function of the above input and the initial state, namely:

$$\boldsymbol{r}(k) = \boldsymbol{r}(k;\, \boldsymbol{u},\, \boldsymbol{d},\, \boldsymbol{f},\, \boldsymbol{\epsilon};\, \boldsymbol{x}_0,\, \boldsymbol{z}_0).$$

Saying that the residual is not affected by $\boldsymbol{u}$ and $\boldsymbol{d}$ means that, for any two distinct input sequences $\boldsymbol{u}^1(k), \boldsymbol{u}^2(k)$ and $\boldsymbol{d}^1(k), \boldsymbol{d}^2(k), k = 1, 2, \ldots$,

$$\boldsymbol{r}\left(k;\, \boldsymbol{u}^1,\, \boldsymbol{d},\, \boldsymbol{f},\, \boldsymbol{\epsilon};\, \boldsymbol{x}_0,\, \boldsymbol{z}_0\right) = \boldsymbol{r}\left(k;\, \boldsymbol{u}^2,\, \boldsymbol{d},\, \boldsymbol{f},\, \boldsymbol{\epsilon};\, \boldsymbol{x}_0,\, \boldsymbol{z}_0\right)$$

and

$$\boldsymbol{r}\left(k;\, \boldsymbol{u},\, \boldsymbol{d}^1,\, \boldsymbol{f},\, \boldsymbol{\epsilon};\, \boldsymbol{x}_0,\, \boldsymbol{z}_0\right) = \boldsymbol{r}\left(k;\, \boldsymbol{u},\, \boldsymbol{d}^2,\, \boldsymbol{f},\, \boldsymbol{\epsilon};\, \boldsymbol{x}_0,\, \boldsymbol{z}_0\right),$$

whatever the initial state and the other input sequences.

One way to solve the problem is to design a filter which meets the first condition and then to check whether the second requirement is fulfilled. In order to maximise the chances for this second condition to hold, one should make sure that, when imposing condition 1, no useful information contained in $\boldsymbol{y}$ is lost. Only the information corrupted by an unknown input should be cancelled. A filter that meets the latter condition, together with the first condition of Problem 6.13 is called an innovation filter for reasons that will be clarified in the next subsection.

To construct a residual generator, one will first solve the innovation filter design problem below. Next fault sensitivity of the filter output will be checked to see whether condition 2 is met in Problem 6.13. In the affirmative, the innovation filter is a residual generator. The issue of fault sensitivity is the object of a specific subsection.

Problem 6.14 (Innovation filter design)
Determine a stable linear time-invariant filter (6.174) with the least number of output signals such that, in the absence of fault (i.e. $\boldsymbol{f}(k) = 0$ for all k):

1. *The sequence of output values $\boldsymbol{r}(k)$, $k = 1, 2, \cdots$ is a zero mean white noise vector sequence which is not affected by* **u** *and* **d**, *once the transient due to initial conditions has vanished.*
2. *No information on the fault contained in $\boldsymbol{y}$ is lost, except if it is affected by the unknown input vector* **d**.

An observer-based approach will be used to solve the Problem 6.14.

Two situations have to be distinguished, namely the absence of unknown input ($\boldsymbol{E}_d = \boldsymbol{O}$ and $\boldsymbol{F}_d = \boldsymbol{O}$) and the presence of unknown input.

No unknown input. In this situation the design of an innovation filter amounts to the design of a steady state Kalman filter. Such a filter provides a prediction of the output $\boldsymbol{y}(k)$, $\hat{\boldsymbol{y}}(k)$, given the data up to time $k-1$, namely, given $\boldsymbol{u}(i)$, $\boldsymbol{y}(i)$, $i = 1, 2, \ldots, k-1$. The output prediction error, $\boldsymbol{r}(k) = \boldsymbol{y}(k) - \hat{\boldsymbol{y}}(k)$ is called the innovation in standard Kalman filter literature, because it consists of the new information contained in $\boldsymbol{y}(k)$, which was not available in $\boldsymbol{y}(1), \ldots, \boldsymbol{y}(k-1)$. The innovation sequence is known to be a white noise sequence not affected by $\boldsymbol{u}$ (once the transient due to initial conditions has decayed to zero). Hence it fulfils condition 1 of Problem 6.14. Besides, the information about $\boldsymbol{f}$ contained in the sequence of data is also contained in the innovation sequence. For this reason, the innovation is said to be a sufficient statistics for the fault vector $\boldsymbol{f}$. Thus condition 2 of Problem 6.14 is also fulfilled by the innovation sequence, and hence the latter is a suitable candidate as a residual sequence. It is the fact that the innovation sequence meets conditions 1 and 2 of Problem 6.14 that justifies the name innovation filter.

To state the design procedure precisely, let us introduce the notion of a regular quadruple $(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}, \boldsymbol{D})$.

Definition 6.6 (Regular quadruple $(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}, \boldsymbol{D})$**)**
$(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}, \boldsymbol{D})$ *is regular if the matrix*

$$\begin{pmatrix} -z\boldsymbol{I} + \boldsymbol{A} & \boldsymbol{B} \\ \boldsymbol{C} & \boldsymbol{D} \end{pmatrix}$$

has full row rank for all z on the unit cycle ($z = \exp(j\omega)$, $\omega \in \mathrm{I\!R}$).

It is assumed below that the pair $(\boldsymbol{C}, \boldsymbol{A})$ is detectable, and $(\boldsymbol{A}, \boldsymbol{B}_\epsilon, \boldsymbol{C}, \boldsymbol{D}_\epsilon)$ is regular.

Remark 6.14 *Uncorrelated state and measurement noise sequences*
In the particular case where the sequence $\boldsymbol{w}(k)$ and $\boldsymbol{v}(k)$ are uncorrelated (which amounts to $\boldsymbol{B}_\epsilon \boldsymbol{D}_\epsilon' = \boldsymbol{O}$), the above regularity assumption can be replaced by the classical requirement that the pair $(\boldsymbol{A}, \boldsymbol{B}_\epsilon)$ has no uncontrollable mode on the unit circle. □

Under such hypotheses, the equations for the steady state Kalman filter can be written as

$$\begin{aligned} \hat{\boldsymbol{x}}(k+1) &= \boldsymbol{A}\hat{\boldsymbol{x}}(k) + \boldsymbol{B}\boldsymbol{u}(k) - \boldsymbol{K}(\boldsymbol{y}(k) - \boldsymbol{C}\hat{\boldsymbol{x}}(k) - \boldsymbol{D}\boldsymbol{u}(k)), \\ \hat{\boldsymbol{x}}(0) &= \hat{\boldsymbol{x}}_0 \\ \boldsymbol{r}(k) &= \boldsymbol{y}(k) - \boldsymbol{C}\hat{\boldsymbol{x}}(k) - \boldsymbol{D}\boldsymbol{u}(k), \end{aligned} \tag{6.175}$$

where the filter gain $\boldsymbol{K}$ is given by

$$\boldsymbol{K} = -(\boldsymbol{A}\boldsymbol{P}\boldsymbol{C}' + \boldsymbol{Q}_{wv})(\boldsymbol{Q}_v + \boldsymbol{C}\boldsymbol{P}\boldsymbol{C}')^{-1}, \tag{6.176}$$

$\boldsymbol{P}$ being the symmetric semi-positive definite solution of the following discrete algebraic Riccati equation

$$
\begin{aligned}
P &= APA' - (APC' + Q_{wv})(Q_v + CPC')^{-1} \cdot \\
&\quad \cdot (CPA' + Q'_{wv}) + Q_w .
\end{aligned}
\tag{6.177}
$$

Fig. 6.16 illustrates the state-space implementation of the innovation filter.

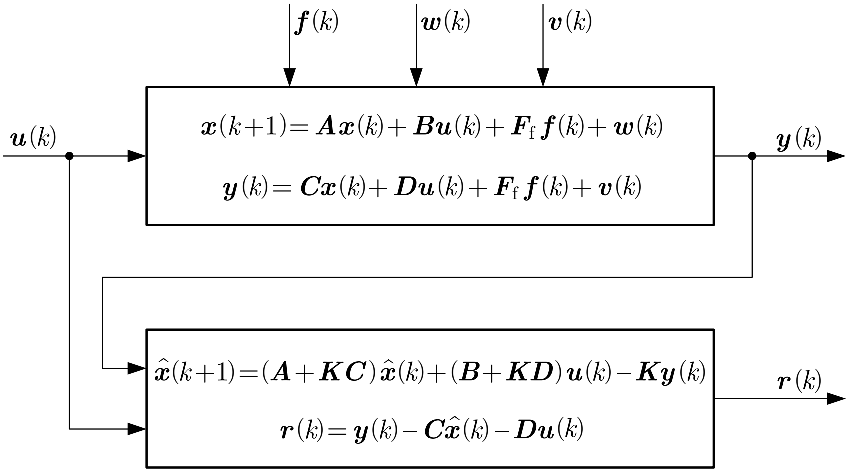


Fig. 6.16. Block diagram of the supervised system together with the innovation filter

In a transfer function form, this filter is described by

$$
\begin{aligned}
r(z) &= V_{ru}(z)u(z) + V_{ry}(z)y(z) \\
&= (-C(zI - A - KC)^{-1}(B + KD) - D)u(z) + \\
&\quad + (C(zI - A - KC)^{-1}K + I)y(z) .
\end{aligned}
\tag{6.178}
$$

If the pair $(C,\ A)$ is not detectable, it is still possible to design an innovation filter by extracting the observable part of system (6.172) and designing a Kalman filter for this observable subsystem. Notice that this approach can also be considered when (6.172) is detectable but not observable, if one wishes to obtain a residual generator with the lowest possible order.

Remark 6.15 *Time varying Kalman filter*
To assure coherency with Sections 6.3 and 6.4 and to ease the study of the sensitivity to the fault, a steady state Kalman filter is considered above. This implies that whiteness of the sequence $r(k)$ is only reached after the transient has vanished. A white noise sequence can be generated from time $k = 0$, if a (time-varying) Kalman filter is used instead of a steady state Kalman filter. Equation (6.175), (6.176) and (6.177) are then replaced by

$$
\begin{aligned}
\hat{x}(k+1) &= A\hat{x}(k) + Bu(k) - K(k)(y(k) - C\hat{x}(k) - Du(k)) \\
\hat{x}(0) &= m_0 \\
r(k) &= y(k) - C\hat{x}(k) - Du(k),
\end{aligned}
\tag{6.179}
$$

where the filter gain $\boldsymbol{K}(k)$ is given by

$$\boldsymbol{K}(k) = -(\boldsymbol{A}\boldsymbol{P}(k)\boldsymbol{C}' + \boldsymbol{Q}_{wv})(\boldsymbol{Q}_v + \boldsymbol{C}\boldsymbol{P}(k)\boldsymbol{C}')^{-1}, \tag{6.180}$$

$\boldsymbol{P}(k)$ being the solution of the following discrete Riccati equation

$$\begin{aligned}\boldsymbol{P}(k+1) \quad &= \quad \boldsymbol{A}\boldsymbol{P}(k)\boldsymbol{A}' - (\boldsymbol{A}\boldsymbol{P}(k)\boldsymbol{C}' + \boldsymbol{Q}_{wv})(\boldsymbol{Q}_v + \boldsymbol{C}\boldsymbol{P}(k)\boldsymbol{C}')^{-1} \cdot \\ &\quad \cdot (\boldsymbol{C}\boldsymbol{P}(k)\boldsymbol{A}' + \boldsymbol{Q}'_{wv}) + \boldsymbol{Q}_w, \qquad \boldsymbol{P}(0) = \boldsymbol{\Pi}_0\end{aligned} \tag{6.181}$$

For implementation purpose, it is interesting to separate the equations of the Kalman filter in a two-stage update procedure at each sampling time: a measurement update and a time update. The approach is the object of the following remark. This implementation of the Kalman filter allows one to handle missing measurements in a straightforward way. □

Remark 6.16 *Handling missing measurements*

The measurement update consists in taking into account the new information brought by an additional measurement, say $\boldsymbol{y}(k)$, in order to compute $\hat{\boldsymbol{x}}(k|k)$, the best estimate (in the least square sense) of $\boldsymbol{x}(k)$ given $\boldsymbol{u}(i), \boldsymbol{y}(i), i = 1, 2, \cdots, k$. The latter is deduced from $\boldsymbol{u}(k), \boldsymbol{y}(k)$ and from $\hat{\boldsymbol{x}}(k)$ the best prediction of $\boldsymbol{x}(k)$ given $\boldsymbol{u}(i), \boldsymbol{y}(i), i = 1, 2, \cdots, k-1$. The time update then uses the plant model in order to predict the state evolution one step ahead.

An additional hypothesis is needed to use the algorithm below: the variance of the measurement noise, $\boldsymbol{Q}_v$ should be positive definite and diagonal. Thus, $\boldsymbol{Q}_v = \text{ diag } (q_{v,1} \cdots q_{v,p})$, where $q_{vi} > 0, i = 1, \cdots, p$. The "diagonality" condition can be enforced by a suitable change of output variable when $\boldsymbol{Q}_v$ is positive definite. It suffices to set $\boldsymbol{y}_d(k) = \boldsymbol{Q}_v^{-1/2}\boldsymbol{y}(k)$, so that the variance of $\boldsymbol{y}_d(k)$ is the $p \times p$ identity matrix.

