

Chapter 5

Synthesis of Fault Detection and Isolation Filters

This chapter presents general synthesis procedures of fault detection filters which solve the fault detection problems formulated in Chap. 3. The synthesis procedures are described in terms of input–output models, which allow simpler conceptual presentations. Numerically reliable state-space representation based synthesis algorithms, well-suited for robust software implementations, are described in Chap. 7.

In the recently developed computational procedures for the synthesis of fault detection filters, two important computational paradigms emerged, which are instrumental in developing generally applicable, numerically reliable and computationally efficient synthesis methods. The first paradigm is the use of factorization-based synthesis methods. Accordingly, for all presented synthesis procedures, it is possible to express the TFM of the final filter $Q(\lambda)$ in a factored form as

$$Q(\lambda) = Q_K(\lambda) \cdots Q_2(\lambda)Q_1(\lambda), \quad (5.1)$$

where $Q_1(\lambda)$, $Q_2(\lambda)Q_1(\lambda)$, \dots , can be interpreted as partial syntheses addressing specific requirements. Since each partial synthesis may represent a valid fault detection filter, this approach has a high flexibility in using or combining different synthesis techniques. The factorization-based synthesis approach naturally leads to the so-called integrated computational algorithms, with strongly coupled successive computational steps. For a K -step synthesis procedure to determine $Q(\lambda)$ in the factored form (5.1), K updating operations of the form $Q(\lambda) \leftarrow Q_i(\lambda)Q(\lambda)$ are performed for $i = 1, \dots, K$, where $Q_i(\lambda)$ is the factor computed at the i -th synthesis step. The state-space description based filter updating formulas are described in Chap. 7 for specific synthesis steps.

The second paradigm is the use of the nullspace method as a first synthesis step to reduce all synthesis problems to simpler problems, which allow to easily check the solvability conditions and address least-order synthesis problems. The nullspace-based synthesis approach is described in Sect. 5.1. In Sects. 5.2–5.7 specific synthesis

procedures, relying on the nullspace method, are presented for each of the fault detection and isolation problems formulated in Chap. 3.

5.1 Nullspace-Based Synthesis

A useful parametrization of all fault detection filters can be obtained on the basis of conditions $R_u(\lambda) = 0$ and $R_d(\lambda) = 0$ in (3.23). For any fault detection filter $Q(\lambda)$ the condition $[R_u(\lambda) R_d(\lambda)] = 0$ is equivalent to

$$Q(\lambda) \begin{bmatrix} G_u(\lambda) & G_d(\lambda) \\ I_{m_u} & 0 \end{bmatrix} = 0.$$

Thus, any fault detection filter $Q(\lambda)$ must be a left annihilator of the TFM

$$G(\lambda) := \begin{bmatrix} G_u(\lambda) & G_d(\lambda) \\ I_{m_u} & 0 \end{bmatrix}. \quad (5.2)$$

Let r_d be the normal rank of $G_d(\lambda)$ (i.e., maximal rank over all λ). Using standard linear algebra results (see Sect. 9.1.3), there exists a maximal full row rank left annihilator $N_l(\lambda)$ of size $(p - r_d) \times (p + m_u)$ such that $N_l(\lambda)G(\lambda) = 0$. Any such an $N_l(\lambda)$ represents a *basis* of $\mathcal{N}_L(G(\lambda))$, the left nullspace of $G(\lambda)$. Using this fact, we have the following straightforward parametrization of all fault detection filters:

Theorem 5.1 *Let $N_l(\lambda)$ be a basis of $\mathcal{N}_L(G(\lambda))$, with $G(\lambda)$ defined in (5.2). Then, any fault detection filter $Q(\lambda)$ satisfying (3.23) can be expressed in the form*

$$Q(\lambda) = V(\lambda)N_l(\lambda), \quad (5.3)$$

where $V(\lambda)$ is a suitable TFM.

Proof Let $q^{(i)}(\lambda)$ be the i -th row of $Q(\lambda)$. Since $q^{(i)}(\lambda)G(\lambda) = 0$, it follows that $q^{(i)}(\lambda) \in \mathcal{N}_L(G(\lambda))$ and therefore there exists a vector $v^{(i)}(\lambda)$ such that $q^{(i)}(\lambda) = v^{(i)}(\lambda)N_l(\lambda)$, representing a linear combination of the nullspace basis vectors. Thus, we build $V(\lambda)$ in (5.3) as a TFM whose i -th row is $v^{(i)}(\lambda)$. ■

Remark 5.1 For any non-singular polynomial or rational matrix $M(\lambda)$ of appropriate dimension, $\tilde{N}_l(\lambda) := M(\lambda)N_l(\lambda)$ is also a nullspace basis. Frequently, $M(\lambda)$ is the denominator matrix of a *left coprime factorization* (LCF) of an original basis $N_l(\lambda)$ in the form

$$N_l(\lambda) = M(\lambda)^{-1}\tilde{N}_l(\lambda), \quad (5.4)$$

where the factors $M(\lambda)$ and $\tilde{N}_l(\lambda)$ are determined to satisfy special requirements, such as properness, or to have only poles in a certain “good” region of the complex plane (e.g., in the stability region), or both. In this case, if $N_l(\lambda)$ is a basis, then

$\tilde{N}_l(\lambda) = M(\lambda)N_l(\lambda)$ is a basis as well. Moreover, $M(\lambda)$ has as zeros all poles of $N_l(\lambda)$ lying outside of the “good” region. For more details on coprime factorizations see Sect. 9.1.6. \square

An interesting property of nullspace bases is the following elementary fact. Consider a column partitioning of $G(\lambda)$ as $G(\lambda) = \begin{bmatrix} G_1(\lambda) & G_2(\lambda) \end{bmatrix}$ and let $N_{l,1}(\lambda)$ be a basis of $\mathcal{N}_L(G_1(\lambda))$ and $N_{l,2}(\lambda)$ be a basis of $\mathcal{N}_L(N_{l,1}(\lambda)G_2(\lambda))$. Then, $N_{l,2}(\lambda)N_{l,1}(\lambda)$ is a basis of $\mathcal{N}_L(G(\lambda))$. Using this fact with the following partitioning

$$G(\lambda) = \begin{bmatrix} G_1(\lambda) & | & G_2(\lambda) \end{bmatrix} := \begin{bmatrix} G_u(\lambda) & | & G_d(\lambda) \\ I_{m_u} & | & 0 \end{bmatrix},$$

we immediately obtain the left nullspace basis $N_l(\lambda)$ in the factorized form

$$N_l(\lambda) = N_{l,d}(\lambda) \begin{bmatrix} I_p & -G_u(\lambda) \end{bmatrix}, \quad (5.5)$$

where $N_{l,d}(\lambda)$ is a $(p - r_d) \times p$ TFM representing a basis of $\mathcal{N}_L(G_d(\lambda))$. This form leads to simple expressions of $N_l(\lambda)$ for particular cases as $N_l(\lambda) = N_{l,d}(\lambda)$ if $m_u = 0$, or $N_l(\lambda) = \begin{bmatrix} I_p & -G_u(\lambda) \end{bmatrix}$ if $m_d = 0$, or $N_l(\lambda) = I_p$ if $m_u + m_d = 0$.

A proper and stable representation of $N_l(\lambda)$ for arbitrary rational or polynomial matrices $G_u(\lambda)$, $G_d(\lambda)$, $G_w(\lambda)$ and $G_f(\lambda)$ can be obtained from the LCF

$$\begin{bmatrix} G_u(\lambda) & G_d(\lambda) & G_w(\lambda) & G_f(\lambda) \end{bmatrix} = \widehat{M}^{-1}(\lambda) \begin{bmatrix} \widehat{G}_u(\lambda) & \widehat{G}_d(\lambda) & \widehat{G}_w(\lambda) & \widehat{G}_f(\lambda) \end{bmatrix}, \quad (5.6)$$

where $\widehat{M}(\lambda)$ and $\begin{bmatrix} \widehat{G}_u(\lambda) & \widehat{G}_d(\lambda) & \widehat{G}_w(\lambda) & \widehat{G}_f(\lambda) \end{bmatrix}$ are proper and stable factors. With obvious replacements, the left nullspace basis $N_l(\lambda)$ can be chosen as

$$N_l(\lambda) = \widehat{N}_{l,d}(\lambda) \begin{bmatrix} \widehat{M}(\lambda) & -\widehat{G}_u(\lambda) \end{bmatrix}, \quad (5.7)$$

where $\widehat{N}_{l,d}(\lambda)$ is a $(p - r_d) \times p$ proper and stable TFM representing a basis of $\mathcal{N}_L(\widehat{G}_d(\lambda))$. If $m_u = m_d = 0$, then we can formally set $N_l(\lambda) := \widehat{M}(\lambda)$.

For the particular form of the nullspace basis in (5.7), we have the following straightforward corollary of Theorem 5.1:

Corollary 5.1 *Let $\widehat{G}_d(\lambda)$ and $\widehat{G}_u(\lambda)$ be the TFMs defined in (5.6) and let $\widehat{N}_{l,d}(\lambda)$ be a basis of $\mathcal{N}_L(\widehat{G}_d(\lambda))$. Then, any fault detection filter $Q(\lambda)$ satisfying (3.23) can be expressed in the form*

$$Q(\lambda) = W(\lambda)\widehat{N}_{l,d}(\lambda) \begin{bmatrix} \widehat{M}(\lambda) & -\widehat{G}_u(\lambda) \end{bmatrix}, \quad (5.8)$$

where $W(\lambda)$ is a suitable TFM.

The parametrization result of Theorem 5.1 underlies the nullspace method based synthesis procedures of fault detection filters, which form the main focus of this book. All synthesis procedures of the fault detection filters rely on the initial factored form

$$Q(\lambda) = \overline{Q}_1(\lambda)Q_1(\lambda), \quad (5.9)$$

where $Q_1(\lambda) = N_l(\lambda)$ is a basis of $\mathcal{N}_L(G(\lambda))$, while $\overline{Q}_1(\lambda)$ is a factor to be subsequently determined. The nullspace-based first step allows to reduce all synthesis problems formulated for the system (3.2) to simpler problems, which make straightforward to check the solvability conditions.

Using the factored form (5.9), the fault detection filter in (3.3) can be rewritten in the alternative form

$$\mathbf{r}(\lambda) = \overline{Q}_1(\lambda)Q_1(\lambda) \begin{bmatrix} \mathbf{y}(\lambda) \\ \mathbf{u}(\lambda) \end{bmatrix} = \overline{Q}_1(\lambda)\overline{\mathbf{y}}(\lambda), \quad (5.10)$$

where

$$\overline{\mathbf{y}}(\lambda) := Q_1(\lambda) \begin{bmatrix} \mathbf{y}(\lambda) \\ \mathbf{u}(\lambda) \end{bmatrix} = \overline{G}_f(\lambda)\mathbf{f}(\lambda) + \overline{G}_w(\lambda)\mathbf{w}(\lambda), \quad (5.11)$$

with

$$[\overline{G}_f(\lambda) \ \overline{G}_w(\lambda)] := Q_1(\lambda) \begin{bmatrix} G_f(\lambda) & G_w(\lambda) \\ 0 & 0 \end{bmatrix}. \quad (5.12)$$

With this first preprocessing step, we reduced the original problems formulated for the system (3.2) to simpler ones, which can be formulated for the reduced system (5.11) (without control and disturbance inputs), for which we have to determine the TFM $\overline{Q}_1(\lambda)$ of the simpler fault detection filter (5.10).

Remark 5.2 At this stage, we can assume that both $Q_1(\lambda)$ and the TFMs of the reduced system (5.11) are proper and even stable. This can be always achieved by replacing any basis $N_l(\lambda)$, with a stable basis $Q_1(\lambda) = M(\lambda)N_l(\lambda)$, where $M(\lambda)$ is an invertible, stable and proper TFM, of least McMillan degree, such that $M(\lambda)[N_l(\lambda) \ \overline{G}_f(\lambda) \ \overline{G}_w(\lambda)]$ is stable and proper. Such an $M(\lambda)$ can be determined as the minimum-degree denominator of a stable and proper LCF of $[N_l(\lambda) \ \overline{G}_f(\lambda) \ \overline{G}_w(\lambda)]$ (see Sect. 9.1.6). Even if $N_l(\lambda)$ is a minimal basis, the resulting stable basis $Q_1(\lambda)$ is, in general, not a minimal basis. \square

We conclude this section with the derivation of simpler conditions for checking the fault detectability conditions studied in the Sect. 3.3. The following result characterizes the complete fault detectability of the system (3.2) as the complete fault input observability property of the reduced system (5.11).