The following notations are introduced in the algorithm below: $\boldsymbol{c}_i$ and $\boldsymbol{d}_i$ denote respectively the i^{th} row of $\boldsymbol{C}$ and $\boldsymbol{D}$.

Algorithm 6.12 *Measurement and time update for the innovation filter*

Initialization: Set $\boldsymbol{P}(0) = \boldsymbol{\Pi}_0, \hat{\boldsymbol{x}}(0) = \boldsymbol{m}_0$.

At each sampling time:

1. Measurement update.
   Set $\boldsymbol{P}_0(k) = \boldsymbol{P}(k), \hat{\boldsymbol{x}}_0(k|k) = \hat{\boldsymbol{x}}(k)$.
   For $i = 1$ up to p, compute $\boldsymbol{P}_i(k|k)^{-1} = \boldsymbol{P}_{i-1}(k)^{-1} + \boldsymbol{c}_i'\boldsymbol{c}_i/q_{v,i}$.
   Set $\boldsymbol{P}(k|k) = \boldsymbol{P}_p(k|k)$.
   For $i = 1$ up to p, compute

$$\begin{aligned}\boldsymbol{K}_{f,i}(k) &= \boldsymbol{P}(k|k)\boldsymbol{c}_i'/q_{v,i}\\ \hat{\boldsymbol{x}}_i(k|k) &= \hat{\boldsymbol{x}}_{i-1}(k|k) + \boldsymbol{K}_{f,i}(k)(y_i(k) - \boldsymbol{c}_i\hat{\boldsymbol{x}}(k) - \boldsymbol{d}_i\boldsymbol{u}(k)).\end{aligned}$$

   Set $\hat{\boldsymbol{x}}(k|k) = \hat{\boldsymbol{x}}_p(k|k)$.

2. Time update.
   Compute successively

$$\begin{aligned}\boldsymbol{K}_f(k) &= \boldsymbol{P}(k|k)\boldsymbol{C}'\boldsymbol{Q}_v^{-1}\\ \boldsymbol{P}(k+1) &= \boldsymbol{A}\boldsymbol{P}(k|k)\boldsymbol{A}' + \boldsymbol{Q}_w - \boldsymbol{Q}_{wv}(\boldsymbol{Q}_v + \boldsymbol{C}\boldsymbol{P}(k)\boldsymbol{C}')^{-1}\boldsymbol{Q}_{wv}'\\ &\quad -\boldsymbol{A}\boldsymbol{K}_f(k)\boldsymbol{Q}_{wv}' - \boldsymbol{Q}_{wv}\boldsymbol{K}_f(k)'\boldsymbol{A}'\\ \hat{\boldsymbol{x}}(k+1) &= \boldsymbol{A}\hat{\boldsymbol{x}}(k|k) + \boldsymbol{Q}_{wv}(\boldsymbol{Q}_v + \boldsymbol{C}\boldsymbol{P}(k)\boldsymbol{C}')^{-1}\cdot\\ &\quad \cdot(\boldsymbol{y}(k) - \boldsymbol{C}\hat{\boldsymbol{x}}(k) - \boldsymbol{D}\boldsymbol{u}(k)).\end{aligned}$$

3. Computation of the residual.
   For $i = 1$ up to p, compute the components of the residual vector

$$r_i(k) = y_i(k) - \boldsymbol{c}_i\hat{\boldsymbol{x}}(k) - \boldsymbol{d}_i\boldsymbol{u}(k)).$$

Result: Residual vector $\boldsymbol{r}(k)$ for increasing time horizon k.

When a measurement is missing, it suffices to skip the corresponding measurement update, namely to skip the corresponding value of index i in the "for" loops. □

With unknown input. In this case, the design of an innovation filter consists of a two step procedure. First a reduced system having no unknown input is extracted from the original system. Then a steady state Kalman filter is designed for this subsystem and the candidate residual signal is nothing but the innovation associated to this filter. As above, to check whether the innovation sequence is a residual, its sensitivity to the fault vector $\boldsymbol{f}$ has to be verified, which is the object of the next subsection.

The idea behind the extraction of the subsystem will first be sketched in the case, where $\boldsymbol{E}_d = \boldsymbol{O}$ (no unknown input affecting $\boldsymbol{y}$). Next a complete algorithm will be provided to solve Problem 6.14. The justification of this algorithm is relatively involved, and the interested reader is invited to consult the bibliography for the proofs.

To extract a subsystem which has not $\boldsymbol{d}$ as input, let $\boldsymbol{x}_{sub}$ denote the state of this subsystem and set

$$\boldsymbol{x}_{sub}(k) = \boldsymbol{\Pi x}(k), \tag{6.182}$$

where $\boldsymbol{\Pi}$ is an $n_{sub} \times n$ matrix (with $n_{sub} \leq n$) to be determined. By multiplying the first Eq. (6.172) by $\boldsymbol{\Pi}$ on the left, and by taking (6.182) into account, one gets

$$\begin{aligned}\boldsymbol{x}_{sub}(k+1) &= \boldsymbol{\Pi A x}(k) + \boldsymbol{\Pi B u}(k) + \boldsymbol{\Pi F}_d \boldsymbol{d}(k) \\ &\quad + \boldsymbol{\Pi F}_f \boldsymbol{f}(k) + \boldsymbol{\Pi B}_\epsilon \boldsymbol{\epsilon}(k).\end{aligned} \tag{6.183}$$

If the following relations are imposed

$$\boldsymbol{\Pi A} = \bar{\boldsymbol{A}}\boldsymbol{\Pi} + \bar{\boldsymbol{B}}\boldsymbol{C} \tag{6.184}$$

$$\boldsymbol{\Pi F}_d = \boldsymbol{O}, \tag{6.185}$$

where $\bar{\boldsymbol{A}}$ and $\bar{\boldsymbol{B}}$ are unknown matrices to be determined, then (6.183) can be written as

$$\begin{aligned}\boldsymbol{x}_{sub}(k+1) &= \bar{\boldsymbol{A}}\boldsymbol{x}_{sub}(k) + \bar{\boldsymbol{B}}\left(\boldsymbol{y}(k) - \boldsymbol{D u}(k) - \boldsymbol{E}_f \boldsymbol{f}(k) - \boldsymbol{D}_\epsilon \boldsymbol{\epsilon}(k)\right) \\ &\quad + \boldsymbol{\Pi B u}(k) + \boldsymbol{\Pi F}_f \boldsymbol{f}(k) + \boldsymbol{\Pi B}_\epsilon \boldsymbol{\epsilon}(k)\end{aligned} \tag{6.186}$$

by using (6.182) and the output of Eq. (6.172) (in which $\boldsymbol{E}_d$ is assumed to be null). Introducing the abbreviations

$$\begin{aligned}\tilde{\boldsymbol{B}} &= \boldsymbol{\Pi B} - \bar{\boldsymbol{B}}\boldsymbol{D} \\ \tilde{\boldsymbol{F}}_f &= \boldsymbol{\Pi F}_f - \bar{\boldsymbol{B}}\boldsymbol{E}_f \\ \tilde{\boldsymbol{B}}_\epsilon &= \boldsymbol{\Pi B}_\epsilon - \bar{\boldsymbol{B}}\boldsymbol{D}_\epsilon\end{aligned}$$

into (6.186) yields

$$\boldsymbol{x}_{sub}(k+1) = \bar{\boldsymbol{A}}\boldsymbol{x}_{sub}(k) + \tilde{\boldsymbol{B}}\boldsymbol{u}(k) + \bar{\boldsymbol{B}}\boldsymbol{y}(k) + \tilde{\boldsymbol{F}}_f \boldsymbol{f}(k) + \tilde{\boldsymbol{B}}_\epsilon \boldsymbol{\epsilon}(k). \tag{6.187}$$

This system has no unknown input $\boldsymbol{d}$ as could be expected by imposing (6.185). To design a Kalman filter based on the state Eq. (6.187) when $\boldsymbol{f} = 0$, the part of the measurement $\boldsymbol{y}$ which depends on $\boldsymbol{x}_{sub}$, $\boldsymbol{u}$ and $\boldsymbol{\epsilon}$ only should be determined. This is achieved by defining the signal $\boldsymbol{y}_{sub}$ as

$$\boldsymbol{y}_{sub}(k) = \boldsymbol{M y}(k) = \boldsymbol{M C x}(k) + \boldsymbol{M D u}(k) + \boldsymbol{M D}_\epsilon \boldsymbol{\epsilon}(k), \tag{6.188}$$

where $\boldsymbol{M}$ is unknown. Imposing

$$\boldsymbol{M C} = \boldsymbol{L \Pi}, \tag{6.189}$$

(6.188) becomes

$$\boldsymbol{y}_{sub}(k) = \boldsymbol{L x}_{sub}(k) + \boldsymbol{M D u}(k) + \boldsymbol{M D}_\epsilon \boldsymbol{\epsilon}(k), \tag{6.190}$$

which has the required form.

Now, provided $(\boldsymbol{L},\ \bar{\boldsymbol{A}})$ is detectable, and $(\bar{\boldsymbol{A}},\ \tilde{\boldsymbol{B}}_\epsilon,\ \boldsymbol{L},\ \boldsymbol{M}\boldsymbol{D}_\epsilon)$ is regular, a Kalman filter can be designed for the subsystem (6.187), (6.190), when $\boldsymbol{f} = 0$

$$\begin{aligned}\hat{\boldsymbol{x}}_{sub}(k+1) \quad &= \quad \bar{\boldsymbol{A}}\hat{\boldsymbol{x}}_{sub}(k) + \tilde{\boldsymbol{B}}\boldsymbol{u}(k) + \bar{\boldsymbol{B}}\boldsymbol{y}(k) - \boldsymbol{K}_{sub}\ (\boldsymbol{y}_{sub} \\ &\quad -\boldsymbol{L}\hat{\boldsymbol{x}}_{sub}(k) - \boldsymbol{M}\boldsymbol{D}\boldsymbol{u}(k))\,,\end{aligned} \tag{6.191}$$

where

$$\boldsymbol{K}_{sub} = -(\bar{\boldsymbol{A}}\boldsymbol{P}_{sub}\,\boldsymbol{L}' + \tilde{\boldsymbol{B}}_\epsilon\,\boldsymbol{D}'_\epsilon\,\boldsymbol{M}')(\boldsymbol{M}\boldsymbol{D}_\epsilon\,\boldsymbol{D}'_\epsilon\,\boldsymbol{M}' + \boldsymbol{L}\boldsymbol{P}_{sub}\,\boldsymbol{L}')^{-1}$$

with $\boldsymbol{P}_{sub}$ the symmetric semi-positive definite solution of

$$\begin{aligned}\boldsymbol{P}_{sub} \quad = \quad & \bar{\boldsymbol{A}}\boldsymbol{P}_{sub}\,\bar{\boldsymbol{A}}' - \left(\bar{\boldsymbol{A}}\boldsymbol{P}_{sub}\,\boldsymbol{L}' + \tilde{\boldsymbol{B}}_\epsilon\,\boldsymbol{D}'_\epsilon\,\boldsymbol{M}'\right)\left(\boldsymbol{M}\boldsymbol{D}_\epsilon\,\boldsymbol{D}'_\epsilon\boldsymbol{M}'\right. \\ & \left.+\boldsymbol{L}\boldsymbol{P}_{sub}\,\boldsymbol{L}'\right)^{-1}\left(\boldsymbol{L}\boldsymbol{P}_{sub}\bar{\boldsymbol{A}}' + \boldsymbol{M}\boldsymbol{D}_\epsilon\,\tilde{\boldsymbol{B}}'_\epsilon\right) + \tilde{\boldsymbol{B}}_\epsilon\,\tilde{\boldsymbol{B}}'_\epsilon.\end{aligned}$$

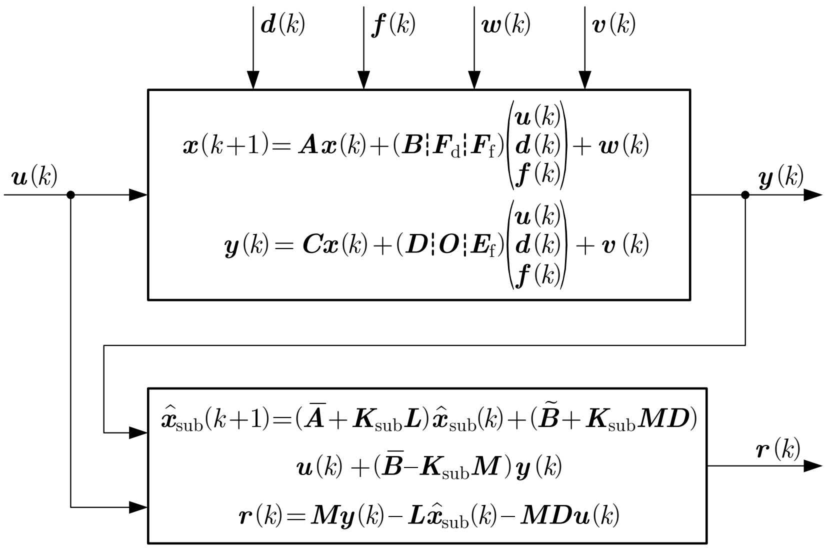


Fig. 6.17. Block diagram of the supervised system together with the innovation filter in the presence of unknown inputs

The associated output reconstruction error is given by

$$\boldsymbol{r}(k) = \boldsymbol{y}_{sub}(k) - \boldsymbol{L}\hat{\boldsymbol{x}}_{sub}(k) - \boldsymbol{M}\boldsymbol{D}\boldsymbol{u}(k)\,. \tag{6.192}$$

which can be evaluated from the available data. It can be checked that it fulfils the conditions for an innovation sequence. Indeed, the state estimation error, $\tilde{\boldsymbol{x}}_{sub}(k) = \boldsymbol{x}_{sub}(k) - \hat{\boldsymbol{x}}_{sub}(k)$, verifies the following equation obtained by subtracting (6.191) from (6.187):

$$\begin{aligned}\tilde{\boldsymbol{x}}_{sub}(k+1) \quad &= \quad \left(\bar{\boldsymbol{A}} + \boldsymbol{K}_{sub}\,\boldsymbol{L}\right)\,\tilde{\boldsymbol{x}}_{sub}\,(k) + \tilde{\boldsymbol{F}}_f\,\boldsymbol{f}(k) \\ &\quad + \left(\tilde{\boldsymbol{B}}_\epsilon + \boldsymbol{K}_{sub}\,\boldsymbol{M}\boldsymbol{D}_\epsilon\right)\boldsymbol{\epsilon}(k)\,. \end{aligned} \tag{6.193}$$

This error is clearly not affected by $\boldsymbol{d}$ and $\boldsymbol{u}$, and so is the associated innovation $\boldsymbol{r}(k) = \boldsymbol{L}\tilde{\boldsymbol{x}}_{sub}\,(k) + \boldsymbol{M}\boldsymbol{D}_\epsilon\,\boldsymbol{\epsilon}(k)$. Condition 1 of Problem 6.14 is thus fulfilled. To assure that the maximum amount of information on the fault has been kept (condition 2 of Problem 6.14), $\boldsymbol{x}_{sub}$ should have the largest possible dimension ($\boldsymbol{\Pi}$ should have the largest possible number of rows). The implementation of the innovation filter is summarised in the block diagram of Fig. 6.17

The design of an innovation filter essentially amounts to solving the set of nonlinear algebraic Eqs. (6.184), (6.185), (6.189). Despite the nonlinearity, an algorithm based only on linear algebraic operations can be derived. It is presented below in the general situation, where $\boldsymbol{E}_d \neq \boldsymbol{O}$.

Algorithm 6.13 *Innovation filter design for a system with unknown input*

Given: A system of the form (6.172).

Compute:

1. Determine the integer n_{sub} together with full row rank and full column rank matrices $\boldsymbol{\Gamma}$ and $\boldsymbol{\Phi}$ respectively such that

$$\begin{pmatrix} -z\boldsymbol{I}_{n_{sub}} + \boldsymbol{A}_{sub} & \boldsymbol{B}_{sub} \\ \boldsymbol{C}_{sub} & \boldsymbol{D}_{sub} \end{pmatrix} = \boldsymbol{\Gamma} \begin{pmatrix} -z\boldsymbol{I}_n + \boldsymbol{A} & \boldsymbol{F}_d & \boldsymbol{B}_\epsilon \\ \boldsymbol{C} & \boldsymbol{E}_d & \boldsymbol{D}_\epsilon \end{pmatrix} \cdot \begin{pmatrix} \boldsymbol{\Phi} & \boldsymbol{O} \\ \boldsymbol{O} & \boldsymbol{I}_{n_\epsilon} \end{pmatrix}.$$

The Algorithm 6.14 presented below can be used to compute $n_{sub}, \boldsymbol{\Gamma}$ and $\boldsymbol{\Phi}$. The n_{sub}-dimensional subsystem $(\boldsymbol{A}_{sub}, \boldsymbol{B}_{sub}, \boldsymbol{C}_{sub}, \boldsymbol{D}_{sub})$ has no unknown input. Let p_{sub} denote the number of rows of $\boldsymbol{C}_{sub}$.

2. Design a Kalman filter for the following reduced system:

$$\begin{aligned} \boldsymbol{x}_{sub}(k+1) &= \boldsymbol{A}_{sub}\,\boldsymbol{x}_{sub}(k) + \tilde{\boldsymbol{B}}_{sub}\,\boldsymbol{u}(k) - \boldsymbol{\Gamma}_{12}\,\boldsymbol{y}(k) \\ &\quad + \boldsymbol{B}_{sub}\,\boldsymbol{\epsilon}_{sub}(k) \\ \boldsymbol{x}_{sub}(0) &= \boldsymbol{x}_{sub,0} \\ \boldsymbol{y}_{sub}(k) &= \boldsymbol{\Gamma}_{22}\,\boldsymbol{y}(k) = \boldsymbol{C}_{sub}\,\boldsymbol{x}_{sub}(k) + \tilde{\boldsymbol{D}}_{sub}\,\boldsymbol{u}(k) \\ &\quad + \boldsymbol{D}_{sub}\,\boldsymbol{\epsilon}_{sub}(k), \end{aligned}$$

where $\boldsymbol{\epsilon}_{sub}(k)$ is a sample of a white noise sequence with variance equal to the identity matrix,

$$\boldsymbol{\Gamma} = \begin{pmatrix} \boldsymbol{\Gamma}_{11} & \boldsymbol{\Gamma}_{12} \\ \boldsymbol{\Gamma}_{21} & \boldsymbol{\Gamma}_{22} \end{pmatrix}$$

with $\boldsymbol{\Gamma}_{11}, \boldsymbol{\Gamma}_{12}, \boldsymbol{\Gamma}_{21}, \boldsymbol{\Gamma}_{22}$ respectively $n_{sub} \times n$, $n_{sub} \times p$, $p_{sub} \times n$ and $p_{sub} \times p$ -dimensional matrices,

$$\begin{aligned} \tilde{\boldsymbol{B}}_{sub} &= \boldsymbol{\Gamma}_{11}\,\boldsymbol{B} + \boldsymbol{\Gamma}_{12}\,\boldsymbol{D}, \\ \tilde{\boldsymbol{D}}_{sub} &= \boldsymbol{\Gamma}_{21}\,\boldsymbol{B} + \boldsymbol{\Gamma}_{22}\,\boldsymbol{D}. \end{aligned}$$

The resulting innovation qualifies as a residual.

Result: An innovation filter for system (6.172).

Here is the algorithm to be used in step a).

Algorithm 6.14 *Computation of* n_{sub}, $\boldsymbol{\Gamma}$, $\boldsymbol{\Phi}$

Initialisation: Let

$$\boldsymbol{Z} = \begin{pmatrix} -\boldsymbol{I}_n & \boldsymbol{O}_{n \times n_d} \\ \boldsymbol{O}_{p \times n} & \boldsymbol{O}_{p \times n_d} \end{pmatrix}, \boldsymbol{W} = \begin{pmatrix} \boldsymbol{A} & \boldsymbol{F}_d \\ \boldsymbol{C} & \boldsymbol{E}_d \end{pmatrix}.$$

Set

$$\boldsymbol{Z}^* = \boldsymbol{Z}\ ,\ \boldsymbol{W}^* = \boldsymbol{W}\ ,\ \boldsymbol{M} = \boldsymbol{I}_{n+p} \text{ and } \boldsymbol{N} = \boldsymbol{I}_{n+n_d}\ .$$

Compute:

a. While $\boldsymbol{Z}^*$ is not full column rank, do

1. perform a singular value decomposition of $\boldsymbol{Z}^*$,

$$\boldsymbol{Z}^* = \left(\boldsymbol{U}^1_{Z^*}\ \ \boldsymbol{U}^2_{Z^*}\right) \begin{pmatrix} \boldsymbol{\Sigma}_{Z^*} & \boldsymbol{O} \\ \boldsymbol{O} & \boldsymbol{O} \end{pmatrix} \begin{pmatrix} \boldsymbol{V}^1_{Z^*} \\ \boldsymbol{V}^2_{Z^*} \end{pmatrix},$$

and compress the columns of $\boldsymbol{Z}^*$ by computing the right hand side of the first equality below:

$$\left(\boldsymbol{Z}^*_1\ \ \boldsymbol{O}\right) = \boldsymbol{Z}^* \left(\boldsymbol{V}^{1'}_{Z^*}\ \ \boldsymbol{V}^{2'}_{Z^*}\right) = \left(\boldsymbol{U}^1_{Z^*}\boldsymbol{\Sigma}_{Z^*}\ \ \boldsymbol{O}\right).$$

2. Let $\left(\boldsymbol{W}^*_1\ \ \boldsymbol{W}^*_2\right) = \boldsymbol{W}^* \left(\boldsymbol{V}^{1'}_{Z^*}\ \ \boldsymbol{V}^{2'}_{Z^*}\right)$.

3. Find the highest rank full row rank matrix $\boldsymbol{L}$ satisfying $\boldsymbol{L}\boldsymbol{W}^*_2 = 0$ as follows. Perform a singular value decomposition of

$$\boldsymbol{W}^*_2\text{:}\ \boldsymbol{W}^*_2 = \left(\boldsymbol{U}^1_{W^*_2}\ \ \boldsymbol{U}^2_{W^*_2}\right) \begin{pmatrix} \boldsymbol{\Sigma}_{W^*_2} & \boldsymbol{O} \\ \boldsymbol{O} & \boldsymbol{O} \end{pmatrix} \begin{pmatrix} \boldsymbol{V}^1_{W^*_2} \\ \boldsymbol{V}^2_{W^*_2} \end{pmatrix}.$$

Noticing that

$$\begin{pmatrix} \boldsymbol{U}^{1'}_{W^*_2} \\ \boldsymbol{U}^{2'}_{W^*_2} \end{pmatrix} \boldsymbol{W}^*_2 = \begin{pmatrix} \boldsymbol{\Sigma}_{W^*_2}\boldsymbol{V}^1_{W^*_2} \\ \boldsymbol{O} \end{pmatrix},$$

one gets $\boldsymbol{L} = \boldsymbol{U}^{2'}_{W^*_2}$.

4. Let $\boldsymbol{Z}^* = \boldsymbol{L}\boldsymbol{Z}^*_1$, $\boldsymbol{W}^* = \boldsymbol{L}\boldsymbol{W}^*_1$, $\boldsymbol{M} = \boldsymbol{L}\boldsymbol{M}$, $\boldsymbol{N} = \boldsymbol{N}\boldsymbol{V}^{1'}_{Z^*}$, end do.

b. Determine an invertible matrix $\boldsymbol{T}$ such that

$$\boldsymbol{T}\boldsymbol{Z}^* = \begin{pmatrix} -\boldsymbol{I} \\ \boldsymbol{O} \end{pmatrix},$$

where the dimension of $\boldsymbol{I}$ is obviously equal to rank$\boldsymbol{Z}^*$. Such a matrix can be computed as follows:

$$\boldsymbol{T} = \begin{pmatrix} (\boldsymbol{\Sigma}_{Z^*}\boldsymbol{V}_{Z^*})^{-1} & \boldsymbol{O} \\ \boldsymbol{O} & \boldsymbol{I} \end{pmatrix} \begin{pmatrix} \boldsymbol{U}_{Z^*}^{1'} \\ \boldsymbol{U}_{Z^*}^{2'} \end{pmatrix},$$

where the notations are the same as for the singular value decomposition of $\boldsymbol{Z}^*$ above, except that $\boldsymbol{V}_{Z^*}^1 = \boldsymbol{V}_{Z^*}$ and $\boldsymbol{V}_{Z^*}^2$ does not exist since $\boldsymbol{Z}^*$ has full column rank now.

c. Set $\boldsymbol{\Gamma} = \boldsymbol{T} - \boldsymbol{M}, \quad \boldsymbol{\Phi} = \boldsymbol{N},$

$$\begin{pmatrix} \boldsymbol{A}_{sub} \\ \boldsymbol{C}_{sub} \end{pmatrix} = -\boldsymbol{\Gamma}\boldsymbol{W}\boldsymbol{\Phi}, \quad \begin{pmatrix} \boldsymbol{B}_{sub} \\ \boldsymbol{D}_{sub} \end{pmatrix} = \boldsymbol{\Gamma}\begin{pmatrix} \boldsymbol{B}_{\epsilon} \\ \boldsymbol{D}_{\epsilon} \end{pmatrix}.$$

The above design procedure may fail in different ways:

- When the dimensions of $\boldsymbol{\Gamma}$ and $\boldsymbol{\Phi}$ are such that $\begin{pmatrix} \boldsymbol{A}_{sub} \\ \boldsymbol{C}_{sub} \end{pmatrix}$ is a square matrix, the obtained subsystem has no output, and hence no Kalman filter can be designed and no residual generator can be obtained. This typically occurs when $n_d \geq p$.
- When

$$\begin{pmatrix} z\boldsymbol{I} + \boldsymbol{A}_{sub} & \boldsymbol{B}_{sub} \\ \boldsymbol{C}_{sub} & \boldsymbol{D}_{sub} \end{pmatrix}$$

  has full generic rank, but it looses rank for $z = \exp(-j\omega)$, $\omega \in$ IR, then it is not possible to design a residual generator as the regularity assumption needed for the design of the Kalman filter is not fulfilled. [6]
- When the regularity assumption ceases to be met due to $\boldsymbol{B}_{sub} = \boldsymbol{O}, \boldsymbol{D}_{sub} = \boldsymbol{O}$ or due to

$$\begin{pmatrix} -z\boldsymbol{I} + \boldsymbol{A}_{sub} & \boldsymbol{B}_{sub} \\ \boldsymbol{C}_{sub} & \boldsymbol{D}_{sub} \end{pmatrix}$$

  having not full generic rank, the design is more involved and the reader is referred to the bibliography for this case.