Proposition 5.1 *For the system (3.2) with $w \equiv 0$, let $Q_1(\lambda) = N_l(\lambda)$ be a rational basis of $\mathcal{N}_L(G(\lambda))$, where $G(\lambda)$ is defined in (5.2), and let (5.11) be the corresponding reduced system with $w \equiv 0$. Then, the system (3.2) is completely fault detectable if and only if*

$$\overline{G}_{f_j}(\lambda) \neq 0, \quad j = 1, \dots, m_f. \quad (5.13)$$

Proof To prove necessity we show that if the original system is completely fault detectable, then the reduced system (5.11) is also completely fault detectable (i.e.,

conditions (5.13) are fulfilled). For the completely fault detectable system (3.2), let $Q(\lambda)$ be a filter such that $R_{f_j}(\lambda) \neq 0$ for $j = 1, \dots, m_f$. According to Theorem 5.1, for a given nullspace basis $N_l(\lambda)$, any filter $Q(\lambda)$ can be expressed in the form $Q(\lambda) = W(\lambda)N_l(\lambda)$, where $W(\lambda)$ is a suitable rational matrix. It follows that $R_{f_j}(\lambda) = W(\lambda)\overline{G}_{f_j}(\lambda)$ and therefore, $R_{f_j}(\lambda) \neq 0$ only if $\overline{G}_{f_j}(\lambda) \neq 0$.

The proof of sufficiency is trivial, since with $Q(\lambda) := N_l(\lambda)$ the corresponding $R_f(\lambda) = \overline{G}_f(\lambda)$, and thus satisfies $R_{f_j}(\lambda) \neq 0$ for $j = 1, \dots, m_f$. ■

The following result is a general characterization of the complete strong fault detectability of the system (3.2) in terms of a particular reduced system (5.11) and can serve as an easy check of this property.

Proposition 5.2 *Let Ω be the set of frequencies which characterize the persistent fault signals. For the system (3.2) with $w \equiv 0$ and for $G(\lambda)$ defined in (5.2), let $Q_1(\lambda)$ be a least-order rational basis of $\mathcal{N}_L(G(\lambda))$, such that $Q_1(\lambda)$ and $\overline{G}_f(\lambda)$ in (5.12) have no poles in Ω . Then, the system (3.2) is completely strong fault detectable with respect to Ω if and only if*

$$\overline{G}_{f_j}(\lambda_z) \neq 0, \quad j = 1, \dots, m_f, \quad \forall \lambda_z \in \Omega. \quad (5.14)$$

Proof To prove necessity, we note that complete strong fault detectability implies that there exists a stable filter $Q(\lambda)$ such that the corresponding $R_f(\lambda)$ is stable, and $R_{f_j}(\lambda)$, the j -th column of $R_f(\lambda)$, has no zeros in Ω . According to Theorem 5.1, any filter $Q(\lambda)$ satisfying $Q(\lambda)G(\lambda) = 0$, can be expressed in the form $Q(\lambda) = W(\lambda)Q_1(\lambda)$, where $W(\lambda)$ is a suitable rational matrix. It follows that $R_{f_j}(\lambda) = W(\lambda)\overline{G}_{f_j}(\lambda)$. Assume $\lambda_z \in \Omega$ is a zero of $\overline{G}_{f_j}(\lambda)$, such that $\overline{G}_{f_j}(\lambda_z) = 0$. However, this implies that $R_{f_j}(\lambda_z) = 0$, which contradicts the assumption of complete strong detectability. Therefore, $\overline{G}_{f_j}(\lambda)$ can not have zeros in Ω . This requirement is expressed, for $j = 1, \dots, m_f$, by the conditions (5.14).

To prove sufficiency, we show that for any given basis $Q_1(\lambda)$ without poles in Ω and for $\overline{G}_{f_j}(\lambda)$ without poles and zeros in Ω we can build a stable filter $Q(\lambda)$ such that, $R_{f_j}(\lambda)$ has no zeros in Ω as well. For this we take $Q(\lambda) = M(\lambda)Q_1(\lambda)$, where $[Q_1(\lambda) \overline{G}_f(\lambda)] = M^{-1}(\lambda)[Q(\lambda) R_f(\lambda)]$ is a stable left coprime factorization. The zeros of $M(\lambda)$ are the unstable poles of $[Q_1(\lambda) \overline{G}_f(\lambda)]$. Since by assumption, this TFM has no poles in Ω , it follows that $M(\lambda)$ has no zeros in Ω . Therefore, for any $\lambda_z \in \Omega$, $\det M(\lambda_z) \neq 0$. It follows, for each f_j that if $\overline{G}_{f_j}(\lambda_z) \neq 0$, then $R_{f_j}(\lambda_z) = M(\lambda_z)\overline{G}_{f_j}(\lambda_z) \neq 0$. This proves the complete strong fault detectability with respect to Ω . ■

Remark 5.3 The conditions on the poles of $Q_1(\lambda)$ and $\overline{G}_f(\lambda)$ imposed in Proposition 5.2 are essential to check the complete strong fault detectability. In Example 3.3 with $G_u(s) = 0$, $G_d(s) = 0$ and $G_f(s) = [1 \quad 1/s]$, we can choose $Q_1(s) = s/(s+1)$ to obtain $\overline{G}_f(s) = Q_1(s)G_f(s) = [s/(s+1) \quad 1/(s+1)]$. This system is not completely strong fault detectable with respect to constant faults because $\overline{G}_{f_1}(0) = 0$. The following example shows, that the check of strong fault detectability may lead to an erroneous result if the condition on the poles of $Q_1(\lambda)$ is not fulfilled. □

Example 5.1 Consider the continuous-time system (3.2) from Example 3.1 with

$$G_u(s) = \begin{bmatrix} \frac{1}{s} \\ \frac{1}{s} \end{bmatrix}, \quad G_d(s) = \begin{bmatrix} 0 \\ \frac{s}{s+3} \end{bmatrix}, \quad G_f(s) = \begin{bmatrix} \frac{s+1}{s+2} \\ \frac{1}{s+2} \end{bmatrix}$$

and $\Omega = \{0\}$. This system is not strongly fault detectable. To see this, we employ the check based on Proposition 5.2.

A stable rational (minimal) basis is

$$Q_1(s) = \begin{bmatrix} \frac{s}{s+1} & 0 & -\frac{1}{s+1} \end{bmatrix},$$

which leads to

$$\overline{G}_f(s) = \frac{s}{s+2}.$$

Since $\overline{G}_f(s)$ has a zero in 0, the system is not strongly fault detectable for constant faults.

However, if we use, instead, the rational (minimal) basis with a pole in the origin

$$Q_1(s) = \begin{bmatrix} 1 & 0 & -\frac{1}{s} \end{bmatrix},$$

we obtain

$$\overline{G}_f(s) = \frac{s+1}{s+2},$$

for which, the zeros based check indicates, erroneously, strong fault detectability. \diamond

5.2 Solving the Exact Fault Detection Problem

Using Proposition 5.1, the solvability conditions of the *exact fault detection problem* (EFDP) formulated in Sect. 3.5.1 for the system (3.2) with $w \equiv 0$ can be expressed as fault input observability conditions for the reduced system (5.11) with $w \equiv 0$ according to the following corollary to Theorem 3.7:

Corollary 5.2 *For the system (3.2) with $w \equiv 0$ the EFDP is solvable if and only if the reduced system (5.11) with $w \equiv 0$ is completely fault detectable, or equivalently, the following input observability conditions hold*

$$\overline{G}_{f_j}(\lambda) \neq 0, \quad j = 1, \dots, m_f. \quad (5.15)$$

Using Proposition 5.2, the solvability conditions of the EFDP with the strong detection condition (3.25) can be equivalently expressed as conditions on the lack of zeros in Ω for all columns of the TFM $\overline{G}_f(\lambda)$ of reduced system (5.11) according to the following corollary to Theorem 3.8:

Corollary 5.3 *Let $\Omega \subset \partial C_s$ be a given set of frequencies, and assume that the reduced system (5.11) has been obtained by choosing $Q_1(\lambda)$ without poles in Ω and such that also $\overline{G}_f(\lambda)$ in (5.12) has no poles in Ω . Then, for $w \equiv 0$, the EFDP with the*

strong detection condition (3.25) is solvable if and only if the reduced system (5.11) with $w \equiv 0$ is completely strong fault detectable with respect to Ω , or equivalently, the following conditions hold

$$\overline{G}_{f_j}(\lambda_z) \neq 0, \quad j = 1, \dots, m_f, \quad \forall \lambda_z \in \Omega. \quad (5.16)$$

When solving the EFDP, it is obvious that any stable and proper rational nullspace basis $Q_1(\lambda)$ already represents a solution, provided the complete fault detectability conditions (5.15) or the complete strong fault detectability conditions (5.16) are fulfilled and $\overline{G}_f(\lambda)$ is stable. According to Remark 5.2, the dynamics of both $Q_1(\lambda)$ and $\overline{G}_f(\lambda)$ (i.e., their poles) can be arbitrarily assigned. Moreover, fault detection filters with an arbitrary number of outputs $q \leq p - r_d$ can be easily obtained, by building linear combinations of the rows of $Q_1(\lambda)$.

Example 5.2 Consider a continuous-time system with the transfer function matrices

$$G_u(s) = \begin{bmatrix} \frac{s+1}{s+2} \\ \frac{s+2}{s+3} \end{bmatrix}, \quad G_d(s) = \begin{bmatrix} \frac{1}{s+2} \\ 0 \end{bmatrix}, \quad G_w(s) = 0, \quad G_f(s) = \begin{bmatrix} \frac{s+1}{s+2} & 0 \\ 0 & 1 \end{bmatrix}.$$

A minimal left nullspace basis of $G(\lambda)$ defined in (5.2) for $\lambda = s$ can be obtained in the form (5.5) as $N_l(s) = N_{l,d}(s) \begin{bmatrix} I_2 & -G_u(s) \end{bmatrix}$, with $N_{l,d}(s) = \begin{bmatrix} 0 & 1 \end{bmatrix}$. We obtain $Q_1(s) = N_l(s)$ as

$$Q_1(s) = \begin{bmatrix} 0 & 1 & -\frac{s+2}{s+3} \end{bmatrix}$$

and the TFMs of the reduced system (5.11) are

$$\overline{G}_w(s) = 0, \quad \overline{G}_f(s) = \begin{bmatrix} 0 & 1 \end{bmatrix}.$$

The presence of a zero column in $\overline{G}_f(s)$ indicates that the EFDP has no solution, because the fault f_1 and the disturbance d share the same signal space. By appropriately redefining d and w , we will address this problem in Example 5.5 and show that an approximate solution of this problem is still possible. Note that the filter with $Q(\lambda) = N_l(s)$ can be still used for the detection of f_2 . \diamond

We can exploit in various ways the existing freedom in determining fault detection filters which solve the EFDP. For practical use, it is sometimes advantageous to impose for the number of residual signals q a certain low value, as for example, $q = 1$, which leads to *scalar output* fault detection filters. Of both theoretical and practical interest are fault detection filters which have the least possible order (i.e., least McMillan degree). For example, least-order scalar output fault detection filters can be employed to build banks of scalar output filters with global least-orders to solve the more involved FDIPs.

For the computation of a least-order solution we can choose the factor $\overline{Q}_1(\lambda)$ in (5.9) in the factored form

$$\overline{Q}_1(\lambda) = Q_3(\lambda)Q_2(\lambda),$$

where $Q_2(\lambda)$ is a $q \times (p - r_d)$ proper TFM determined such that $Q_2(\lambda)Q_1(\lambda)$ has least-order, while $Q_3(\lambda)$ is a $q \times q$ proper, stable and invertible TFM determined such that

both the overall filter $Q(\lambda) = Q_3(\lambda)Q_2(\lambda)Q_1(\lambda)$ and $R_f(\lambda) = Q_3(\lambda)Q_2(\lambda)\overline{G}_f(\lambda)$ are stable. The least possible order of the fault detection filter $Q(\lambda)$ is uniquely determined by the fulfillment of a certain admissibility condition. When solving the EFD, we say that the filter $Q(\lambda)$ is *admissible*, if the fault detection conditions (3.24) are fulfilled by the corresponding $R_f(\lambda)$. Thus, an admissible choice of $Q_2(\lambda)$ must guarantee the admissibility of $Q(\lambda)$. Since $Q_3(\lambda)$ is invertible, its choice plays no role in ensuring admissibility. Interestingly, a least-order filter synthesis can be always achieved by a scalar output fault detection filter.