Example 6.15 *Innovation filter design for the ship example*

Let us consider the linearised augmented ship-steering model described by combining the wave model and the ship-steering system

[6] Fulfilment of this condition can be checked by computing the zeros of system $(\boldsymbol{A}_{sub}, \boldsymbol{B}_{sub}, \boldsymbol{C}_{sub}, \boldsymbol{D}_{sub})$ and by verifying that none of them lies on the unit circle.

$$\begin{pmatrix} \dot{x}_{w1} \\ \dot{x}_{w2} \\ \dot{\omega}_3 \\ \dot{\psi} \end{pmatrix} = \begin{pmatrix} -2\eta_\omega \sigma_0 & -\sigma_0^2 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & b\eta_1 & 0 \\ 1 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} x_{w1} \\ x_{w2} \\ \omega_3 \\ \psi \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ b \\ 0 \end{pmatrix} \delta + \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} w_\omega$$

$$\begin{pmatrix} \omega_{3m} \\ \psi_m \end{pmatrix} = \begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_{w1} \\ x_{w2} \\ \omega_3 \\ \psi \end{pmatrix} + \begin{pmatrix} f_\omega \\ f_\psi \end{pmatrix} + \begin{pmatrix} \nu_\omega \\ \nu_\psi \end{pmatrix}.$$

A sampled-data model of this system has been obtained at a sampling rate of $0.5Hz$. The resulting equations are:

$$\begin{pmatrix} x_{w1}(k+1) \\ x_{w2}(k+1) \\ \omega_3(k+1) \\ \psi(k+1) \end{pmatrix} = \begin{pmatrix} -0.1281 & -0.6365 & 0 & 0 \\ 0.9945 & 0.1106 & 0 & 0 \\ 0 & 0 & 0.0000 & 0 \\ 0.9945 & -0.8894 & 0.0500 & 1.0000 \end{pmatrix} \begin{pmatrix} x_{w1}(k) \\ x_{w2}(k) \\ \omega_3(k) \\ \psi(k) \end{pmatrix}$$
$$+ \begin{pmatrix} 0 \\ 0 \\ 0.0500 \\ 0.0975 \end{pmatrix} \delta(k) + \boldsymbol{w}(k) \tag{6.194}$$

$$\begin{pmatrix} \omega_{3m}(k) \\ \psi_m(k) \end{pmatrix} = \begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_{w1}(k) \\ x_{w2}(k) \\ \omega_3(k) \\ \psi(k) \end{pmatrix}$$
$$+ \begin{pmatrix} f_\omega(k) \\ f_\psi(k) \end{pmatrix} + \begin{pmatrix} v_\omega(k) \\ v_\psi(k) \end{pmatrix}. \tag{6.195}$$

The covariance matrix of the state noise $\boldsymbol{w}(k)$ can be evaluated by the sampling procedure described in Appendix 2. It yields

$$E(\boldsymbol{w}(k)\boldsymbol{w}(k)') = \boldsymbol{Q}_w = \begin{pmatrix} 0.0015 & 0.0056 & 0.0019 & 0.0056 \\ 0.0056 & 0.0322 & 0.0077 & 0.0322 \\ 0.0019 & 0.0077 & 0.0024 & 0.0077 \\ 0.0056 & 0.0322 & 0.0077 & 0.0322 \end{pmatrix}.$$

The measurement noise sequence is characterised by a covariance matrix given as

$$\boldsymbol{Q}_v = \begin{pmatrix} 0.0001 & 0 \\ 0 & 0.005 \end{pmatrix}.$$

State and measurement noise are supposed to be uncorrelated, hence $\boldsymbol{Q}_{wv} = \boldsymbol{O}$.

The considered input signal $\delta(t)$ is a sine wave with period 20π seconds.

Figure 6.18 gives the evolution of the sampled output signals in healthy working mode (first 300 samples), when a 0.1 deg/s bias on the turn rate $\omega_3(k)$ is added (from sample 301 to sample 600), and when this bias disappears bringing the system back to healthy working mode (sample 601 to 900) In other words, a step-like fault f_ω occurs between sample 301 and 600.

From the above model, the following Kalman filter is deduced [7].

[7] The gain of this filter can be computed by MATLAB function *dlqe* for instance.

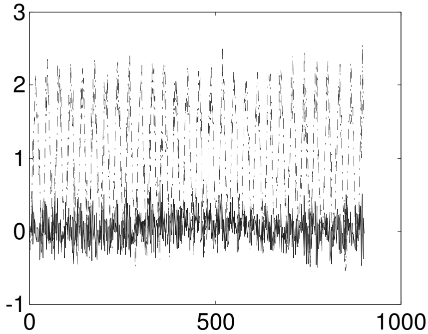


Fig. 6.18. Sampled output sequence of ship model in healthy and faulty working modes; ω_{3m} as a function of sample number (continuous line), ψ_m as a function of sample number (dash-dotted line)

$$\begin{pmatrix} \hat{x}_{w1}(k+1) \\ \hat{x}_{w2}(k+1) \\ \hat{\omega}_3(k+1) \\ \hat{\psi}(k+1) \end{pmatrix} = \begin{pmatrix} -0.1281 & -0.6365 & 0 & 0 \\ 0.9945 & 0.1106 & 0 & 0 \\ 0 & 0 & 0.0000 & 0 \\ 0.9945 & -0.8894 & 0.0500 & 1.0000 \end{pmatrix} \begin{pmatrix} \hat{x}_{w1}(k) \\ \hat{x}_{w2}(k) \\ \hat{\omega}_3(k) \\ \hat{\psi}(k) \end{pmatrix}$$
$$+ \begin{pmatrix} 0 \\ 0 \\ 0.0500 \\ 0.0975 \end{pmatrix} \delta(k) + \begin{pmatrix} -0.3265 & -0.4544 \\ 0.6710 & 0.0067 \\ 0.0000 & 0.0000 \\ 0.6880 & 0.0119 \end{pmatrix}$$
$$\cdot \left(\begin{pmatrix} \omega_{3m}(k) \\ \psi_m(k) \end{pmatrix} - \begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \hat{x}_{w1}(k) \\ \hat{x}_{w2}(k) \\ \hat{\omega}_3(k) \\ \hat{\psi}(k) \end{pmatrix} \right). \tag{6.196}$$

The innovation is computed from

$$r(k) = \begin{pmatrix} \omega_{3m}(k) \\ \psi_m(k) \end{pmatrix} - \begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \hat{x}_{w1}(k) \\ \hat{x}_{w2}(k) \\ \hat{\omega}_3(k) \\ \hat{\psi}(k) \end{pmatrix}. \tag{6.197}$$

The innovation sequences for the data of Fig. 6.18 is plotted in Fig. 6.19. The change in the mean of the innovation sequence due to the fault is visible. However, such a change cannot be detected by comparing the signals to a simple threshold. □

The existence of a filter that meets the conditions in Problem 6.14 does not guarantee that the filter output (namely the innovation) is useful for fault detection. It should be affected by $\boldsymbol{f}$ in order to meet the second condition of Problem 6.13, and thus to be suitable as a residual. This issue is addressed in the next subsection.

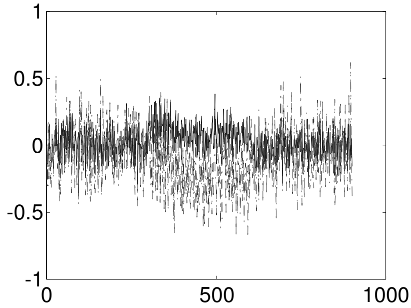


Fig. 6.19. Innovation sequences computed by (6.196), (6.197) from the data of Fig. 6.18; first component (continuous line); second component (dash-dotted line)

Sensitivity to faults or fault detectability. There are several ways to define the sensitivity of an innovation signal to a fault $\boldsymbol{f}$ or, equivalently, the detectability of a fault by a given innovation signal.

In a similar way as for the deterministic case, an innovation filter for system (6.172) is said to be *fault sensitive* if its output is affected by $\boldsymbol{f}$. Equivalently, the fault is said to be detectable in this case.

If the transfer function from $\boldsymbol{f}$ to $\boldsymbol{r}$ is left invertible, then the innovation filter is *strictly fault sensitive*.

It can be shown that system (6.172) has a (strict) fault sensitive innovation filter which solves Problem 6.14 if and only if every innovation filter solution of Problem 6.14 is (strictly) fault sensitive. Thus (strict) fault sensitivity is a property of the supervised system (6.172), and it does not depend upon the choice of innovation filter. Therefore, (strict) fault sensitivity of (6.172) will be referred to in the sequel.

Assuming the pair $(\boldsymbol{C}_{sub}, \boldsymbol{A}_{sub})$ resulting from the design procedure is observable, the following necessary and sufficient conditions for sensitivity can be exploited.

System (6.172) is fault sensitive if and only if [8]:

$$\Im \begin{pmatrix} \boldsymbol{F}_f \\ \boldsymbol{E}_f \end{pmatrix} \not\subset Ker(\boldsymbol{\Gamma})\,. \tag{6.198}$$

System (6.172) is strictly fault sensitive if and only if system $(\boldsymbol{A}_{sub}$, $\boldsymbol{F}_{f,sub}$, $\boldsymbol{C}_{sub}$, $\boldsymbol{E}_{f,sub})$, where

$$\begin{pmatrix} \boldsymbol{F}_{f,sub} \\ \boldsymbol{E}_{f,sub} \end{pmatrix} = \boldsymbol{\Gamma} \begin{pmatrix} \boldsymbol{F}_f \\ \boldsymbol{E}_f \end{pmatrix} \tag{6.199}$$

is left invertible.

[8] The image (space) $\Im(\boldsymbol{X})$ of a linear transformation associated to the $n \times m$ matrix $\boldsymbol{X}$ is the set of all vectors $\boldsymbol{y}$ in IR^n that equal $\boldsymbol{X}\boldsymbol{u}$ for some $\boldsymbol{u}$ in IR^m. The kernel (or null space) $Ker(\boldsymbol{X})$ of a linear transformation associated to the $n \times m$ matrix $\boldsymbol{X}$ is the set of all vectors $\boldsymbol{u}$ in IR^m that fulfil $\boldsymbol{X}\boldsymbol{u} = 0$

Yet another notion is strong fault sensitivity, which is typically considered for scalar faults. As for a deterministic residual, the innovation signal is strongly fault sensitive when it reaches a non-zero steady state value for a step-like fault, $f(k) = \bar{f} 1_{\{k>k_0\}}$, for any constant non-zero $\bar{f}$. This property can be checked a posteriori by computing the steady state gain of the transfer function between fault f and innovation $\boldsymbol{r}$ and verifying that it has at least one non-zero entry.

Remark 6.17 *Comments on strong fault detectabilty*
In a deterministic framework, necessary and sufficient conditions for the existence of a residual generator which is strongly fault sensitive for a given system have been developed [187]. The corresponding fault is said to be strongly detectable. It is unclear whether the innovation signal computed as the output of the filter (6.175) (or as the innovation of a Kalman filter for the subsystem in step 2 of the algorithm 6.13) is necessarily strongly fault sensitive, when a strongly detectable fault is considered. □

Distribution of the residual vector and residual evaluation. For proper choice of the residual evaluation method, it is necessary to analyse the statistical distribution of $\boldsymbol{r}(k)$. For the sake of simplicity, the situation, where $\boldsymbol{x}_0$, $\boldsymbol{v}(k)$, $\boldsymbol{w}(k)$, $k = 0, 1, \ldots$, are normally distributed is considered. Then, the residual has asymptotically (as k tends to infinity) a Gaussian distribution with known variance and with zero mean or non-zero mean, depending on whether $\boldsymbol{f}(k)$ asymptotically vanishes or not assuming the fault is strongly detectable. The normal distribution results from the linearity of the filter and the supervised process.