The **Procedure EFD** given below summarizes the main computational steps of the synthesis of least-order fault detection filters. In view of potential applications of **Procedure EFD**, we devised this procedure to be applicable to the complete faulty system (2.1), including also the noise inputs.

Procedure EFD: Exact synthesis of fault detection filters

Inputs : $\{G_u(\lambda), G_d(\lambda), G_f(\lambda), G_w(\lambda)\}, q$

Outputs: $Q(\lambda), R_f(\lambda), R_w(\lambda)$

- 1) Compute a $(p - r_d) \times (p + m_u)$ minimal basis matrix $Q_1(\lambda)$ for the left nullspace of $G(\lambda)$ defined in (5.2), where $r_d := \text{rank } G_d(\lambda)$;
 set $Q(\lambda) = Q_1(\lambda)$ and compute $\begin{bmatrix} R_f(\lambda) & R_w(\lambda) \end{bmatrix} = Q_1(\lambda) \begin{bmatrix} G_f(\lambda) & G_w(\lambda) \\ 0 & 0 \end{bmatrix}$.
Exit if exists $j \in \{1, \dots, m_f\}$ such that $R_{fj}(\lambda) = 0$ (no solution exists).
- 2) Choose a $\min(q, p - r_d) \times (p - r_d)$ rational matrix $Q_2(\lambda)$ such that $Q_2(\lambda)Q(\lambda)$ has least McMillan degree and $Q_2(\lambda)R_{fj}(\lambda) \neq 0, j = 1, \dots, m_f$;
 compute $Q(\lambda) \leftarrow Q_2(\lambda)Q(\lambda), R_f(\lambda) \leftarrow Q_2(\lambda)R_f(\lambda)$ and $R_w(\lambda) \leftarrow Q_2(\lambda)R_w(\lambda)$.
- 3) Choose a proper and stable invertible rational matrix $Q_3(\lambda)$ such that $Q_3(\lambda)Q(\lambda), Q_3(\lambda)R_f(\lambda)$ and $Q_3(\lambda)R_w(\lambda)$ have desired stable dynamics; compute $Q(\lambda) \leftarrow Q_3(\lambda)Q(\lambda), R_f(\lambda) \leftarrow Q_3(\lambda)R_f(\lambda), R_w(\lambda) \leftarrow Q_3(\lambda)R_w(\lambda)$.

This procedure illustrates several computational paradigms common to all synthesis algorithms presented in this book, such as: the use of product form representations of the filter and the use of the associated filter updating techniques, the use of nullspace method as the first computational step, the determination of least-order of the resulting filter on the basis of suitable admissibility conditions, or the arbitrary assignment of filter dynamics using coprime factorization techniques.

The computational details of the above procedure differ according to the type of the employed nullspace basis at Step 1). We consider first the case when at Step 1) of **Procedure EFD**, $Q(\lambda) = Q_1(\lambda)$ is a minimal polynomial basis and the corresponding $R_f(\lambda)$ satisfies $R_{fj}(\lambda) \neq 0$ for $j = 1, \dots, m_f$. For simplicity, we determine a least-order fault detection filter with scalar output (i.e., for $q = 1$). At Step 2) we have to determine $Q_2(\lambda) = \phi(\lambda)$, where $\phi(\lambda)$ is a polynomial vector, such that $\phi(\lambda)Q_1(\lambda)$ has least degree and $\phi(\lambda)R_{fj}(\lambda) \neq 0$ for $j = 1, \dots, m_f$. Assume $Q_1(\lambda)$ is formed of $p - r_d$ row vectors $v_i(\lambda)$, where $v_i(\lambda)$ is a polynomial basis vector of degree n_i . We assume that the basis vectors $v_i(\lambda)$ are ordered such that $n_1 \leq n_2 \leq \dots \leq n_{p-r_d}$. We can easily construct linear combinations of basis vectors of final degree n_i , for $i = 1, \dots, p - r_d$, by choosing $\phi(\lambda) = \phi^{(i)}(\lambda)$, with

$$\phi^{(i)}(\lambda) = [\phi_1^{(i)}(\lambda) \dots \phi_i^{(i)}(\lambda) 0 \dots 0], \quad (5.17)$$

where $\phi_j^{(i)}(\lambda)$ is a polynomial of maximum degree $n_i - n_j$ and $\phi_i^{(i)}(\lambda)$ is a nonzero constant value. The achievable least-order can be determined by successively constructing linear combinations of polynomials with increasing degrees $n_1, n_2, \dots, n_{p-r_d}$ (e.g., with randomly generated coefficients). For each trial degree n_i , the condition $\phi^{(i)}(\lambda)R_{f_j}(\lambda) \neq 0$ for $j = 1, \dots, m_f$ is checked. The search stops for the first value of i for which this condition is fulfilled. At Step 3) we can often choose $Q_3(\lambda) = 1/d(\lambda)$, with $d(\lambda)$ a polynomial of degree n_i with only stable roots. However, if the resulting $Q_3(\lambda)R_f(\lambda)$ is not stable or not proper, then $Q_3(\lambda)$ must be computed to also enforce the stability of $Q_3(\lambda)R_f(\lambda)$ as well as of $Q_3(\lambda)R_w(\lambda)$. This can be achieved by replacing $Q(\lambda)$, $R_f(\lambda)$ and $R_w(\lambda)$ resulted at Step 2) with the proper and stable factors $\tilde{Q}(\lambda)$, $\tilde{R}_f(\lambda)$ and $\tilde{R}_w(\lambda)$, respectively, resulting from a LCF with proper and stable factors

$$[Q(\lambda) R_f(\lambda) R_w(\lambda)] = Q_3^{-1}(\lambda)[\tilde{Q}(\lambda) \tilde{R}_f(\lambda) \tilde{R}_w(\lambda)], \quad (5.18)$$

where the poles of the scalar transfer function $Q_3(\lambda)$ can be freely assigned.

The polynomial nullspace approach allows to easily solve the least-order synthesis problem of fault detection filters with scalar outputs. The least-order is bounded below by n_i , the degree of the i -th basis vector, where i is the first index for which there exists a $\phi^{(i)}(\lambda)$ of the form (5.17) such that $\phi^{(i)}(\lambda)R_{f_j}(\lambda) \neq 0$ for $j = 1, \dots, m_f$ (with $R_f(\lambda)$ computed at Step 1) of **Procedure EFD**). The value n_i for the McMillan degree of the final filter $Q(\lambda)$ can be often achieved, as for example, when $G_f(\lambda)$ and $G_w(\lambda)$ are already stable and proper.

Remark 5.4 The Step 2) of this synthesis procedure can be significantly simplified by determining directly the least degree of candidate polynomial vectors suited to solve the EFDP, instead of iterating with candidate vectors of increasing orders. For this purpose, we can use the $(p - r_d) \times m_f$ structure matrix S_{R_f} associated to the resulting $R_f(\lambda)$ at Step 1) of **Procedure EFD**. Let S_{R_f} be the binary matrix (see Sect. 3.4), whose the (i, j) -th element is set to 1 if the (i, j) -th element of $R_f(\lambda)$ is nonzero, and otherwise is set to 0. Let i be the least row index such that the leading i rows of S_{R_f} contain at least one nonzero element in all columns. It follows, that we can build, using a polynomial vector $\phi^{(i)}(\lambda)$ of the form (5.17), a linear combination of the first i basis vectors of least degree n_i , such that all faults can be detected. A straightforward simplification is to use, instead of the polynomial vector $\phi^{(i)}(\lambda)$ in (5.17), a constant vector (with the same structure)

$$h^{(i)} = [h_1, \dots, h_i, 0, \dots, 0], \quad (5.19)$$

with $h_j \neq 0, j = 1, \dots, i$, to build a linear combination of basis vectors up to degree n_i (e.g., using randomly generated values). The nonzero components of $h^{(i)}$ can be interpreted as weighting factors of the individual basis vectors. Therefore, an optimal

choice of these weights can maximize the overall sensitivity of residual to faults. Suitable fault sensitivity measures for this purpose are discussed in Remark 5.6. \square

Remark 5.5 Although the EFDP can be always solved using a scalar output fault detection filter of least dynamical order, there may exist advantages when using filters with more than one output. First, it may be possible with a residual vector with several components to enforce a more uniform sensitivity of the residual vector to individual fault components. This aspect is related to an increased number of free parameters which can be thus optimally chosen (see Remark 5.6). A second potential advantage is that with several residual outputs, it may be possible to also achieve a certain block isolation of group of faults. For example, suitable combinations of individual basis vectors in $Q_1(\lambda)$ can be easily constructed using the binary information coded in the structure matrix S_{R_f} associated to the resulting $R_f(\lambda)$ at Step 1) of the **Procedure EFD**. This can be advantageous especially in the case when the expected magnitudes of the residual signals may significantly vary for different groups of faults. A more involved synthesis procedure to achieve block isolation can be performed using several scalar output filters, where each filter is designed to be sensitive to a group of faults and insensitive to the rest of faults (see **Procedure EFDI** in Sect. 5.4). \square

When using a proper rational basis instead a polynomial one at Step 1) of the **Procedure EFD**, a synthesis approach leading directly to a proper filter can be devised. Assume $Q_1(\lambda)$ is a simple minimal proper rational basis (see Sect. 9.1.3 for the definition of simple bases) formed of $p-r_d$ rational row vectors $v_i(\lambda)/d_i(\lambda)$, where $v_i(\lambda)$ is a polynomial vector of degree n_i and $d_i(\lambda)$ is a stable polynomial of degree n_i . We assume that the vectors $v_i(\lambda)$ are the basis vectors of a minimal polynomial basis, ordered such that $n_1 \leq n_2 \leq \dots \leq n_{p-r_d}$, and each denominator $d_i(\lambda)$ divides $d_j(\lambda)$ for $i < j$. It follows immediately, that a linear combination $h^{(i)}Q_1(\lambda)$ of the first i rows with $h^{(i)}$ of the form (5.19) has a McMillan degree n_i . At Step 2), choosing the least index i such that $h^{(i)}R_f(\lambda) \neq 0$ for $j = 1, \dots, m_f$, allows to take $Q_2(\lambda) := h^{(i)}$. Often the choice $Q_3(\lambda) = 1$ at Step 3) solves the synthesis problem. However, if $R_f(\lambda)$ is unstable or not proper, then the same computational approach, based on the LCF in (5.18), can be used as in the case of a polynomial basis.

The **Procedure EFD** employing polynomial or simple proper nullspace bases involves polynomial manipulations and therefore is not a reliable computational approach for large order systems due to the intrinsic high sensitivity of polynomial-based representations. A numerically reliable alternative algorithm employs minimal (non-simple) proper bases and is based on state-space computations described in details in Sect. 7.4 (see also Sect. 10.3.2). The importance of **Procedure EFD**, and especially of the synthesis with least-order scalar fault detection filters, lies in being the basic computational procedure which allows to solve the more involved fault detection and isolation problem formulated in Sect. 3.5.3.

Remark 5.6 Steps 2) and 3) of **Procedure EFD** can be easily embedded into an optimization-based tuning procedure to determine an optimal $Q_2(\lambda)$ which ensures a more uniform sensitivity of the detector to individual faults. The free parameters to be tuned are the polynomial coefficients of $\phi^{(i)}(\lambda)$ in (5.17) or the nonzero components

of the real vector $h^{(i)}$ in (5.19). It is assumed that for given values of these parameters at Step 2), the computations at Step 3) follow automatically to produce a stable candidate solution $Q(\lambda)$. For optimal tuning of parameters, the *sensitivity condition* can be used as a criterion to be minimized. For a given $R_f(\lambda)$, this criterion is defined as

$$\xi := \max_j \|R_{f_j}(\lambda)\|_\infty / \min_j \|R_{f_j}(\lambda)\|_\infty. \quad (5.20)$$

For tuning based on strong fault detectability, a similar *sensitivity condition* can be defined in terms of the gains at a selected frequency λ_s as

$$\xi^s := \max_j \|R_{f_j}(\lambda_s)\|_2 / \min_j \|R_{f_j}(\lambda_s)\|_2. \quad (5.21)$$

A large value of the sensitivity condition ξ (or ξ^s) indicates potential difficulties in detecting faults due to a substantial gap between the maximum and minimum gains. In such cases, employing fault detection filters with several outputs ($q > 1$) could be advantageous. \square

Example 5.3 Consider a continuous-time system with the TFMs

$$G_u(s) = \begin{bmatrix} \frac{s+1}{s+2} \\ \frac{s+2}{s+3} \end{bmatrix}, \quad G_d(s) = \begin{bmatrix} \frac{s-1}{s+2} \\ 0 \end{bmatrix}, \quad G_w(s) = 0, \quad G_f(s) = \begin{bmatrix} \frac{s+1}{s+2} & 0 \\ \frac{s+2}{s+3} & 1 \end{bmatrix}.$$

The fault f_1 corresponds to an additive actuator fault, while f_2 describes an additive sensor fault in the second output y_2 . The TFM $G_d(s)$ is non-minimum phase, having an unstable zero at 1.