In order to characterise this distribution, let us consider the situation, where there is no unknown input, and hence the residual generator is given by (6.175). The reasoning below also applies to the Kalman filter designed for the system given in step 2 of the algorithm 6.13, but the notations are more cumbersome. The first two moments of the distribution of $\boldsymbol{r}(k)$ can be computed as follows. Let $\tilde{\boldsymbol{x}}(k) = \boldsymbol{x}(k) - \hat{\boldsymbol{x}}(k)$. Then classical results on steady state Kalman filters provide the following expression for the mean and the variance of $\tilde{\boldsymbol{x}}(k)$ in the absence of fault:

$$\begin{aligned} \lim_{k\to\infty} E(\tilde{\boldsymbol{x}}(k)) &= 0 \\ \lim_{k\to\infty} E\left(\tilde{\boldsymbol{x}}(k)\, \tilde{\boldsymbol{x}}(k)'\right) &= \boldsymbol{P}, \end{aligned}$$

with $\boldsymbol{P}$ given as the semi-positive definite solution of (6.177). By substituting the second equation of (6.172) (with $\boldsymbol{E}_d = \boldsymbol{O}$) for $\boldsymbol{y}(k)$ in the expression of $\boldsymbol{r}(k)$ (6.175), the residual can be written as

$$\boldsymbol{r}(k) = \boldsymbol{C}\tilde{\boldsymbol{x}}(k) + \boldsymbol{E}_f\, \boldsymbol{f}(k) + \boldsymbol{D}_\epsilon\, \boldsymbol{\epsilon}(k). \tag{6.200}$$

When $\boldsymbol{f}(k)$ vanishes as k tends to infinity, one deduces from (6.200) with $\boldsymbol{f}(k) = 0$:

$$\begin{aligned} \boldsymbol{r}_m &= \lim_{k\to\infty} E\left(\boldsymbol{r}(k)\right) = 0 \\ \boldsymbol{Q}_r &= \lim_{k\to\infty} E\left(\left(\boldsymbol{r}(k) - \boldsymbol{r}_m\right)\left(\boldsymbol{r}(k) - \boldsymbol{r}_m\right)'\right) = \boldsymbol{CPC}' + \boldsymbol{D}_\epsilon\, \boldsymbol{D}_\epsilon'. \end{aligned}$$

If, on the contrary $\lim_{k\to\infty} \boldsymbol{f}(k) = \bar{\boldsymbol{f}} \neq 0$, the residual mean is non-zero. It can be obtained from the transfer function between $\boldsymbol{f}(z)$ and $\boldsymbol{r}(z)$ deduced from (6.172) and (6.178), namely $\boldsymbol{V}_{ry}(z)\left(\boldsymbol{C}(z\boldsymbol{I}-\boldsymbol{A})^{-1}\boldsymbol{F}_f+\boldsymbol{E}_f\right)$. Indeed, the mean of the residual is nothing but the steady state value of the residual for $\boldsymbol{f}(k) = \bar{\boldsymbol{f}}$. Thus,

$$\boldsymbol{r}_m = \lim_{k\to\infty} E(\boldsymbol{r}(k)) = \boldsymbol{V}_{ry}(1)\left(\boldsymbol{C}(\boldsymbol{I}-\boldsymbol{A})^{-1}\boldsymbol{F}_f+\boldsymbol{E}_f\right)\bar{\boldsymbol{f}}.$$

Stability of the supervised system is implicitly assumed when writing this expression. The variance of the residual is unchanged, since the fault signal is considered as deterministic.

The problem of fault detection thus amounts to deciding between the following two hypotheses

$$\mathcal{H}_0: \quad \mathcal{L}(\boldsymbol{r}(k)) = As\,\mathcal{N}(0,\, \boldsymbol{Q}_r) \tag{6.201}$$

$$\mathcal{H}_1: \quad \mathcal{L}(\boldsymbol{r}(k)) = As\,\mathcal{N}\left(\boldsymbol{V}_{ry}(1)\left(\boldsymbol{C}(\boldsymbol{I}-\boldsymbol{A})^{-1}\boldsymbol{F}_f+\boldsymbol{E}_f\right)\bar{\boldsymbol{f}},\, \boldsymbol{Q}_r\right), \tag{6.202}$$

where the notation $\mathcal{L}(\boldsymbol{r}(k))$ denotes the distribution of $\boldsymbol{r}(k)$, and

$$\mathcal{L}(\boldsymbol{r}(k)) = As\,\mathcal{N}(\boldsymbol{a},\, \boldsymbol{X})$$

indicates that this distribution is normal with mean $\boldsymbol{a}$ and variance $\boldsymbol{X}$ as k tends to infinity. Notice that the residual must be strongly sensitive to fault $\boldsymbol{f}$ for the distributions under $\mathcal{H}_0$ and $\mathcal{H}_1$ to be different.

The asymptotic character of (6.201), (6.202) is due to the effects of initial conditions and filter transients upon occurrence of a fault. Neglecting such transients, and assuming that $\bar{\boldsymbol{f}}$ is known, one can recast the above problem as the following test of hypotheses.

Problem 6.15 (Test of hypotheses: transient not accounted for)
Given a sequence of residual vectors $\boldsymbol{r}(1),\ \ldots,\ \boldsymbol{r}(k)$, obtained as the output of filter (6.175), choose between the following two hypotheses at the current time instant k:

$\mathcal{H}_0$: $\mathcal{L}(\boldsymbol{r}(i)) = \mathcal{N}(0,\, \boldsymbol{CPC}' + \boldsymbol{D}_\epsilon \boldsymbol{D}'_\epsilon)$ *for* $1 \le i \le k$,

$\mathcal{H}_1$: *From time instant 1 up to an unknown time instant* k_0, $\boldsymbol{r}(i)$, $i = 1,\ \ldots,\ k_0-1$ *is distributed as*

$$\mathcal{L}(\boldsymbol{r}(i)) = \mathcal{N}(0,\, \boldsymbol{CPC}' + \boldsymbol{D}_\epsilon \boldsymbol{D}'_\epsilon)$$

while for time instant $i \ge k_0$

$$\mathcal{L}(\boldsymbol{r}(i)) = \mathcal{N}\left((\boldsymbol{V}_{ry}(1)\,\boldsymbol{C}(\boldsymbol{I}-\boldsymbol{A})^{-1}\boldsymbol{F}_f+\boldsymbol{E}_f)\bar{\boldsymbol{f}},\, \boldsymbol{CPC}' + \boldsymbol{D}_\epsilon\boldsymbol{D}'_\epsilon\right).$$

This problem is of the form of a change detection in the mean of a Gaussian vector sequence (Problem 6.9) with $\boldsymbol{z}(i)$ replaced by $\boldsymbol{r}(i)$, $\boldsymbol{\mu}_0 = 0$, $\boldsymbol{Q} = \boldsymbol{CPC}' + \boldsymbol{D}_\epsilon \boldsymbol{D}'_\epsilon$ and $\boldsymbol{\mu}_1 = \boldsymbol{\beta} = \left(\boldsymbol{V}_{ry}(1)\,\boldsymbol{C}(\boldsymbol{I}-\boldsymbol{A})^{-1}\boldsymbol{F}_f+\boldsymbol{E}_f\right)\bar{\boldsymbol{f}}$. Hence, the CUSUM algorithm based on a step-like change can be used for residual evaluation, with $\bar{f}$ taken as twice the minimum magnitude of the fault to be detected or as the most likely magnitude of this fault. The complete fault detection system is depicted in Fig. 6.20.

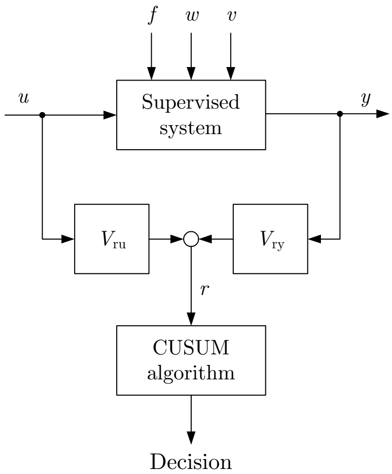


Fig. 6.20. Fault detection system

Remark 6.18 χ^2*-test*
In some applications, particularly in the area of predictive maintenance, the delay for detection may not be a crucial factor, and one may resort to an off-line approach to solve a simplified version of the above hypotheses testing problem. The most recent data over a sliding window $[k-M+1,\, k]$ are considered, and the time instant k_0 is set to 1, which amounts to considering that the change has affected all elements of the batch of data. The method to solve this hypotheses testing problem relies on the χ^2-test which is presented in the appendix for a scalar data set. □

When stating the above hypotheses testing problem, the transient of the system and the residual generator upon occurrence of a fault are not taken into account. This may significantly affect the detection delay. If a priori knowledge on the fault sequence $\boldsymbol{f}(i),\, i = k_0,\, k_0+1,\, \ldots$ is available, the performance of the detection system can be improved by introducing a suitable dynamical profile of change in the CUSUM algorithm.

Most commonly, step-like changes in the fault sequence are considered, namely

$$\begin{array}{lllll} \boldsymbol{f}(i) & = & 0 & i & = & 1,\, 2,\, \ldots,\, k_0-1 \\ \boldsymbol{f}(i) & = & \bar{\boldsymbol{f}} & i & \geq & k_0 \end{array} \qquad (6.203)$$

or, in a compact way, $\boldsymbol{f}(i) = \bar{\boldsymbol{f}}_{\{i \geq k_0\}}$, where $\bar{\boldsymbol{f}}$ is a constant vector.

Due to the linearity of the system (6.172) and the filter (6.175), the residual sequence can be written as

$$\boldsymbol{r}(k) = \boldsymbol{r}_0(k) + \boldsymbol{\rho}\,(k,\, k_0), \qquad (6.204)$$

where $\boldsymbol{r}_0(k)$ is the value of the residual in the absence of fault, and $\boldsymbol{\rho}\,(k,\, k_0)$ is the contribution to $\boldsymbol{r}(k)$ of a fault occurring at time $k_0 \leq k$. In the case of a step-like fault considered above, $\boldsymbol{\rho}(k,\, k_0)$ can be computed easily; it only depends on the difference $k - k_0$, and hence, is written with an abuse of notation $\boldsymbol{\rho}(k,\, k_0) =$

$\boldsymbol{\rho}(k-k_0)$. For the sake of simplicity, only a scalar fault sequence is considered. Then, $\boldsymbol{\rho}\,(k-k_0) = \tilde{\boldsymbol{\rho}}\,(k-k_0)\bar{f}$, where $\tilde{\boldsymbol{\rho}}\,(k)$ is the response of system (6.172), (6.175) to a fault signal of the form (6.203) with $\bar{f} = 1$, for $\boldsymbol{u}(k) = 0,\ \boldsymbol{d}(k) = 0$ and $\boldsymbol{\epsilon}(k) = 0$ for all $k > 0$, and for zero initial conditions. It coincides with the step response of the system with transfer function $\boldsymbol{V}_{ry}(z)\ \left(\boldsymbol{C}(z\boldsymbol{I}-\boldsymbol{A})^{-1}\boldsymbol{E}_f + \boldsymbol{F}_f\right)$. The hypotheses testing problem when taking into account the dynamical profile of the change can be written as

Problem 6.16 (Test of hypotheses: transient accounted for)
Given a sequence of residual vectors $\boldsymbol{r}(1),\ \ldots,\ \boldsymbol{r}(k)$, obtained as the output of filter (6.175), choose between the following two hypotheses at the current time instant k

$\mathcal{H}_0$: $\mathcal{L}(\boldsymbol{r}(i)) = \mathcal{N}(0,\ \boldsymbol{CPC}' + \boldsymbol{D}_\epsilon\,\boldsymbol{D}_\epsilon')$ *for* $1 \le i \le k$,
$\mathcal{H}_1$: *From time instant 1 up to an unknown time instant* k_0, $\boldsymbol{r}(i)$, $i = 1,\ \ldots,\ k_0 - 1$ *is distributed as*

$$\mathcal{L}\,(\boldsymbol{r}(i)) = \mathcal{N}(0,\ \boldsymbol{CPC}' + \boldsymbol{D}_\epsilon\,\boldsymbol{D}_\epsilon') \tag{6.205}$$

while for time instant $i \ge k_0$,

$$\mathcal{L}\,(\boldsymbol{r}(i)) = \mathcal{N}(\tilde{\boldsymbol{\rho}}\,(i-k_0)\bar{f},\ \boldsymbol{CPC}' + \boldsymbol{D}_\epsilon\,\boldsymbol{D}_\epsilon'). \tag{6.206}$$

This problem is in the form of Problem 6.10. (6.205), (6.206) precisely have the form (6.154), (6.155) with $\boldsymbol{r}(i)$ replacing $\boldsymbol{z}(i)$, $\boldsymbol{CPC}' + \boldsymbol{D}_\epsilon\,\boldsymbol{D}_\epsilon'$ replacing $\boldsymbol{Q}$, $\tilde{\boldsymbol{\rho}}\,(i - k_0)\bar{f}$ replacing $\boldsymbol{\rho}(i-k_0)$ and $\boldsymbol{\mu}_0 = 0$. The CUSUM algorithm based on a known dynamical profile of change can thus be applied with $\boldsymbol{\rho}(k) = \tilde{\boldsymbol{\rho}}\,(k)\bar{f}$, where $\bar{f}$ is taken as twice the minimum magnitude of the change to be detected or as the most likely magnitude of this change.

Remark 6.19 *Delay in dynamic profile*
In the statement of problem 6.10, $\rho(j)$ is supposed to be different from zero for $j > 0$. This hypothesis is not verified when the transfer function $\boldsymbol{V}_{ry}(z)\ \left(\boldsymbol{C}(z\boldsymbol{I}-\boldsymbol{A})^{-1}\boldsymbol{E}_f + \boldsymbol{F}_f\right)$ has no direct feedthrough term. In this case, one should use $\rho(k) = \tilde{\rho}(k-\tau)\bar{f}$, where τ denotes the minimum delay in the n_r elements of the mentioned transfer function. □

Notice that strong fault sensitivity is no more a required property of the residual in order to achieve fault detection, when the dynamical profile of the change is accounted for. Indeed, it suffices that the distributions (6.205), (6.206) be different for some time interval. Checking that the fault subsists by reinitialisation of the CUSUM algorithm is however impossible when the residual is not strongly fault sensitive.