At Step 1) of the **Procedure EFD**, a proper minimal left nullspace basis can be determined, consisting of a single row vector, which we can choose, for example,

$$Q_1(s) = \left[0 \quad 1 \quad -\frac{s+2}{s+3} \right].$$

For the reduced system (5.11) computed at Step 1) we obtain

$$R_f(s) = \overline{G}_f(s) = \begin{bmatrix} \frac{s+2}{s+3} & 1 \end{bmatrix},$$

which shows that according to Corollary 5.2 the EFDP has a solution. Since this basis is already stable, $Q(s) = Q_1(s)$ is a least-order solution of the EFDP. \diamond

Example 5.4 Consider an unstable continuous-time system with the TFMs

$$G_u(s) = \begin{bmatrix} \frac{s+1}{s-2} \\ \frac{s+2}{s-3} \end{bmatrix}, \quad G_d(s) = \begin{bmatrix} \frac{s-1}{s+2} \\ 0 \end{bmatrix}, \quad G_w(s) = 0, \quad G_f(s) = \begin{bmatrix} \frac{s+1}{s-2} & 0 \\ \frac{s+2}{s-3} & 1 \end{bmatrix},$$

where as before, the fault f_1 corresponds to an additive actuator fault, while f_2 describes an additive sensor fault in the second output y_2 , with the difference that the underlying system is unstable. The TFM $G_d(s)$ is non-minimum phase, having an unstable zero at 1.

At Step 1) of the **Procedure EFD**, a proper minimal left nullspace basis can be determined, consisting of a single row vector, which we can choose, for example,

$$Q_1(s) = \begin{bmatrix} 0 & 1 & -\frac{s+2}{s-3} \end{bmatrix}.$$

For the reduced system (5.11) computed at Step 1) we obtain

$$R_f(s) = \bar{G}_f(s) = \begin{bmatrix} \frac{s+2}{s-3} & 1 \end{bmatrix},$$

which shows that according to Corollary 5.2 the EFDP has a solution. Since $Q(s) = Q_1(s)$ is unstable, it must be suitably updated. With $Q_2(s) = 1$ at Step 2) and $Q_3(s) = \frac{s-3}{s+3}$ at Step 3) we finally obtain

$$Q(s) = \begin{bmatrix} 0 & \frac{s-3}{s+3} & -\frac{s+2}{s+3} \end{bmatrix}, \quad R_f(s) = \begin{bmatrix} \frac{s+2}{s+3} & \frac{s-3}{s+3} \end{bmatrix}.$$

The script **Ex5_4** in Listing 5.1 solves the considered EFDP, by computing intermediary results which differ from those of this example. The script **Ex5_4c** (not listed) is a compact version of this script, which calls the function **efdsyn**, a prototype implementation of **Procedure EFD**. \diamond

Listing 5.1 Script **Ex5_4** to solve the EFDP of Example 5.4 using **Procedure EFD**

```

% Uses the Control Toolbox and the Descriptor System Tools

% define s as an improper transfer function
s = tf('s');
% define Gu(s), Gd(s), Gf(s)
Gu = [(s+1)/(s-2); (s+2)/(s-3)]; % enter Gu(s)
Gd = [(s-1)/(s+2); 0]; % enter Gd(s)
Gf = [(s+1)/(s-2) 0; (s+2)/(s-3) 1]; % enter Gf(s)
p = 2; mu = 1; md = 1; mf = 2; % set dimensions

% compute a left nullspace basis Q1 of [Gu Gd; I0] and Rf,1 = Q1[Gf; 0]
Q1 = glnull(ss([Gu Gd; eye(mu, mu+md)]));
Rf1 = gir(Q1*[Gf; zeros(mu, mf)]);

% check solvability using a random frequency
if min(abs(evalfr(Rf1, rand))) > 0.01
    % compute a stable left coprime factorization [Q1 Rf,1] = Q3^-1[Q Rf]
    % enforce stability degree -3
    [Q_Rf, Q3] = glcf([Q1, Rf1], struct('sdeg', -3));
    % extract Q and Rf
    Q = Q_Rf(:, 1:p+mu); Rf = Q_Rf(:, p+mu+1:end);
    % normalize Q and Rf to match example
    sc = evalfr(Rf(1,1), inf); Q = tf(Q/sc); Rf = tf(Rf/sc);
else
    disp('No solution exists')
end

```

5.3 Solving the Approximate Fault Detection Problem

Using the factorized representation $Q(\lambda) = \bar{Q}_1(\lambda)Q_1(\lambda)$ in (5.9) with $Q_1(\lambda)$ chosen proper and stable, it follows that $Q(\lambda)$ solves the *approximate fault detection problem* (AFDP) formulated in Sect. 3.5.2 for the system (3.2) if and only if $\bar{Q}_1(\lambda)$ solves the AFDP for the reduced system (5.11). By a suitable choice of $Q_1(\lambda)$ we can

always additionally enforce that both $\overline{G}_w(\lambda)$ and $\overline{G}_f(\lambda)$ in (5.11) are proper, which will be assumed throughout this section. The solvability conditions of the AFDP for the system (3.2) can be replaced by similar conditions for the reduced system (3.2) according to the following corollary to Theorem 3.9:

Corollary 5.4 *For the system (3.2) the AFDP is solvable if and only if the system (5.11) is completely fault detectable, or equivalently, the following input observability conditions hold*

$$\overline{G}_{f_j}(\lambda) \neq 0, \quad j = 1, \dots, m_f.$$

We have seen in the proof of Theorem 3.9, that a solution of the AFDP can be determined by solving the related EFDP with $w \equiv 0$, using, for example, **Procedure EFD**. The usefulness of such a solution can be assessed in terms of the magnitudes of the minimum size detectable fault inputs in the presence of noise inputs. While for small noise levels such a solution may often be satisfactory, for large noise levels a purposely designed fault detection filter, which maximizes the magnitudes of the minimum size detectable fault inputs for the given class of noise inputs, usually represents a better solution. Such a solution, which aims to maximize the sensitivity of residual to faults and, simultaneously, to minimize the effects of noise on the residual, can be targeted by solving a suitably formulated optimization problem.

Consider a fault detection filter $Q(\lambda)$, in the general parameterized form (5.9), which has the internal form

$$\mathbf{r}(\lambda) := R_f(\lambda)\mathbf{f}(\lambda) + R_w(\lambda)\mathbf{w}(\lambda).$$

Let $\gamma > 0$ be an admissible level for the effect of the noise signal $w(t)$ on the residual $r(t)$, which can be imposed, for example, as

$$\|R_w(\lambda)\|_{2/\infty} \leq \gamma, \quad (5.22)$$

where $\|\cdot\|_{2/\infty}$ denotes either the \mathcal{H}_2 - or \mathcal{H}_∞ -norm. The \mathcal{H}_2 -norm corresponds to the case when $w(t)$ is a white noise signal, while the \mathcal{H}_∞ -norm is better suited when $w(t)$ is an unknown signal with bounded energy (or power). The choice of γ usually reflects the desired robustness of the fault detection filter to reject the noise. The value $\gamma = 0$ can be used to formulate the EFDP as a particular AFDP. For $\gamma > 0$ it is always possible, via a suitable scaling of the filter, to use the normalized value $\gamma = 1$.

As measures of the sensitivity of residuals to faults, several “indices” have been proposed in the literature to characterize the least sensitivity in terms of $R_f(\lambda)$. Such an index, commonly denoted by $\|R_f(\lambda)\|_-$, has been defined in terms of the least singular value (denoted by $\underline{\sigma}(\cdot)$) of the frequency response of $R_f(\lambda)$ as

$$\|R_f(\lambda)\|_- := \inf_{\omega \in \Omega} \underline{\sigma}(R_f(\omega)), \quad (5.23)$$

where $\Omega \subset \partial\mathbb{C}_s$ is a finite or infinite set of frequency values on the boundary of the appropriate stability domain. In view of Definition 3.4 (see Sect. 3.3), the requirement $\|R_f(\lambda)\|_- > 0$ can be interpreted as a complete strong fault detectability condition. In some works, the formulation of the AFDP involves the determination of a filter $Q(\lambda)$ which maximizes the index (5.23) such that the noise attenuation constraint (5.22) is simultaneously fulfilled. This binding of the formulation of the AFDP to a particular optimization-based solution method is generally not desirable, since it imposes additional constraints, usually of purely technical character, on the solvability of the AFDP. While the satisfaction of such constraints guarantees the solvability of the underlying mathematical optimization problem, these conditions are usually not necessary for the solvability of the AFDP (according to the formulation in Sect. 3.5.2). Two inherent weaknesses in the definition of the index $\|R_f(\lambda)\|_-$ worsen additionally the solvability of the optimization-based formulation of the AFDP.

A first issue is that the index (5.23) is meaningful only when $m_f \leq p$, because if $m_f > p$, only the detectability of p out of m_f faults can be assessed by this index, $m_f - p$ singular values being null. It was argued that the case $m_f > p$ can be addressed using a bank of filters, where each filter must be sensitive only to a subset of maximal p faults. However, this leads to an unnecessary increase of the global order of the resulting fault detection filter and therefore represents a strong technical limitation for practical use. The second issue is rather of conceptual nature. The definition (5.23) targets primarily the complete strong fault detectability aspect (see Definition 3.4), and therefore appears to be less adequate to characterize the weaker property of complete fault detectability (see Definition 3.2), which merely requires that each column of $R_f(\lambda)$ must be nonzero. While this property can be still indirectly targeted, for example, by a suitable choice of Ω (e.g., $\Omega = \{\lambda_0\}$ with λ_0 a representative frequency value at which $R_{f_j}(\lambda_0)$ must be nonzero for $j = 1, \dots, m_f$), an alternative index, discussed in what follows, is better suited to address directly the complete fault detectability aspect.

To overcome both these deficiencies, an alternative index will be used to characterize fault sensitivity. This index is defined as

$$\|R_f(\lambda)\|_{2/\infty-} := \min_{1 \leq j \leq m_f} \|R_{f_j}(\lambda)\|_{2/\infty}, \quad (5.24)$$

where $\|\cdot\|_{2/\infty}$ stays for either $\|\cdot\|_2$ or $\|\cdot\|_\infty$, while $\|\cdot\|_{2/\infty-}$ stays for either $\|\cdot\|_{2-}$ or $\|\cdot\|_{\infty-}$ indices defined in terms of \mathcal{H}_2 or \mathcal{H}_∞ norms in (5.24), respectively. The requirement $\|R_f(\lambda)\|_{2/\infty-} > 0$ merely asks that all columns of $R_{f_j}(\lambda)$ are nonzero, and therefore, the index $\|R_f(\lambda)\|_{2/\infty-}$ characterizes the *complete fault detectability* (of an arbitrary number of faults) as defined in Definition 3.2. To characterize the *complete strong fault detectability with respect to Ω* , the modified index $\|\cdot\|_{\Omega-}$ can be used, defined as

$$\|R_f(\lambda)\|_{\Omega-} := \min_{1 \leq j \leq m_f} \left\{ \inf_{\omega \in \Omega} \|R_{f_j}(\omega)\|_2 \right\}. \quad (5.25)$$

For a particular problem, a combination of the two indices (5.24) and (5.25) can also be meaningful, by selecting for the j -th column of $R_f(\lambda)$ either $\|R_{f_j}(\lambda)\|_{2/\infty-}$ or $\|R_{f_j}(\lambda)\|_{\Omega-}$ as a problem specific fault sensitivity measure.

Using the above definitions of the $\|\cdot\|_{2/\infty-}$ and $\|\cdot\|_{\Omega-}$ indices, several optimization problems can be formulated to address the computation of a satisfactory solution of the AFDP for the reduced system (5.11) with $\overline{G}_w(\lambda)$ and $\overline{G}_f(\lambda)$ proper, using the parametrization (5.9) of the fault detection filter with stable $\overline{Q}_1(\lambda)$. In what follows, we only discuss one of the most popular formulations, the $\mathcal{H}_{\infty-}/\mathcal{H}_{\infty}$ synthesis, for which we give a detailed computational procedure. The synthesis goal is to determine $\overline{Q}_1(\lambda)$ which maximizes the fault sensitivity for a given level of noise: Given $\gamma \geq 0$, determine the stable and proper optimal fault detection filter $\overline{Q}_1(\lambda)$ and the corresponding optimal fault sensitivity level $\beta > 0$ such that

$$\beta = \max_{\overline{Q}_1(\lambda)} \left\{ \|\overline{Q}_1(\lambda)\overline{G}_f(\lambda)\|_{\infty-} \mid \|\overline{Q}_1(\lambda)\overline{G}_w(\lambda)\|_{\infty} \leq \gamma \right\}. \quad (5.26)$$

An alternative formulation of an optimization-based solution, called the $\mathcal{H}_{\infty}/\mathcal{H}_{\infty-}$ synthesis, minimizes the effects of noise by imposing a certain fault sensitivity level: Given $\beta > 0$, determine $\gamma \geq 0$ and a stable and proper fault detection filter $\overline{Q}_1(\lambda)$ such that

$$\gamma = \min_{\overline{Q}_1(\lambda)} \left\{ \|\overline{Q}_1(\lambda)\overline{G}_w(\lambda)\|_{\infty} \mid \|\overline{Q}_1(\lambda)\overline{G}_f(\lambda)\|_{\infty-} \geq \beta \right\}. \quad (5.27)$$

The two approaches may lead to different solutions, depending on the properties of the underlying transfer function matrices and problem dimensions. For both cases, the gap β/γ can be interpreted as a measure of the quality of fault detection. For $\gamma = 0$, both formulations include the exact solution (i.e., of the EFDP for $w \equiv 0$) and the corresponding gap is infinite.

Before we discuss the computational issues, we consider a simple example which highlights the roles of fault and noise input signals when solving an AFDP.

Example 5.5 This is the same as Example 5.2, where we redefined the noise input w as d and thus we have

$$G_u(s) = \begin{bmatrix} \frac{s+1}{s+2} \\ \frac{s+2}{s+2} \\ \frac{s+2}{s+3} \end{bmatrix}, \quad G_d(s) = 0, \quad G_w(s) = \begin{bmatrix} \frac{1}{s+2} \\ 0 \end{bmatrix}, \quad G_f(s) = \begin{bmatrix} \frac{s+1}{s+2} & 0 \\ 0 & 1 \end{bmatrix}.$$

A minimal basis is simply $N_I(s) = [I_2 \quad -G_u(s)]$, which leads to $\overline{G}_w(s) = G_w(s)$ and $\overline{G}_f(s) = G_f(s)$. This basis is in fact a solution of an EFDP in the case $w \equiv 0$. Thus, this solution can be also employed to solve the AFDP, as pointed out in the proof of Theorem 3.9. To be useful for practical purposes, a fault detection filter must provide reliable detection of all faults in the presence of noise. This condition is evidently fulfilled by the fault input f_2 , since with $Q(s) = N_I(s)$, the second component of the residual r_2 is simply $r_2 = f_2$, because there is no any interaction between the noise input w and fault input f_2 . However, because f_1 and w share the same input space, the minimal detectable size of f_1 will depend on the possible maximum size of noise input w . Assume $\|w\|_2 \leq \delta_w$, thus for all w we have $\|G_w(s)\mathbf{w}(s)\|_2 \leq \|\overline{G}_w(s)\|_{\infty}\|\mathbf{w}(s)\|_2 \leq \delta_w/2$. Thus,

the minimum size of detectable faults $f_{1,min}$ satisfies $\|G_{f_1}(s)\mathbf{f}_{1,min}(s)\|_2 > \delta_w/2$. The solution of this problem depends on the classes of faults considered. Assuming $\mathbf{f}_{1,min}(s) = \eta/s$ (thus a step input fault of amplitude η), the resulting asymptotic value of $G_{f_1}(s)\mathbf{f}_{1,min}(s)$ is $G_{f_1}(0)\eta = \eta/2$. It follows that we can reliably detect constant faults, provided their amplitude satisfies $\eta > \delta_w$. More generally, for step inputs in f_1 , the condition $\eta > \|G_w(s)\|_\infty \delta_w / G_{f_1}(0)$ must be fulfilled for reliable detection. Similar conditions can be established in the case of sinusoidal fault inputs. \diamond

To solve the $\mathcal{H}_{\infty-}/\mathcal{H}_{\infty}$ optimization problem (5.26), we devise a synthesis procedure based on successive simplifications of the original problem by reducing it to simpler problems with the help of a factorized representation of the fault detection filter. We start with the factorized representation (5.9) of the fault detection filter $Q(\lambda)$, where $Q_1(\lambda)$ is a left nullspace basis of $G(\lambda)$ in (5.2) and $\bar{Q}_1(\lambda)$ has to be determined. Let $\bar{G}_f(\lambda)$ and $\bar{G}_w(\lambda)$ be the TFMs of the reduced system (5.11) determined according to (5.12). We can immediately check the solvability conditions of the AFDP of Corollary 5.2 as $\|\bar{G}_f(\lambda)\|_{\infty-} > 0$. Assume that this test indicates the solvability of the AFDP. In this context, we introduce a useful concept to simplify the presentation. A fault detection filter $Q(\lambda)$ is called *admissible* if the corresponding $R_f(\lambda)$ satisfies $\|R_f(\lambda)\|_{\infty-} > 0$ (i.e., it has all its columns nonzero).

Let q be the desired number of residual components. As in the case of an EFDP, if a solution of the AFDP exists, then generally we can always use a scalar output fault detection filter (thus choose $q = 1$). However, larger values of q can be advantageous, because generally involve more free parameters which can be appropriately tuned. In the proposed synthesis procedure (see **Procedure AFD**), the choice of q is restricted to $q \leq r_w \leq p - r_d$, where $r_w := \text{rank } \bar{G}_w(\lambda)$ and $r_d := \text{rank } G_d(\lambda)$. This choice is, however, only for convenience, because it leads to a simpler synthesis procedure. As shown in Remark 5.10, in practical applications q must only satisfy $q \leq p - r_d$, which limits q to the maximum number of left nullspace basis vectors of $G(\lambda)$ in (5.2) (i.e., the number of rows of $Q_1(\lambda)$). This bound on q is the same as in the case of solving the EFDP.

At the next step, we use a factorized representation of $\bar{Q}_1(\lambda)$ in the form $\bar{Q}_1(\lambda) = \bar{Q}_2(\lambda)Q_2(\lambda)$, where the $r_w \times (p - r_d)$ factor $Q_2(\lambda)$ is determined such that $Q_2(\lambda)\bar{G}_w(\lambda)$ has full row rank r_w , and the product $Q_2(\lambda)Q_1(\lambda)$ is admissible and has least McMillan degree. If this latter requirement is not imposed, then a simple choice is $Q_2(\lambda) = H$, where H is a $r_w \times (p - r_d)$ full row rank constant matrix which ensures admissibility (e.g., chosen as a randomly generated matrix with orthonormal columns). This choice corresponds to building $Q_2(\lambda)Q_1(\lambda)$ as r_w linear combinations of the left nullspace basis vectors contained in the rows of $Q_1(\lambda)$.

At this stage, the optimization problem to be solved falls in one of two categories. The *standard case* is when $Q_2(\lambda)\bar{G}_w(\lambda)$ has no zeros on the boundary of the stability domain $\partial\mathbb{C}_s$ (i.e., on the extended imaginary axis in the continuous-time case, or on the unit circle centred in the origin in the discrete-time case). The *nonstandard case* corresponds to the presence of such zeros. This categorization can be easily revealed at the next step, which also involves the computation of the respective zeros. For the full row rank TFM $Q_2(\lambda)\bar{G}_w(\lambda)$ we compute the quasi-co-outer-co-inner factorization

$$Q_2(\lambda)\bar{G}_w(\lambda) = G_{wo}(\lambda)G_{wi}(\lambda), \quad (5.28)$$

where the quasi-co-outer factor $G_{wo}(\lambda)$ is invertible, having only stable zeros excepting possible zeros on the boundary of the stability domain, and $G_{wi}(\lambda)$ is co-inner (i.e., $G_{wi}(\lambda)G_{wi}^*(\lambda) = I$ with $G_{wi}^*(s) = G_{wi}^T(-s)$ in the continuous-time case, and $G_{wi}^*(z) = G_{wi}^T(1/z)$ in the discrete-time case).

We choose $\bar{Q}_2(\lambda) = \bar{Q}_3(\lambda)Q_3(\lambda)$, with $Q_3(\lambda) = G_{wo}^{-1}(\lambda)$ and $\bar{Q}_3(\lambda)$ to be determined. Using (5.10)–(5.12), the fault detection filter in (3.3) can be rewritten as

$$\mathbf{r}(\lambda) = \bar{Q}_3(\lambda)Q_3(\lambda)Q_2(\lambda)\bar{\mathbf{y}}(\lambda) = \bar{Q}_3(\lambda)\tilde{\mathbf{y}}(\lambda), \quad (5.29)$$

where

$$\tilde{\mathbf{y}}(\lambda) := Q_3(\lambda)Q_2(\lambda)\bar{\mathbf{y}}(\lambda) = \tilde{G}_f(\lambda)\mathbf{f}(\lambda) + G_{wi}(\lambda)\mathbf{w}(\lambda), \quad (5.30)$$

with

$$\tilde{G}_f(\lambda) := Q_3(\lambda)Q_2(\lambda)\bar{G}_f(\lambda). \quad (5.31)$$

It follows, that $\bar{Q}_3(\lambda)$ can be determined as the solution of

$$\beta = \max_{\bar{Q}_3(\lambda)} \{ \|\bar{Q}_3(\lambda)\tilde{G}_f(\lambda)\|_{\infty-} \mid \|\bar{Q}_3(\lambda)\|_{\infty} \leq \gamma \}, \quad (5.32)$$

where we used that $\|\bar{Q}_3(\lambda)G_{wi}(\lambda)\|_{\infty} = \|\bar{Q}_3(\lambda)\|_{\infty}$.

In the standard case, we can always ensure that both the partial filter defined by the product of stable factors $Q_3(\lambda)Q_2(\lambda)Q_1(\lambda)$ and $\tilde{G}_f(\lambda)$ are stable. Thus, $\bar{Q}_3(\lambda)$ is determined as $\bar{Q}_3(\lambda) = Q_4$, where Q_4 is a constant matrix representing the optimal solution of the reduced problem

$$\beta = \max_{Q_4} \{ \|Q_4\tilde{G}_f(\lambda)\|_{\infty-} \mid \|Q_4\|_{\infty} \leq \gamma \},$$

such that the resulting detector $Q(\lambda) = Q_4Q_3(\lambda)Q_2(\lambda)Q_1(\lambda)$ is admissible. For square $Q_4(\lambda)$, $Q_4 = \gamma I$ is the simplest $\mathcal{H}_{\infty-}/\mathcal{H}_{\infty}$ optimal solution.

We give the following result without proof. For proofs in continuous- and discrete-time, see [77, 78], respectively.

Theorem 5.2 *For the reduced system (5.11) and with a suitable choice of $Q_2(\lambda)$ assume that we have $\|Q_2(\lambda)\bar{G}_f(\lambda)\|_{\infty-} > 0$, $Q_2(\lambda)\bar{G}_w(\lambda)$ has full row rank and $Q_2(\lambda)\bar{G}_w(\lambda)$ has no zeros on the boundary of the stability domain. Then, for $\gamma > 0$ the $\mathcal{H}_{\infty-}/\mathcal{H}_{\infty}$ optimal solution of the optimization problem (5.26) is*

$$\bar{Q}_{1,opt}(\lambda) := \gamma G_{wo}^{-1}(\lambda)Q_2(\lambda),$$

where $G_{wo}(\lambda)$ is the co-outer factor of the co-outer–co-inner factorization (5.28).