Remark 6.20 *Fault sequence*
The choice of a step-like fault sequence can be made without loss of generality. Indeed, other signal forms could possibly be represented as the step response of a linear system, and this linear model could be included in the state-space Eqs. (6.171). □

Example 6.15 (cont.) *Ship example*

The CUSUM algorithm based on the knowledge of the dynamical profile of change will be used to detect the occurrence of fault f_ω. In order to determine the dynamical profile of the change to be used in the algorithm, it suffices to consider the response of the system made of Eqs. (6.194), (6.195), (6.196), (6.197) to a step-like fault f_ω, keeping all other input signals equal to zero and starting with zero initial conditions. This corresponds to the step response of the transfer function $\boldsymbol{V}_{ry}(z) = \left(\boldsymbol{C}\,(z\boldsymbol{I} - \boldsymbol{A})^{-1}\boldsymbol{E}_f + \boldsymbol{F}_f\right)$ with respect to the first input.

Given the specifications, one decides that the smallest bias on ω_3 to be detected is 0.025 deg/s. $\bar{f}_{\omega_3}$ is set to twice this value, which yields 0.05 deg/s. A step of magnitude 0.05 deg/s is thus applied as signal for f_ω. The vector dynamical profile of change with respect to fault f_ω , $0.05\,\tilde{\boldsymbol{\rho}}_{\omega_3}$ is plotted in Fig. 6.21.

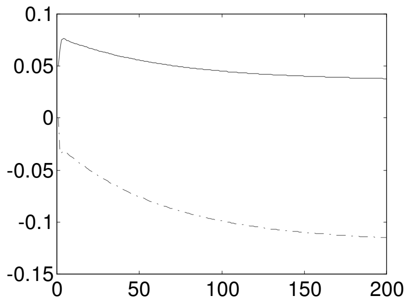


Fig. 6.21. Dynamic profile of change for fault f_ω; first component of $0.05\,\tilde{\rho}_{\omega_3}$ (continuous line); second component (dash-dotted line)

The evolution of the CUSUM decision functions for detection of f_ω, g_{ω_3} is plotted in Fig. 6.22. The indicated threshold (dashed line) has been set on the basis of the value of the decision function for the first 300 samples (healthy working mode). The reinitialisation policy is the reset procedure indicated in the description of the algorithm. One notices the repeated threshold crossing of the decision function g_{ω_3} while the fault is present (from sample 300 to 600). □

6.8.3 Fault estimation

In this section, a model of the form (6.171) in which $n_f = 1$ is considered. Besides, it is assumed that step-like faults of unknown magnitude occur. Thus a scalar sequence $f(i)$, $i = 1, 2, \ldots$ of the form (6.203) with an unknown constant $\bar{f}$ is assumed.

The problem can be stated as:

Problem 6.17 (Fault estimation)
Given

1. *a model of the process of the form (6.171) subject to a scalar step-like fault sequence* $f(i) = \bar{f}1_{\{i \geq k_0\}}$ *of unknown magnitude* $\bar{f}$.

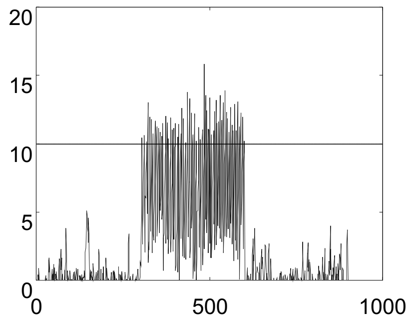


Fig. 6.22. CUSUM decision function resulting from application to the innovation sequence of Fig. 6.19 of the CUSUM algorithm based on the known dynamical profile of change (Fig. 6.21)

2. *a sequence of measured process input and output values:*

$$(\boldsymbol{y}(i),\ \boldsymbol{u}(i))_{1\leq i\leq k}\ ,$$

where k denotes the current time instant.

Choose between the following two hypotheses

$\mathcal{H}_0$*: healthy operating condition,*
$\mathcal{H}_1$*: faulty operating condition.*

When $\mathcal{H}_1$ is selected, an estimate of the fault occurrence time, $\hat{k}_0$, and of the fault magnitude, $\hat{\bar{f}}$, should be provided.

As for the fault detection problem, a two step procedure is used to solve this problem. The first step, namely the residual generation, is the same for both problems. For residual evaluation, a generalised likelihood ratio algorithm is used to obtain an estimate of the fault magnitude. Indeed, given the specific fault model, the residual evaluation reduces to Problem 6.16 in which $\bar{f}$ is unknown. Hence, it is of the form of Problem 6.11. (6.205), (6.206) precisely have the form (6.161), (6.162) with $\boldsymbol{r}(i)$ replacing $\boldsymbol{z}(i)$, $\boldsymbol{CPC}' + \boldsymbol{D}_\epsilon\,\boldsymbol{D}_\epsilon'$ replacing $\boldsymbol{Q}$, $\bar{f}$ replacing ν and $\boldsymbol{\mu}_0 = 0$. The GLR algorithm based on a known dynamical profile of change but an unknown fault magnitude can thus directly be used to process the residual vector in order to obtain an on-line solution to Problem 6.17.

Example 6.15 (cont.) *Ship example*

Let us again consider the innovation sequence depicted in Fig. 6.19. Instead of using a CUSUM algorithm, we now perform a GLR algorithm on this sequence. A dynamical profile of change has to be provided. It can be computed as for the CUSUM algorithm and one gets a profile similar to Fig. 6.21 except that the minimum fault magnitude is not accounted for. Thus to obtain the dynamical profile $\tilde{\rho}_{\omega 3}$, the signal f_ω which is used is a step function with unit magnitude instead of the magnitude of 0.05 deg/s used previously.

M is chosen as 50. This allows one to determine a quite precise estimate of the fault magnitude in the reinitialisation procedure. Given the values of the decision function obtained for the first 300 data, which correspond to the set $\{z_0(1), \cdots, z_0(N_0)\}$, and given its values upon occurrence of the fault, the threshold h is set to 30. The evolution of the GLR decision function is plotted in Fig. 6.23. Each time the threshold is crossed, an alarm is generated, and the decision function remains equal to zero until enough data are available for estimating the fault magnitude in a reliable way. The recursive algorithm restarts at $\hat{k}_0 + M$.

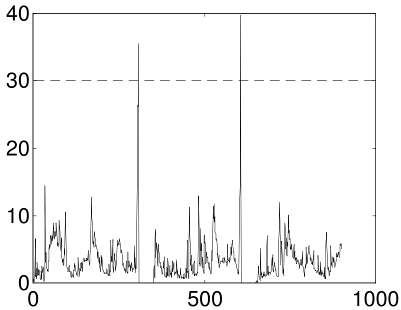


Fig. 6.23. GLR decision function resulting from application to the innovation sequence of Fig. 6.19 of the algorithm with known dynamical profile of change

Note that successive changes separated by less than M samples cannot be handled properly. For the considered data, an alarm is generated at time instants 308 and 606. The estimated change times are 300 and 601 while the actual changes occur at 301 and 601. All numbers should be multiplied by the sampling period to obtain time in seconds. The estimates of the change magnitude used for reinitilisation are respectively 0.1020 for the positive change and -0.1073 for the negative change (disappearance of the fault). Remember that the actual change magnitude is 0.1 in both cases. Notice that the estimate of the change magnitude plotted in Fig. 6.24 converges relatively fast after occurrence of a fault. Hence the horizon M could possibly be chosen smaller for this situation, yet this value is used to make the convergence of the estimate visible in the plot.

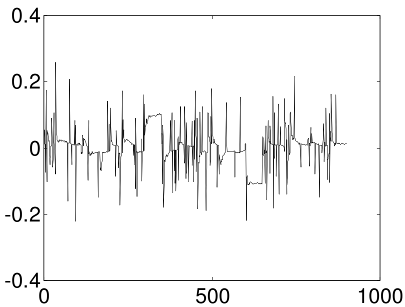


Fig. 6.24. Estimate of the change magnitude resulting from application to the innovation sequence of Fig. 6.19 of the GLR algorithm with known dynamical profile of change

6.8.4 Fault isolation

Up to now the plant model used in the stochastic framework only contained one single (possibly vector) fault to be detected. However, most often several faults may affect the behaviour of the supervised process, and one should not only detect them, but also isolate the faulty components. An appropriate model to describe the process then takes the form

$$\begin{aligned} \boldsymbol{x}(k+1) &= \boldsymbol{A}\boldsymbol{x}(k) + \boldsymbol{B}\boldsymbol{u}(k) + \sum_{i=1}^{n_f} \boldsymbol{F}_i f_i(k) + \boldsymbol{B}_\epsilon\, \boldsymbol{\epsilon}(k) \\ \boldsymbol{x}(0) &= \boldsymbol{x}_0 \\ \boldsymbol{y}(k) &= \boldsymbol{C}\boldsymbol{x}(k) + \boldsymbol{D}\boldsymbol{u}(k) + \sum_{i=1}^{n_f} \boldsymbol{E}_i f_i(k) + \boldsymbol{D}_\epsilon\, \boldsymbol{\epsilon}(k), \end{aligned} \tag{6.207}$$

where, for the sake of simplicity, scalar faults f_i, $i = 1, \ldots, n_f$ are considered. One way to detect and isolate a single fault, say f_α, is to design a residual which is only sensitive to that fault and to evaluate it in an appropriate way. This can be achieved by recasting the problem as a fault detection problem for a specific system with unknown input. Obviously, the unknown input vector must be made of the faults not to be detected, and thus the model takes the form (6.171) with $\boldsymbol{d} = \left[f_1, \ldots, f_{\alpha-1}, f_{\alpha+1}, \ldots, f_{n_f}\right]'$ and $f = f_\alpha$. One can now proceed as in the section on fault detection (and estimation) to build a system that detects, isolates and possibly estimates fault f_α.

If each fault must be detected and isolated, one solution is to solve n_f fault detection (and estimation) problems of the form just mentioned. This yields a bank of residual generators, each one being affected by a single fault. The table below represents the situation when $n_f = 3$.

Table 6.4 Effects of the faults on the residuals

	$\boldsymbol{f}_1$	$\boldsymbol{f}_2$	$\boldsymbol{f}_3$
$\boldsymbol{r}_1$	$\times$	0	0
$\boldsymbol{r}_2$	0	$\times$	0
$\boldsymbol{r}_3$	0	0	$\times$

where a $\times$ indicates that the fault in the corresponding column affects the residual of the corresponding row. Each residual can be processed individually by a GLR algorithm or a CUSUM algorithm, according as an estimate of the fault magnitude is needed or not.

From the conditions for fault detectability, the following necessary conditions can be deduced for the above scheme to work

$$\operatorname{rank}\left(\left(\boldsymbol{H}_{yf_\ell}(s)\;\; \boldsymbol{H}_{yf_j}(s)\right)\right) > \operatorname{rank}\, \boldsymbol{H}_{yf_j}(s) \tag{6.208}$$
$$\text{for all } \ell, j = 1, \ldots, n_f, \ell \neq j,$$

where $\boldsymbol{H}_{yf_\ell}(z)$ is the transfer matrix between f_ℓ and y^9. A necessary condition for (6.208) to hold is $n_f \leq p$, where p is the number of measured output signals (dimension of $\boldsymbol{y}$).

When it is not possible to design residual generators in such a way that each residual is only sensitive to a single fault, it is still possible to achieve fault isolation provided the zero entries in the table characterising the effect of the faults on the residual have a different pattern in each column. However, only a diagonal structure such as in Table 6.4 allows isolation of multiple simultaneous faults.

Remark 6.21 *Accounting for correlation between residual vectors*
Since the different residual vectors are built on the basis of the same stochastic model, they are generally correlated. Hence the residual evaluation should ideally be carried out on the stacked residual vector

$$\boldsymbol{r}(k) = \begin{pmatrix} \boldsymbol{r}_1(k)' & \boldsymbol{r}_2(k)' & \ldots & \boldsymbol{r}_{n_f}(k)' \end{pmatrix}'.$$

The problem can then be written in the form of a multiple hypotheses testing. The interested reader is referred to [179], [114] for the algorithms to be used. □

Example 6.15 (cont.) *Ship example*

Faults on both the rate sensor and the angular position sensor are now considered. Figure 6.25 depicts the measurement signals obtained when step-like faults with magnitude 0.1 deg/s and 0.5 deg are respectively introduced on ω_{3m} between time instant 301 and 600 and on ψ_m between time instant 900 and 1200. All time data are expressed in number of sampling periods.