In the nonstandard case, both the partial detector $\tilde{Q}(\lambda) := Q_3(\lambda)Q_2(\lambda)Q_1(\lambda)$ and $\tilde{G}_f(\lambda)$ can result unstable or improper due to the presence of poles on the boundary of the stability domain in the factor $Q_3(\lambda) = G_{wo}^{-1}(\lambda)$. In this case, we choose $\bar{Q}_3(\lambda) = Q_5Q_4(\lambda)$, where $Q_4(\lambda)$ results from a LCF with stable and proper factors

$$[\tilde{Q}(\lambda) \tilde{G}_f(\lambda)] = Q_4^{-1}(\lambda)[\hat{Q}(\lambda) \hat{G}_f(\lambda)],$$

while Q_5 is a constant matrix which solves

$$\beta = \max_{Q_5} \{ \|Q_5 \hat{G}_f(\lambda)\|_{\infty} \mid \|Q_5 Q_4(\lambda)\|_{\infty} \leq \gamma \}.$$

Since $Q_4(\lambda)$ can be always chosen diagonal and such that its diagonal elements have \mathcal{H}_{∞} -norms equal to 1, this choice will significantly simplify the solution of the above problem. For example, the choice $Q_5 = \gamma I$ is always a possibility to obtain a fault detection filter.

Remark 5.7 The presence of unstable zeros of $G_{wo}(\lambda)$ on the boundary of the stability domain prevents the computation of an “optimal” solution of the \mathcal{H}_{∞} -/ \mathcal{H}_{∞} -optimization problem. When solving practical applications, this apparent limitation is superfluous, because the presence of these zeros represents in fact an advantage rather than a disadvantage. For example, in the case of a continuous-time system, a zero at infinity (e.g., in the case when the original $G_w(s)$ is strictly proper) confers to $G_{wo}(s)$ a low-pass character as well, such that high-frequency noise will be attenuated in the noise input channel. Similarly, a zero in the origin will cancel all constant variations in the noise, thus will also attenuate slowly varying noise inputs. Finally, a pair of conjugated zeros on the imaginary axis will attenuate all sinusoidal noise signals of nearby frequencies. This behaviour is thus very similar to that of notch filters, which are purposely included in the feedback loops to address disturbance attenuation or rejection problems in control systems design. The above approach for the nonstandard case simply copes with the presence of zeros on the boundary of the stability domain. \square

Remark 5.8 In the nonstandard case, we can alternatively regularize the problem by replacing $G_{wo}(\lambda)$ in (5.28) by $G_{wo,\varepsilon}(\lambda)$, which, for $\varepsilon > 0$, is a minimum-phase spectral factor satisfying

$$G_{wo,\varepsilon}(\lambda)G_{wo,\varepsilon}^{\sim}(\lambda) = \varepsilon^2 I + G_{wo}(\lambda)G_{wo}^{\sim}(\lambda).$$

By choosing $\bar{Q}_2(\lambda) = \bar{Q}_3(\lambda)Q_3(\lambda)$ with $Q_3(\lambda) = G_{wo,\varepsilon}^{-1}(\lambda)$, we arrive to the same optimization problem (5.32) for $\bar{Q}_3(\lambda)$ as for the standard case. The solution of the AFDP along this line has been discussed in [52]. \square

The dynamical order of the resulting residual generator in the standard case, is the order of $Q_3(\lambda)$ if we choose $Q_4(\lambda)$ a constant matrix. This order results from the conditions that $Q_2(\lambda)\bar{G}_w(\lambda)$ has full row rank and $Q_2(\lambda)Q_1(\lambda)$ has least-order and is admissible (i.e., $\|Q_2(\lambda)\bar{G}_f(\lambda)\|_{\infty} > 0$). For each candidate $Q_2(\lambda)$, the corresponding optimal $Q_3(\lambda)$ results automatically, but the different “optimal” detectors for the same level γ of noise attenuation performance can have significantly differing fault detection performance levels (measured via the optimal cost β). Finding the best compromise between achieved order and the achieved performance (measured

via the gap β/γ), should take into account that larger orders and larger number of detector outputs q potentially lead to better performance.

The **Procedure AFD**, given in what follows, allows the synthesis of least-order fault detection filters to solve the AFDP employing an $\mathcal{H}_{\infty-}/\mathcal{H}_{\infty}$ optimization-based approach. This procedure includes also the **Procedure EFD** in the case when an exact solution exists. Similar synthesis procedures relying on alternative optimization-based formulations (e.g., $\mathcal{H}_{\infty-}/\mathcal{H}_2$, $\mathcal{H}_2-/\mathcal{H}_{\infty}$, $\mathcal{H}_2-/\mathcal{H}_2$, $\mathcal{H}_{\Omega-}/\mathcal{H}_{\infty}$, $\mathcal{H}_{\Omega-}/\mathcal{H}_2$ as well as their finite frequency range counterparts) can be devised by only adapting appropriately the last computational step of the **Procedure AFD**.

Procedure AFD: Approximate synthesis of fault detection filters

Inputs : $\{G_u(\lambda), G_d(\lambda), G_f(\lambda), G_w(\lambda)\}$, q, γ

Outputs: $Q(\lambda), R_f(\lambda), R_w(\lambda), \beta$

- 1) Compute a $(p - r_d) \times (p + m_u)$ minimal proper stable basis $Q_1(\lambda)$ for the left nullspace of $G(\lambda)$ defined in (5.2), where $r_d = \text{rank } G_d(\lambda)$; set $Q(\lambda) = Q_1(\lambda)$ and compute

$$[R_f(\lambda) \ R_w(\lambda)] = Q_1(\lambda) \begin{bmatrix} G_f(\lambda) & G_w(\lambda) \\ 0 & 0 \end{bmatrix}.$$

Exit if exists $j \in \{1, \dots, m_f\}$ such that $R_{f_j}(\lambda) = 0$ (no solution).

- 2) Compute $r_w = \text{rank } R_w(\lambda)$; if $r_w = 0$, set $q_1 = \min(p - r_d, q)$; else, set $q_1 = r_w$; choose a $q_1 \times (p - r_d)$ rational matrix $Q_2(\lambda)$ such that $Q_2(\lambda)Q(\lambda)$ is admissible, has least McMillan degree and, if $r_w > 0$ then $Q_2(\lambda)R_w(\lambda)$ has full row rank r_w ; compute $Q(\lambda) \leftarrow Q_2(\lambda)Q(\lambda)$, $R_f(\lambda) \leftarrow Q_2(\lambda)R_f(\lambda)$ and $R_w(\lambda) \leftarrow Q_2(\lambda)R_w(\lambda)$.
- 3) If $r_w > 0$, compute the quasi-co-outer-co-inner factorization $R_w(\lambda) = R_{wo}(\lambda)R_{wi}(\lambda)$, where the quasi-co-outer factor $R_{wo}(\lambda)$ is invertible and has only stable zeros, excepting possible zeros on the boundary of the stability domain, and $R_{wi}(\lambda)$ is co-inner; with $Q_3(\lambda) = R_{wo}^{-1}(\lambda)$ compute $Q(\lambda) \leftarrow Q_3(\lambda)Q(\lambda)$, $R_f(\lambda) \leftarrow Q_3(\lambda)R_f(\lambda)$ and $R_w(\lambda) \leftarrow R_{wi}(\lambda)$.
- 4) Compute $Q_4(\lambda)$ such that $Q_4(\lambda)Q(\lambda)$ and $Q_4(\lambda)R_f(\lambda)$ are proper and stable; compute $Q(\lambda) \leftarrow Q_4(\lambda)Q(\lambda)$, $R_f(\lambda) \leftarrow Q_4(\lambda)R_f(\lambda)$, and $R_w(\lambda) \leftarrow Q_4(\lambda)R_w(\lambda)$.
- 5) If $r_w > 0$, choose $Q_5 \in \mathbb{R}^{\min(q, r_w) \times q_1}$ such that $\|Q_5Q_4(\lambda)\|_{\infty} = \gamma$ and $\beta = \|Q_5R_f(\lambda)\|_{\infty-} > 0$; update $Q(\lambda) \leftarrow Q_5Q(\lambda)$, $R_f(\lambda) \leftarrow Q_5R_f(\lambda)$, and $R_w(\lambda) \leftarrow Q_5R_w(\lambda)$; else, set $\beta = \infty$.

Remark 5.9 The threshold selection approach of Sect. 3.6 can be applied to determine a threshold value τ which guarantees the lack of false alarms. For any selected value of the threshold τ , we can estimate for $j = 1, \dots, m_f$ the magnitude δ_{f_j} , of the minimum size detectable fault $f_j \neq 0$, provided $f_k = 0 \forall k \neq j$. Consider the internal representation of the resulting fault detection filter in the form

$$\mathbf{r}(\lambda) = \sum_{j=1}^{m_f} R_{f_j}(\lambda) \mathbf{f}_j(\lambda) + R_w(\lambda) \mathbf{w}(\lambda).$$

By using (3.39) in the frequency domain (via Plancherel's theorem), δ_{f_j} can be computed from

$$2\tau = \inf_{\|\mathbf{f}_j\|=\delta_{f_j}} \|\mathbf{R}_{f_j}(\lambda) \mathbf{f}_j(\lambda)\|_2 = \delta_{f_j} \|R_{f_j}(\lambda)\|_{\Omega^-},$$

where we used the properties of the index defined in (5.25). For $w(t)$ having bounded energy and satisfying $\|w\|_2 \leq \delta_w$, we obtain

$$\delta_{f_j} = \frac{2\|R_w(\lambda)\|_{\infty} \delta_w}{\|R_{f_j}(\lambda)\|_{\Omega^-}}. \quad (5.33)$$

The resulting value of δ_{f_j} can be used to assess the ‘‘practical usefulness’’ of any solution. A small value of $\|R_{f_j}(\lambda)\|_{\Omega^-}$ may indicate a large size of the minimal detectable faults for a particular choice of Ω . Therefore, various alternative choices of Ω may be used to arrive to more realistic estimates. For example, Ω can be defined as a relevant interval of frequency values, or only a finite set of relevant frequencies (e.g., the DC-gain frequency λ_s). \square

Example 5.6 If we apply **Procedure AFD** to solve the $\mathcal{H}_{\infty-}/\mathcal{H}_{\infty}$ synthesis problem for the system in Example 5.5, the resulting optimization problem is nonstandard, because $G_w(s)$ has a zero at infinity. Let choose $\gamma = 1$. At Step 1) we set $Q_1(s) = N_I(s)$, with $N_I(s)$ determined in Example 5.5. We have that $R_w(s) = G_w(s)$ and $R_f(s) = G_f(s)$. Since each column of $R_f(s)$ is nonzero, the AFDP is solvable. Since $r_w = 1$, at Step 2), we can employ a constant vector $Q_2(\lambda) = [1 \ 1]$ to obtain the updated quantities

$$Q(s) = \begin{bmatrix} 1 & 1 \\ 1 & 1 - \frac{2s^2 + 8s + 7}{(s+2)(s+3)} \end{bmatrix}, \quad R_w(s) = \frac{1}{s+2}, \quad R_f(s) = \begin{bmatrix} \frac{s+1}{s+2} & 1 \end{bmatrix}. \quad (5.34)$$

At Step 3), the quasi-outer factor $G_{w\theta}(s)$ is simply $R_w(s)$ and, being strictly proper, has thus a zero at infinity. With $Q_3(s) = R_w^{-1}(s)$, the resulting $Q(s)$ and $R_f(s)$ are therefore improper. At Step 4), we choose $Q_4(s)$ of unity $\mathcal{H}_{\infty-}$ -norm of the form $Q_4(s) = a/(s+a)$ with $a \geq 2$. For $\gamma = 1$ we obtain at Step 5) with $Q_5 = 1$ the final $Q(s)$, $R_f(s)$, and $R_w(s)$

$$Q(s) = \begin{bmatrix} a \frac{s+2}{s+a} & a \frac{s+2}{s+a} & -a \frac{2s^2 + 8s + 7}{(s+a)(s+3)} \end{bmatrix}, \quad R_f(s) = \begin{bmatrix} a \frac{s+1}{s+a} & a \frac{s+2}{s+a} \end{bmatrix}, \quad R_w(s) = \frac{a}{s+a}.$$

Since $\beta = \|R_f(s)\|_{\infty-} = a$, it follows that β can be arbitrarily large, and thus the $\mathcal{H}_{\infty-}/\mathcal{H}_{\infty}$ problem (5.26) has no optimal solution. Although not optimal, the resulting fault detection filter can be reliably employed for detecting faults, whose minimum amplitude is above a certain threshold. The value of this threshold can be easily determined using information on the size and waveform of the noise input.

The script **Ex5_6** in Listing 5.2 solves the AFDP considered in this example. \diamond

Listing 5.2 Script **Ex5_6** to solve the AFDP of Example 5.6 using Procedure AFD

```

% Uses the Control Toolbox and the Descriptor System Tools

% define s as an improper transfer function
s = tf('s');
% define Gu(s), Gw(s), Gf(s)
Gu = [(s+1)/(s+2); (s+2)/(s+3)]; % enter Gu(s)
Gw = [1/(s+2); 0]; % enter Gw(s)
Gf = [(s+1)/(s+2) 0; 0 1]; % enter Gf(s)
p = 2; mu = 1; mw = 1; mf = 2; % enter dimensions

tol = 1.e-7; % set tolerance for rank tests

% choose the left nullspace as Q=[I - Gu] and
% initialize Rf = Q[Gf; 0] = Gf and Rw = Q[Gw; 0] = Gw
Q = ss([eye(p) -Gu]); Rf = ss(Gf); Rw = ss(Gw);

% compress Rw to a full row rank matrix
rw = rank(evalfr(Rw,rand)); nb = size(Q,1);
if rw < nb
    h = ones(rw,nb);
    % may use alternatively h = rand(rw,nb);
    Q = h*Q; Rf = h*Rf; Rw = h*Rw;
end

% compute the quasi-co-outer-co-inner factorization Rw = RwoRwi
[Rwi,Rwo] = goifac(Rw,tol);

% compute optimal filter (for standard case)
Q = gir(Rwo\Q,tol); % update Q ← Rwo-1Q
Rf = gir(Rwo\Rf,tol); Rw = Rwi; % update Rf ← Rwo-1Rf and Rw ← Rwi

% check for poles on the extended imaginary axis
poles = gpole([Q Rf]);
if max(real(poles)) == inf || min(abs(real(poles))) < 0.0001
    % compute a stable and proper left coprime factorization
    % of [Q Rf Rw]; enforce stability degree -3
    opts = struct('sdeg',-3,'smarg',-3);
    [Q_Rf_Rw,M] = glcf(gir([Q,Rf,Rw],tol),opts);
    % adjust denominator to unit infinity norm to match example
    scale = -norm(M,inf);
    Q = minreal(tf(Q_Rf_Rw(:,1:p+mu)/scale));
    Rf = minreal(tf(Q_Rf_Rw(:,p+mu+1:p+mu+mf)/scale));
    Rw = minreal(tf(Q_Rf_Rw(:,p+mu+mf+1:end)/scale));
end

```

Example 5.7 We solve the problem in Example 5.6 using the alternative approach suggested in Remark 5.8. At Steps 1) and 2) we determine the same $Q(s)$, $R_w(s)$ and $R_f(s)$ as in (5.34). The quasi-outer factor is as before $G_{wo}(s) = R_w(s)$ and is strictly proper, having thus a zero at infinity. For $\varepsilon > 0$, we determine $G_{wo,\varepsilon}(s)$ such that $G_{wo,\varepsilon}(s)G_{wo,\varepsilon}^{\sim}(s) = \varepsilon^2 + G_{wo}(s)G_{wo}^{\sim}(s)$ and we obtain

$$G_{wo,\varepsilon}(s) = \frac{\varepsilon s + \sqrt{1 + 2\varepsilon^2}}{s + 2}.$$

With $Q_3(s) = G_{wo,\varepsilon}^{-1}(s)$, the optimal solution of the problem (5.32) is $\bar{Q}_3(s) = 1$ for which the final $Q(s)$, $R_f(s)$ and $R_w(s)$ are

$$Q(s) = \left[\frac{s+2}{\varepsilon s + \sqrt{1+2\varepsilon^2}} \quad \frac{s+2}{\varepsilon s + \sqrt{1+2\varepsilon^2}} - \frac{2s^2+8s+7}{(\varepsilon s + \sqrt{1+2\varepsilon^2})(s+3)} \right],$$

$$R_f(s) = \left[\frac{s+1}{\varepsilon s + \sqrt{1+2\varepsilon^2}} \quad \frac{s+2}{\varepsilon s + \sqrt{1+2\varepsilon^2}} \right], \quad R_w(s) = \frac{1}{\varepsilon s + \sqrt{1+2\varepsilon^2}}.$$

Since $\beta = \|R_f(s)\|_{\infty-} = 1/\varepsilon$, it follows that β becomes arbitrarily large as $\varepsilon \rightarrow 0$. Although the $\mathcal{H}_{\infty-}/\mathcal{H}_{\infty}$ problem (5.26) has no optimal solution, the resulting filter $Q(s)$ can be acceptable for a large range of values of ε . \diamond

Example 5.8 If we solve the $\mathcal{H}_{\infty}/\mathcal{H}_{\infty-}$ synthesis problem for Example 5.5, the optimal solution $Q(s)$ and the corresponding $R_f(s)$ are simply

$$Q(s) = G_f^{-1}(s)N_l(s) = \begin{bmatrix} \frac{s+2}{s+1} & 0 & -1 \\ 0 & 1 & -\frac{s+2}{s+3} \end{bmatrix}, \quad R_f(s) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix},$$

which lead to the optimal values $\beta = 1$ and $\gamma = 1$. In contrast to the filter in Example 5.7, this filter is optimal (in a certain sense) and able to perform fault isolation as well, by exactly reconstructing fault f_2 and approximately fault f_1 . \diamond

Remark 5.10 The solution of the AFDP can be refined in the case when $r_w < p - r_d$. In this case, it follows that there exists a left nullspace basis $\bar{N}_{l,w}(\lambda)$ such that $\bar{N}_{l,w}(\lambda)\bar{G}_w(\lambda) = 0$, thus the noise input can be exactly decoupled. Also, there exists a maximal subvector $f^{(1)}$ of fault inputs which are completely fault detectable (i.e., the columns of the corresponding $\bar{N}_{l,w}(\lambda)\bar{G}_{f^{(1)}}(\lambda)$ are nonzero), while none of the components of its complementary part $f^{(2)}$ of f is fault detectable (i.e., all columns of the corresponding $\bar{N}_{l,w}(\lambda)\bar{G}_{f^{(2)}}(\lambda)$ are zero), and thus are completely decoupled. Here, we denoted with $\bar{G}_{f^{(1)}}(\lambda)$ and $\bar{G}_{f^{(2)}}(\lambda)$ the columns of $\bar{G}_f(\lambda)$ corresponding to $f^{(1)}$ and $f^{(2)}$, respectively. This allows the partitioning of the reduced system (5.11) as

$$\bar{\mathbf{y}}(\lambda) := \bar{G}_{f^{(1)}}(\lambda)\mathbf{f}^{(1)}(\lambda) + \bar{G}_{f^{(2)}}(\lambda)\mathbf{f}^{(2)}(\lambda) + \bar{G}_w(\lambda)\mathbf{w}(\lambda). \quad (5.35)$$

In general, we can construct $\bar{Q}_1(\lambda)$ and $Q(\lambda)$ in the forms

$$\bar{Q}_1(\lambda) = \begin{bmatrix} \bar{Q}_1^{(1)}(\lambda) \\ \bar{Q}_1^{(2)}(\lambda) \end{bmatrix}, \quad Q(\lambda) = \begin{bmatrix} Q^{(1)}(\lambda) \\ Q^{(2)}(\lambda) \end{bmatrix} := \begin{bmatrix} \bar{Q}_1^{(1)}(\lambda) \\ \bar{Q}_1^{(2)}(\lambda) \end{bmatrix} Q_1(\lambda), \quad (5.36)$$

where $\bar{Q}_1^{(1)}(\lambda)$ solves the EFDP for the reduced system (5.35) with respect to fault components $f^{(1)}$ and decouples $f^{(2)}$ and w in the leading components $r^{(1)}$ of the residual r , while $\bar{Q}_1^{(2)}(\lambda)$ solves the AFDP for the reduced system (5.35) for the fault components $f^{(2)}$ and generates the trailing components $r^{(2)}$ of the residual r . The maximum number of components of $r^{(1)}$ is $p - r_d - r_w$, while $r^{(2)}$ will have maximum r_w components. Thus, the number of components of r is limited to $p - r_d$. The case $f = f^{(1)}$ corresponds to the solution of an EFDP for which **Procedure EFD** can be used, while the case $f = f^{(2)}$ corresponds to the solution of an AFDP, for which **Procedure AFD** can be used. \square

Example 5.9 Consider once again the solution of the $\mathcal{H}_{\infty-}/\mathcal{H}_{\infty}$ synthesis problem for Example 5.5. With $N_l(s)$ chosen as in Example 5.5, we have that the rank of $\overline{G}_w(s)$ (or $R_w(s)$ at Step 1) of **Procedure AFD** is $r_w = 1$. With $\overline{N}_{l,w}(s) = [0 \ 1]$, we obtain $\overline{N}_{l,w}(s)\overline{G}_w(s) = 0$ and $\overline{N}_{l,w}(s)\overline{G}_f(s) = [0 \ 1]$. Thus, with $f^{(1)} = f_2, f^{(2)} = f_1$ and

$$\overline{G}_{f^{(1)}}(s) := \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \quad \overline{G}_{f^{(2)}}(s) := \begin{bmatrix} \frac{s+1}{s+2} \\ 0 \end{bmatrix},$$

we arrive to the partitioned subsystem (5.35). We determine $Q(s)$ in the partitioned form (5.36), where the solution of the EFDP with the above $\overline{G}_{f^{(2)}}(s)$ is simply

$$Q^{(1)}(s) = \overline{N}_{l,w}(s)N_l(s) = \begin{bmatrix} 0 & 1 & -\frac{s+2}{s+3} \end{bmatrix}.$$

We determine $Q^{(2)}(s)$ by solving the AFDP formulated with $\overline{G}_{f^{(1)}}(s)$ and $\overline{G}_w(s)$ using **Procedure AFD**. With $Q_1(s) = N_l(s)$ chosen in Example 5.5 and $Q_2(s) = [1 \ 0]$ we obtain at Step 2)

$$Q^{(2)}(s) = \begin{bmatrix} 1 & 0 & -\frac{s+1}{s+2} \end{bmatrix}, \quad R_w(s) = \frac{1}{s+2}, \quad R_{f^{(2)}}(s) = \frac{s+1}{s+2}.$$

With $Q_3(s) = R_w^{-1}(s)$ at Step 3), $Q_4(s) = Q_3^{-1}(s) = R_w(s)$ at Step 4), and $Q_5 = 2$ we obtain at Step 5) for $\gamma = 1$ the final $Q^{(2)}(s)$ and corresponding $R_{f^{(2)}}(s)$

$$Q^{(2)}(s) = \begin{bmatrix} 2 & 0 & -2\frac{s+1}{s+2} \end{bmatrix}, \quad R_{f^{(2)}}(s) = 2\frac{s+1}{s+2},$$

for which $\beta = \|R_{f^{(2)}}(s)\|_{\infty-} = 2$. The combined solutions according to (5.36) give

$$Q(s) = \begin{bmatrix} Q^{(1)}(s) \\ Q^{(2)}(s) \end{bmatrix} = \begin{bmatrix} 0 & 1 & -\frac{s+2}{s+3} \\ 2 & 0 & -2\frac{s+1}{s+2} \end{bmatrix}, \quad R_f(s) = \begin{bmatrix} 0 & 1 \\ 2\frac{s+1}{s+2} & 0 \end{bmatrix}.$$

The resulting filter is able to perform fault isolation as well, and even the exact reconstruction of the fault f_2 . The optimal value $\beta = 1$ for $\gamma = 1$ is the same as for the ‘‘optimal’’ solution of Example 5.7. However, since the exact solution $Q^{(1)}(s)$ can be arbitrarily scaled, the effective value of β is 2, which is larger than for the ‘‘optimal’’ solution of Example 5.8. \diamond

5.4 Solving the Exact Fault Detection and Isolation Problem

Let S be a given $n_b \times m_f$ structure matrix to be achieved by the fault detection filter $Q(\lambda)$. Using the factorized representation $Q(\lambda) = \overline{Q}_1(\lambda)Q_1(\lambda)$ in (5.9), it follows that, to solve the *exact fault detection and isolation problem* (EFDIP) formulated in Sect. 3.5.3 for the system (3.2) with $w \equiv 0$, the same S must be achieved by $\overline{Q}_1(\lambda)$ for the reduced system (5.11) for $w \equiv 0$. For this, we consider $\overline{Q}_1(\lambda)$ partitioned with n_b block rows, in the form

$$\bar{Q}_1(\lambda) = \begin{bmatrix} \bar{Q}_1^{(1)}(\lambda) \\ \bar{Q}_1^{(2)}(\lambda) \\ \vdots \\ \bar{Q}_1^{(n_b)}(\lambda) \end{bmatrix}, \quad (5.37)$$

where the i -th block row $\bar{Q}_1^{(i)}(\lambda)$ generates the i -th component of the residual vector

$$\mathbf{r}^{(i)}(\lambda) := \bar{Q}_1^{(i)}(\lambda)\bar{\mathbf{y}}(\lambda) \quad (5.38)$$

and achieves the i -th specification contained in the i -th row of S .