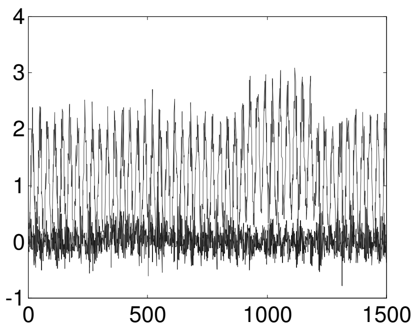


Fig. 6.25. Angular rate and heading measurements

In order to achieve fault isolation, two residual signals are generated, one being sensitive to f_ω, the other to f_ψ. To this end, consider the sampled-data ship model (6.194), (6.195). If a Kalman filter is designed for this system using only the first measurement equation in (6.195), the resulting residual will only be affected by f_ω. Such a filter cannot be designed because the resulting system is not detectable. However, there is no need to estimate the whole state

9 rank $\boldsymbol{H}(z)$ stands for the normal rank of matrix $\boldsymbol{H}(z)$; it can be computed as $\max_z$ rank $\boldsymbol{H}(z)$

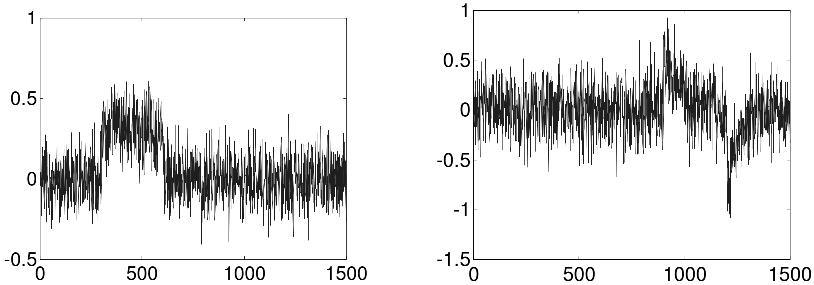
to generate a residual; it suffices to design a Kalman filter for the first 3 state equations in (6.194) and the first measurement equation. This filter takes the form:

$$\begin{pmatrix} \hat{x}_{w1}(k+1) \\ \hat{x}_{w2}(k+1) \\ \hat{\omega}_3(k+1) \end{pmatrix} = \begin{pmatrix} -0.1281 & -0.6365 & 0 \\ 0.9945 & 0.1106 & 0 \\ 0 & 0 & 0.0000 \end{pmatrix} \begin{pmatrix} \hat{x}_{w1}(k) \\ \hat{x}_{w2}(k) \\ \hat{\omega}_3(k) \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ 0.0500 \end{pmatrix} \delta(k) + \begin{pmatrix} -0.6760 \\ 0.8104 \\ 0.0000 \end{pmatrix} \cdot \left(\omega_{3m}(k) - (1\ \ 0\ \ 1) \begin{pmatrix} \hat{x}_{w1}(k) \\ \hat{x}_{w2}(k) \\ \hat{\omega}_3(k) \end{pmatrix} \right) \quad (6.209)$$

The innovation is computed from

$$r_{\omega_3}(k) = \omega_{3m}(k) - (1\ \ 0\ \ 1) \begin{pmatrix} \hat{x}_{w1}(k) \\ \hat{x}_{w2}(k) \\ \hat{\omega}_3(k) \end{pmatrix} \quad (6.210)$$

It is plotted in Fig. 6.26 (a). A significant change in the mean of this signal is visible when the fault on ω_3 is present.



(a) (b)

Fig. 6.26. Residual affected by f_ω (a) or f_ψ (b) only

The design of a residual generator for detection and isolation of f_ψ is based on the model made of Eqs. (6.194) and (6.195). This system is detectable and the innovation, r_ψ, of the Kalman filter based on the above model is affected by fault f_ψ as can be seen in Fig. 6.26 (b). However, the latter fault is not strongly detectable.

Hence for evaluation of residual r_ψ, one has to resort to the GLR algorithm, since it is not possible to detect fault disappearance by successive reinitilisation of a CUSUM algorithm. The latter option is possible for evaluation of r_{ω_3} however. Figure 6.27 represent the GLR decision function obtained by processing the residual sequence of Fig. 6.26 (a) and the CUSUM decision function obtained by processing the residual of Fig. 6.26 (b). Repeated alarms are issued by the CUSUM algorithm, the first occurring at time 300, the last one at time 597. In this time interval, the CUSUM decision function crosses its threshold every 5 samples on the average. Appearance and disappearance of the fault on the angular rate measurement can thus be detected and isolated. As far as the GLR decision function of Fig. 6.27 (a) is concerned, it reaches its threshold at time 904, and the estimated fault occurrence time of f_ψ is

900 (actual value 901). The estimated fault magnitude based on the residual in the time window [900 949] is 0.449, which is in error by 10 %. After reinitialisation, the GLR algorithm detects fault disappearance at time 1203 and it provides instant 1201 as the estimate of the change occurrence time, namely the correct time instant. The estimated change magnitude is -0.618 which is in error by 23 %.

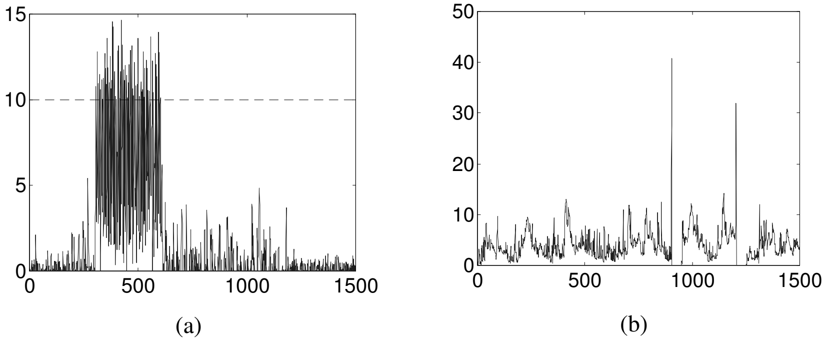


Fig. 6.27. CUSUM decision function and GLR decision function resulting from evaluation of r_ψ (a) and r_{ω_3} (b)

6.9 Exercises

Exercise 6.1 *Residual generator for position actuator*
Consider the system in Fig. 3.8 and parameters given in Exercise 3.3. There is no measurement noise in the exercise.

1. Implement a candidate residual generator. Use the parity equations
$$e(s) = y_m(s) - \hat{y}(s),$$
where
$$\hat{y}_1(s) = \frac{1}{sI_{tot} + \alpha}\left(k_q \eta\, i_m(s)\right),$$
and
$$\hat{y}_2(s) = \frac{1}{Ns}\left(n_m(s)\right).$$
Investigate the properties of these potential residual generators by applying step changes on either of the faults.
2. Consider further the possible fault in the shaft speed sensor. Investigate experimentally whether all three faults can be detected and isolated.
3. Derive the transfer function matrix $\boldsymbol{H}_{yf}(s)$ and use this to explain the observations. □

Exercise 6.2 *Residual generation using the parity space approach*
This exercise deals with residual generator for the industrial actuator. Refer to Fig. 3.7. The disturbance is Q_l. The input is i_{com}. The measurements are n_m and θ_m.

1. Determine the transfer function matrices $\boldsymbol{H}_{yu}(s)$ and $\boldsymbol{H}_{yd}(s)$.
2. Write the transfer function matrix
$$\boldsymbol{H}(s) = \begin{pmatrix} \boldsymbol{H}_{yu}(s) & \boldsymbol{H}_{yd}(s) \\ \boldsymbol{I} & \boldsymbol{O} \end{pmatrix}.$$
3. Write $\boldsymbol{H}(s)$ in the form
$$\boldsymbol{H}(s) = \frac{1}{h(s)}\tilde{\boldsymbol{H}}(s)$$
where $\tilde{H}$ is a polynomial matrix.
4. Determine the rank of $\tilde{\boldsymbol{H}}(s)$.
5. Determine the nullspace of $\tilde{\boldsymbol{H}}'(s)$.
6. From the nullspace of $\tilde{\boldsymbol{H}}'(s)$, determine residual generator(s)
$$\boldsymbol{r}(s) = \boldsymbol{V}_{ru}(s)\boldsymbol{u}(s) + \boldsymbol{V}_{ry}(s)\boldsymbol{y}(s)$$
that make the residual independent of unknown input. Verify this property by showing that
$$\boldsymbol{V}_{ry}(s)\boldsymbol{H}_{yd} = 0.$$
7. Determine the relation
$$\boldsymbol{r}(s) = \boldsymbol{V}_{ry}(s)\boldsymbol{H}_{yf}(s) = \frac{1}{h(s)}\tilde{\boldsymbol{H}}(s)$$
and determine which of the three faults f_i, f_n and f_θ are detectable. □

Exercise 6.3 *Residual generation for single-axis satellite*
In continuation of Exercise 5.3 this exercise deals with residual generation for the single axis satellite.

A state-space model for the single axis is given by

$$\begin{aligned} \dot{x}_1 &= \tfrac{1}{I}(ku_1 + ku_2 + w_0) \\ \dot{x}_2 &= x_1 \\ y_1 &= x_1 + f_1 \\ y_2 &= x_2 + f_2 \\ y_3 &= x_2 + f_3 \\ y_4 &= k_1 u_1 + f_4 \\ y_5 &= k_2 u_2 + f_5, \end{aligned}$$

where x_1 is the angular velocity, x_2 the angle of the satellite and nominal parameters are

$$\begin{aligned} I &= 14.33\,\text{kgm}^2 \\ k_1 &= k_2 = 0.5\,. \end{aligned}$$

There are two input signals, u_1 and u_2 to actuators 1 and 2, respectively. There is one unknown input d. The magnitude of d is not known prior to the launch of the satellite, but it is known that d is constant over time.

There are five measurements: y_1 measures the state x_1, y_2 and y_3 measure the state x_2. y_4 measures the actual torque from actuator 1, y_5 measures the actual torque from actuator 2.

1. Determine the transfer function matrices $\boldsymbol{H}_{yu}(s)$ and $\boldsymbol{H}_{yd}(s)$.
2. Determine the transfer function matrix
$$\boldsymbol{H}(s) = \begin{pmatrix} \boldsymbol{H}_{yu}(s) & \boldsymbol{H}_{yd}(s) \\ \boldsymbol{I} & \boldsymbol{O} \end{pmatrix}.$$
3. Write $\boldsymbol{H}(s)$ in the form
$$\boldsymbol{H}(s) = \frac{1}{h(s)}\tilde{\boldsymbol{H}}(s)$$
where $\tilde{\boldsymbol{H}}$ is a polynomial matrix.
4. Determine the rank of $\tilde{\boldsymbol{H}}(s)$.
5. How many independent residual generators can be expected that are independent of input $\boldsymbol{u}(s)$ and of disturbances $d(s)$.
6. Find the left nullspace of $\tilde{\boldsymbol{H}}(s)$.
7. Determine a residual generator based on the nullspace. □

Exercise 6.4 *Properties of residual generators for single-axis satellite*
This exercise is a continuation of 6.3.

1. Determine the response of the residual vector to the additive faults on y_1 to y_5 by calculating
$$\boldsymbol{r}(s) = \boldsymbol{V}_{ry}(s)\boldsymbol{H}_{yf}(s)\boldsymbol{f}(s).$$
2. Determine which of the above faults are detectable and which are strongly detectable.
3. Determine which of above faults can be isolated.
As pure differentiation or integration are not feasible in the presence of measurement noise, a filter is applied on one of the residuals. Investigate the features of two proposed residual generators. Both have the form
$$\begin{aligned} r_{12}(s) &= \frac{1}{s+\alpha}y_1(s) - \frac{s}{s+\alpha}y_2(s) \\ r_{23}(s) &= y_2(s) - y_3(s). \end{aligned}$$
Version a has $\alpha = 0.01$, version b has $\alpha = 10$.
4. Discuss the properties of the two residual generators (detectability, strong detectability, isolability). Apply a fixed threshold on either set of generators to detect if a fault is present and verify your results by simulation. □

Exercise 6.5 *Residual generator design - optimisation method*
This exercise addresses the position servo from Exercise 3.2, Fig. 5.35. The exercise is to design residual generators based on the standard setup used in robust control. It is assumed that only a single fault can appear at a time.

1. Formulate the FDI problem for the system as a standard problem. Identify the matrices that need to be selected in connection with the design.
2. Design residual generators for fault detection using the standard setup and standard design methods.

3. Design a residual generator for fault isolation and fault estimation using the standard setup and standard design methods. □

Exercise 6.6 *Residual generator with an explicit specification*

This exercise addresses the position servo from Exercise 3.2, Fig. 5.36.

Assume the load possess a dominant disturbance above 0.5 rad/s.

1. Formulate a specification $\boldsymbol{H}_{zd}(s)$ and $\boldsymbol{H}_{zf}(s)$ for the design.
2. Formulate the fault detection and isolation problem for the system as a standard problem. Identify the matrices that need to be selected in connection with the design.
3. Design residual generators for fault detection sing the standard setup and standard design methods.
4. Design a residual generator for fault isolation and fault estimation using the standard setup and standard design methods. □

Exercise 6.7 *Covariance of LP filter output - input is band limited noise*

Given a low-pass filter with the state-space representation

$$\begin{aligned} \dot{x}(t) &= -\alpha x(t) + \alpha w(t) \\ y(t) &= x(t) \end{aligned}$$

with input $w(t)$, a band-limited random signal generated by

$$dw(t) = -\beta w(t)dt + \sigma_w^2 \sqrt{2\beta}\, dv(t).$$

1. Represent the filter in the form
$$d\boldsymbol{x}(t) = \boldsymbol{A}x(t) + \boldsymbol{B}d\boldsymbol{v}(t).$$
2. Let the covariance matrix be
$$\boldsymbol{Q} = E\left\{ \begin{pmatrix} w(t) \\ x(t) \end{pmatrix} \begin{pmatrix} w(t) & x(t) \end{pmatrix} \right\} = \begin{pmatrix} a & c \\ c & b \end{pmatrix}.$$
Calculate the covariance $\boldsymbol{Q}$ as the solution to the Lyapunov equation
$$\boldsymbol{AQ} + \boldsymbol{QA}' + \boldsymbol{BS}_v\boldsymbol{B}' = 0.$$
3. Show that the variance on y , σ_y^2, is
$$\sigma_y^2 = \frac{\alpha}{\alpha + \beta}\sigma_w^2 \tag{6.211}$$
and determine the value of the pole α required to obtain a desired value of σ_y^2 given σ_w^2. □

Exercise 6.8 *Change detection for industrial actuator*
Given a simulation model and residual generators based on parity equations, this exercise deals with detection of faults in the presence of measurement noise and random disturbances.

Let a noise specification for n_m, θ_m and i_m be given by the autocorrelation function

$$R_{ii}(\tau) = \sigma_i^2 e^{-\beta_i |\tau|}$$

where

$$\begin{array}{lll} n_m: & \sigma_n = 3 \text{ rad/s} & \beta = 10 \\ \theta_m: & \sigma_\theta = 0.01 \text{ rad} & \beta = 2 \\ i_m: & \sigma_i = 0.2 \text{ A} & \beta = 10 \end{array} .$$

The noise sources are not correlated.

1. Implement a simple threshold (level) detector on the two residuals from the parity equations. Investigate whether you can detect
   a) a step change of 0.015 rad in the position sensor
   b) a step change of 0.15 A in the power drive current.
2. Design a scalar CUSUM detector of change in mean value. Test for the hypothesis that a fault is present an reflected in the a mean value change of the values given above.
3. Design a detector for the position sensor fault that has a time to detect of 2.5 s. Determine the average time between false alarms. Increase the specified time to detect to 10 s and determine the new average time between false alarms.
4. Investigate experimentally (by simulation) whether the two faults can be detected.
5. Verify the time to detect and the false alarm rates using different seeds of your measurement noise generators.
   Note 1: The results on time to detect and mean time between false alarms assume white noise statistics of the log-likelihood test quantity $s(k)$. When $s(k)$ in not white, the statistical results are only approximate figures that can only be used as guidelines for design. □

Exercise 6.9 *GLR change detection design*
As a continuation of Exercise 6.7, design a GLR estimator.

Design a scalar GLR based detector. Investigate the limits of faults that can be detected and find the threshold on the decision function that gives a time to detect of 2.5 s for a fault of similar magnitude as in Exercise 6.7. □

Exercise 6.10 *Change detection for single-axis satellite*
Referring to 6.3, measurements are subject to measurement noise. A state-space model for the single axis is given by:

$$\begin{aligned} \dot{x}_1 &= \frac{1}{I}(ku_1 + ku_2 + d) \\ \dot{x}_2 &= x_1 \\ y_1 &= x_1 + f_1 + w_1 \\ y_2 &= x_2 + f_2 + w_2 \\ y_3 &= x_2 + f_3 + w_3 \\ y_4 &= k_1 u_1 + f_4 \\ y_5 &= k_2 u_2 + f_5, \end{aligned}$$

where d is an unknown disturbance. The noise specification for w_1, w_2 and w_3 are given by the autocorrelation function

$$R_{ii}(\tau) = \sigma_i^2 e^{-\beta_i|\tau|}$$

where

$$\begin{array}{lll} w_1: & \sigma_1 = 2 \cdot 10^{-4}\ \text{rad/s} & \beta = 10 \\ w_2: & \sigma_2 = 1 \cdot 10^{-5}\ \text{rad} & \beta = 10 \\ w_3: & \sigma_3 = 2 \cdot 10^{-3}\ \text{rad} & \beta = 10. \end{array}$$

The three noise sources are uncorrelated.

Consider two residuals that are supposed used to detect a fault f_2 in the measurement y_2.

$$\begin{aligned} r_{12}(k) &= T\sum_{i=1}^{k} y_1(i) + y_2(0) - y_2(k) \\ r_{23}(k) &= y_2(k) - y_3(k) \end{aligned} \quad (6.212)$$

where T is the sampling time of the measurements. Assume the sampling time is $T = 1$ s.

1. Calculate the variance of r_{12} and r_{23} above.
   It is desired to design a change detector such that faults larger than $2 \cdot 10^{-3}$ rad on y_2 are detected after max. 10 s (10 samples). This is not possible due to the large variance of the noise w_3 on y_3.
2. Determine which variance y_3 should have in order to meet the average time to detect as specified. Design a low-pass filter on r_{23} that will reduce the variance as required. Note that the ARL function is derived on the assumption of a white residual. As you are violating this assumption, validation by simulation is required at a later stage.
3. Design a set of scalar-based change detection algorithms for the case the fault on y_2 has a magnitude of $2 \cdot 10^{-3}$ rad and the change is a step. Verify that the desired time to detect can be obtained. □

Exercise 6.11 *Vector-based change detection for single-axis satellite*

1. Determine the fault signature in the residual vector assuming a fault on y_2 appears as a step.
2. Design a vector-based change detection algorithm for the case the fault on y_2 has a magnitude of $2 \cdot 10^{-3}$ rad and appears as a step. Discuss the properties of the vector-based change detection compared with the set of scalar algorithms. □

Exercise 6.12 *Residual generation in a Luenberger observer*

Consider the following linear time invariant system

$$\begin{aligned} \begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \end{pmatrix} &= \begin{pmatrix} 0 & 1 \\ -2 & -3 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + \begin{pmatrix} f_1 \\ f_2 \end{pmatrix} \\ y &= \begin{pmatrix} 1 & 0 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + f_3 \end{aligned}$$

where $\boldsymbol{x} = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$ is the state, $\boldsymbol{f} = \begin{pmatrix} f_1 \\ f_2 \\ f_3 \end{pmatrix}$ is the fault vector ($f = 0 \iff$ normal operation) and y is the measured output.

1. Define the parameters k_1 and k_2 of a Luenberger observer
$$\begin{aligned} \begin{pmatrix} \dot{z}_1 \\ \dot{z}_2 \end{pmatrix} &= \begin{pmatrix} 0 & 1 \\ -2 & -3 \end{pmatrix} \begin{pmatrix} z_1 \\ z_2 \end{pmatrix} + \begin{pmatrix} k_1 \\ k_2 \end{pmatrix} (y - \hat{y}) \\ \hat{y} &= \begin{pmatrix} 1 & 0 \end{pmatrix} \begin{pmatrix} z_1 \\ z_2 \end{pmatrix} \end{aligned}$$
which has the following property: in the absence of faults, the estimation error
$$\begin{pmatrix} z_1 - x_1 \\ z_2 - x_2 \end{pmatrix}$$
converges to zero with a dynamics associated with the two eigenvalues $\lambda_1 = \lambda_2 = -5$.
2. Determine the transfer function between the residual $r = y - \hat{y}$ and the fault vector $\boldsymbol{f}$ under the form
$$r = G_1(s)f_1 + G_2(s)f_2 + G_3(s)f_3. \; \square \tag{6.213}$$

Exercise 6.13 *Static and dynamic redundancy*

Consider the following system composed of 4 components: process, sensor 1, sensor 2, sensor 3.

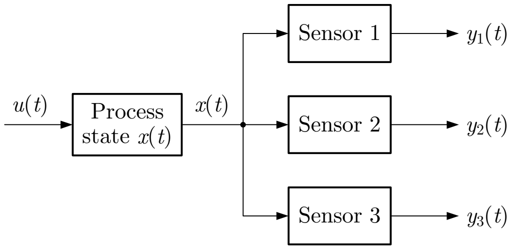


Fig. 6.28. System with three sensors

It is assumed that it can be described by the following linear time invariant model
$$\begin{aligned} \begin{pmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \end{pmatrix} &= \begin{pmatrix} 0 & 1 \\ -2 & -3 \end{pmatrix} \begin{pmatrix} x_1(t) \\ x_2(t) \end{pmatrix} + \begin{pmatrix} 0 \\ 1 \end{pmatrix} u(t) \\ &+ \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} f_1(t) \\ f_2(t) \\ f_3(t) \\ f_4(t) \end{pmatrix} \end{aligned}$$

$$\begin{pmatrix} y_1(t) \\ y_2(t) \\ y_3(t) \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 1 & 1 \\ 2 & 1 \end{pmatrix} \begin{pmatrix} x_1(t) \\ x_2(t) \end{pmatrix} + \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} f_1(t) \\ f_2(t) \\ f_3(t) \\ f_4(t) \end{pmatrix}$$

where

$$\boldsymbol{x}(t) = \begin{pmatrix} x_1(t) \\ x_2(t) \end{pmatrix}$$

characterises the state of the process component, $u(t)$ is the control input,

$$\boldsymbol{y}(t) = \begin{pmatrix} y_1(t) \\ y_2(t) \\ y_3(t) \end{pmatrix}$$

is the vector of all measurements and

$$\boldsymbol{f}(t) = \begin{pmatrix} f_1(t) \\ f_2(t) \\ f_3(t) \\ f_4(t) \end{pmatrix}$$

is the fault vector.

1. What is the association between the faults f_i, $i = 1, 2, 3, 4$ and the system components.
2. Is the state $\boldsymbol{x}(t)$ observable?
3. Is there any static redundancy in this system? What are the detectable / isolable faults?
4. Assume that during a given period of time, only sensor y_1 is operational (for example, y_2 and y_3 are disconnected for maintenance). Is it still possible to estimate the state ? to detect and isolate the faults? □

6.10 Bibliographical notes

The parity relations that were initially studied in [44],[79] and [135] are functions of a sliding window of the most recent sensor output and actuator input values. The idea used to develop parity relations in the time domain was extended to the frequency domain. This has lead to the so-called generalised parity relations [260] which do not necessarily involve only the data of a sliding window. Later this distinction between parity relations and generalised parity relations tended to disappear. The presentation given here is in the line of [187], [188]. A way to assure causality and stability of a filter involving the inverse of a transfer matrix can be found in [113]. A thorough study of the parity space approach to residual generation can also be found in [78]. The equivalence between observer-based and parity space approaches is developed in [166] for instance. Further results on the design of residual generators in the frequency domain can be found in [113].

The systematic computation of analytical redundancy relations for polynomial nonlinear models is developed in [238] and [278]. Details on elimination theory may be found in [49] and [218]. Gröbner bases used in Buchberger's algorithm [27], details and definitions can be found in [46]. The reader is referred to [81] for details as the use of characteristic sets.

A comprehensive reference to fault diagnosis treated as an optimisation problem is [168]. Earlier research results, that relate to the presentation in this book, were published in [56] and [57]. The book [43] has a chapter devoted to this subject. The design of fault diagnosis filters using the standard setup presented in this book originates in [184] and [242].

More information on threshold detection can be found in the classical presentation of this subject of [54] and, for later results, in [48] and [104].

The observer-based approach for residual generation has been the object of numerous studies, cf. to the book [194] for an introductory treatment and references on early works in this area. [43] provides more recent developments on the topic as well as a very complete list of references.

The non-sequential and sequential algorithms for change detection in signals are described in details in the book [5]. The result of Nieman and Pearson, on the optimality of tests based on the likelihood ratio between two hypotheses can be found in Section 4.2.2 of [5]. The properties of sequential algorithms deduced from the ARL function are investigated in Chapter 4 of [5]. A heuristic approach for choosing the design parameters of the GLR algorithm for detection of changes in the mean is presented in [198]. An alternative to the GLR algorithm, which is less time-consuming, is presented in [180].

A numerically stable algorithm to extract the observable part of a given system can be found on page 220 of [42]; it can be used as a first step to design a residual generator based on a Kalman filter for an unobservable system. The design of a residual generator based on a Kalman filter for a system subject to unknown input and additive faults is adapted from [181]. An alternative approach to compute an innovation sequence is to use parity relations and to filter the obtained residual by an appropriate whitening filter. This method has been considered in [198] for instance. It was not possible to examine here the question of robustness with respect to modelling uncertainties of the statistical approach to fault detection. A valuable reference to study this question is [168].

System identification based methods for fault detection, estimation and isolation have not been considered in this chapter but they have also proved useful in many applications. They can be separated in two classes: methods based on explicit parameter estimation and methods based on statistics (such as the statistical local approach). An introduction to the first class of methods can be found in [97] and [99]. For the second class, the reader is referred to [277] and [5].

Active fault detection and isolation has been briefly mentioned in this chapter. The problem of determining an optimal input signal to distinguish between different models (representing healthy and faulty modes) for a given process has been the object of a thorough study in [34] and [175]. [177] suggested novel ways to achieve active fault isolation while a plant is running.

For more information on the material in the appendix on random variables and stochastic processes, the reader can consult [4], [109] or [192] for instance. In particular the approach for sampling a linear stochastic differential equation is borrowed from pages 147-151 of [4].