The solvability conditions of the EFDIP given in Theorem 3.10 (also explicitly given in Theorem 3.5) can be replaced by simpler conditions for the reduced system (5.11). This comes down to checking for $i = 1, \dots, n_b$, the solvability conditions for the i -th specification contained in the i -th row of S . For this purpose, we rewrite for each $i, i = 1, \dots, n_b$, the reduced system (5.11) for $w \equiv 0$ as

$$\bar{\mathbf{y}}(\lambda) = \bar{G}_d^{(i)}(\lambda)\mathbf{d}^{(i)}(\lambda) + \bar{G}_f^{(i)}(\lambda)\mathbf{f}^{(i)}(\lambda), \quad (5.39)$$

where $d^{(i)}$ contains those components f_j of f for which $S_{ij} = 0$, $f^{(i)}$ contains those components f_j of f for which $S_{ij} \neq 0$, while $\bar{G}_d^{(i)}(\lambda)$ and $\bar{G}_f^{(i)}(\lambda)$ are formed from the corresponding sets of columns of $\bar{G}_f(\lambda)$, respectively. Thus, $d^{(i)}$ contains all fault components to be decoupled in the i -th component $r^{(i)}$ of the residual by the i -th filter $\bar{Q}_1^{(i)}(\lambda)$, while $f^{(i)}$ contains those faults which need to be detected in the i -th component $r^{(i)}$ of the residual.

The following corollary to Theorem 3.10 provides the solvability conditions of the EFDIP in terms of the n_b reduced systems formed in (5.39):

Corollary 5.5 *For the system (3.2) with $w \equiv 0$ and a given structure matrix S , the EFDIP is solvable if and only if the system (5.11) with $w \equiv 0$ is S -fault isolable, or equivalently, for $i = 1, \dots, n_b$*

$$\text{rank} [\bar{G}_d^{(i)}(\lambda) \bar{G}_f^{(i)}(\lambda)] > \text{rank} \bar{G}_d^{(i)}(\lambda), \quad \forall j, \quad S_{ij} \neq 0,$$

where $\bar{G}_d^{(i)}(\lambda)$ is formed from the columns $\bar{G}_f^{(i)}(\lambda)$ of $\bar{G}_f(\lambda)$ for which $S_{ij} = 0$.

In other words, to check the fault isolability for the i -th specification, we have simply to check the complete fault detectability of the corresponding reduced system (5.39) with permuted inputs.

A similar corollary to Theorem 3.11 provides the solvability condition for the solution of the EFDIP with strong isolability.

Corollary 5.6 For the system (3.2) with $w \equiv 0$ and $S = I_{m_f}$, the EFDIP is solvable if and only if the system (5.11) with $w \equiv 0$ is strongly fault isolable, or equivalently

$$\text{rank } \overline{G}_f(\lambda) = m_f .$$

To determine the i -th block row $\overline{Q}_1^{(i)}(\lambda)$ of $\overline{Q}_1(\lambda)$ in (5.37), we have to solve an EFDIP for the corresponding reduced system (5.39). For this purpose, the **Procedure EFD** can be applied, which also checks the solvability conditions for the corresponding specification. The resulting overall detector $Q(\lambda)$ and the corresponding $R_f(\lambda)$ are

$$Q(\lambda) = \begin{bmatrix} Q^{(1)}(\lambda) \\ Q^{(2)}(\lambda) \\ \vdots \\ Q^{(n_b)}(\lambda) \end{bmatrix} = \begin{bmatrix} \overline{Q}_1^{(1)}(\lambda) \\ \overline{Q}_1^{(2)}(\lambda) \\ \vdots \\ \overline{Q}_1^{(n_b)}(\lambda) \end{bmatrix} Q_1(\lambda), \quad R_f(\lambda) = \begin{bmatrix} R_f^{(1)}(\lambda) \\ R_f^{(2)}(\lambda) \\ \vdots \\ R_f^{(n_b)}(\lambda) \end{bmatrix}, \quad (5.40)$$

where the i -th block row $R_f^{(i)}(\lambda)$ achieves the i -th specification contained in the i -th row of S .

The **Procedure EFDI**, given below, determines the n_b row blocks $Q^{(i)}(\lambda)$ and $R_f^{(i)}(\lambda)$, $i = 1, \dots, n_b$, of $Q(\lambda)$ and $R_f(\lambda)$, respectively, with the i -th blocks having the desired row dimension q_i .

Procedure EFDI: Exact synthesis of fault detection and isolation filters

Inputs : $\{G_u(\lambda), G_d(\lambda), G_f(\lambda)\}, S \in \mathbb{R}^{n_b \times m_f}, \{q_1, \dots, q_{n_b}\}$

Outputs: $Q^{(i)}(\lambda), R_f^{(i)}(\lambda), i = 1, \dots, n_b$

- 1) Compute a $(p - r_d) \times (p + m_u)$ minimal basis matrix $Q_1(\lambda)$ for the left nullspace of $G(\lambda)$ defined in (5.2), where $r_d := \text{rank } G_d(\lambda)$;
 set $Q(\lambda) = Q_1(\lambda)$ and compute $R_f(\lambda) = Q_1(\lambda) \begin{bmatrix} G_f(\lambda) \\ 0 \end{bmatrix}$.
- 2) For $i = 1, \dots, n_b$
 - 2.1) Define $\overline{G}_d^{(i)}(\lambda)$ as those columns $R_{f_j}(\lambda)$ of $R_f(\lambda)$ for which $S_{ij} = 0$ and $\overline{G}_f^{(i)}(\lambda)$ as those columns $R_{f_j}(\lambda)$ for which $S_{ij} \neq 0$.
 - 2.2) Apply the **Procedure EFD** to the system described by the quadruple $\{0, \overline{G}_d^{(i)}(\lambda), \overline{G}_f^{(i)}(\lambda), 0\}$ to obtain the $q_i \times (p - r_d)$ least-order filter $\overline{Q}_1^{(i)}(\lambda)$. **Exit** if no solution exists.
 - 2.3) Compute $Q^{(i)}(\lambda) = \overline{Q}_1^{(i)}(\lambda)Q(\lambda)$ and $R_f^{(i)}(\lambda) = \overline{Q}_1^{(i)}(\lambda)R_f(\lambda)$.

This synthesis procedure ensures that each block $\overline{Q}_1^{(i)}(\lambda)$ and the corresponding $R_f^{(i)}(\lambda)$ are stable. Thus the overall $R_f(\lambda)$ in (5.40) is also stable. The stability of overall $Q(\lambda)$ in (5.40) can be always ensured, by choosing a stable left nullspace

basis $Q_1(\lambda)$ at Step 1). As it will be shown in Sect. 7.4, this is not necessary, because the computation of both $Q^{(i)}(\lambda) = \overline{Q}_1^{(i)}(\lambda)Q(\lambda)$ and $R_f^{(i)}(\lambda) = \overline{Q}_1^{(i)}(\lambda)R_f(\lambda)$ at Step 2.3) can be done by using state-space representation based updating techniques, which always guarantee that $Q^{(i)}(\lambda)$ and $R_f^{(i)}(\lambda)$ have the state-space representations with the same state and descriptor matrices, and result simultaneously stable.

The applicability of **Procedure EFDI** for a given system relies on the assumption that the structure matrix S is achievable. Therefore, to select a minimal set of specifications which cover all expected fault combinations, it is important to know all achievable specifications for a given system. For a system with m_f faults, the complete set of possible distinct specifications contains $2^{m_f} - 1$ elements. Thus, a brute force approach is based on an exhaustive search, by trying to solve the EFDIP for each of these specifications to find out which specifications are feasible (i.e., the corresponding design was successful). The main problem with this approach is its lack of efficiency, as explained in what follows.

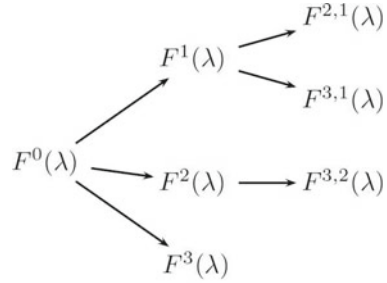
Each synthesis problem of a fault detection filter for a given specification can be reformulated as a standard EFDP, where all faults with zero signatures in the checked specification are redefined as disturbances. With this reformulation, the main computation is the determination of the nullspace basis of a TFM with $p + m_u$ rows and $m_u + m_d + k$ columns, where k denotes the number of null elements in the tested specification (i.e., $0 \leq k < m_f$) and represents the number of additional disturbance inputs which results by recasting the fault inputs to be decoupled as disturbances. The nullspace computation must be performed for *all* $2^{m_f} - 1$ possible specifications, although this may not be necessary if $m_f > p - r_d$, where we recall that r_d is the rank of $G_d(\lambda)$. In what follows, we describe a more efficient approach, where the product representation of nullspace, mentioned in Sect. 5.1, is systematically exploited. The expected efficiency gain arises by replacing the above nullspace computations on matrices with $p + m_u$ rows and at least $m_u + m_d$ columns, with a succession of nullspace determinations on single column matrices with decreasing number of rows. This leads to a significant reduction of the total computational burden.

We now describe a recursive procedure to generate in a systematic and computationally efficient way suitable nullspace bases to serve for the determination of all achievable specifications. We illustrate the core computation with two generic $p_e \times m$ and $p_e \times m_f$ TFMs $G(\lambda)$ and $F(\lambda)$, respectively. The basic computational step consists of successively determining left nullspace bases $N_l(\lambda)$ of $G(\lambda)$ (i.e., $N_l(\lambda)G(\lambda) = 0$) such that the structure matrix of $N_l(\lambda)F(\lambda)$ has up to $\min(m_f, p_e - r) - 1$ zero columns, where $r = \text{rank } G(\lambda)$. To initialize the procedure for the system (2.1), we initialize these TFMs as

$$G(\lambda) = \begin{bmatrix} G_u(\lambda) & G_d(\lambda) \\ I_{m_u} & 0 \end{bmatrix}, \quad F(\lambda) = \begin{bmatrix} G_f(\lambda) \\ 0 \end{bmatrix}, \quad (5.41)$$

with $p_e = p + m_u$ and $m = m_u + m_d$.

Fig. 5.1 Tree of performed computations of fault specifications



To describe the nullspace generation process in more details, let $N_l^0(\lambda)$ be the $(p_e - r) \times p_e$ proper minimal left nullspace basis of $G(\lambda)$ and let S_{F^0} be the structure matrix of $F^0(\lambda) := N_l^0(\lambda)F(\lambda)$. This structure matrix is a $1 \times m_f$ row vector corresponding to $F^0(\lambda)$ seen as an $1 \times m_f$ block row (see the definition of structure matrix in (3.17) based on (3.16)), with the $(1, j)$ -th block row element formed of the j -th column of $F^0(\lambda)$. If $\min(m_f, p_e - r) > 1$, then for each $i = 1, \dots, m_f$, determine the left nullspace basis $N_l^i(\lambda)$ of the i -th column of $F^0(\lambda)$ and let S_{F^i} be the structure matrix corresponding to $F^i(\lambda) := N_l^i(\lambda)F^0(\lambda)$. Each S_{F^i} is a $1 \times m_f$ row vector with the i -th column element equal to zero. If the i -th column is zeroed with $N_l^i(\lambda)$, then $N_l^i(\lambda)$ is a $(p_e - r - 1) \times (p_e - r)$ TFM. If now $p_e - r - 1 > 1$, we continue by computing for each j -th column of $F^i(\lambda)$, $j > i$, the corresponding left nullspace $N_l^{j,i}(\lambda)$ and the corresponding structure matrix $S_{F^{j,i}}$ of $F^{j,i}(\lambda) := N_l^{j,i}(\lambda)F^i(\lambda)$. Each $S_{F^{j,i}}$ will have zeros in its i -th and j -th columns. This process continues in a similar way until all nonzero $S_{F^{k,\dots,j,i}}$ have been generated. The resulting S is formed by concatenating row-wise the determined $S_{F^0}, S_{F^1}, \dots, S_{F^{m_f}}, S_{F^{2,1}}, \dots, S_{F^{m_f-1}}, \dots, S_{F^{m_f, m_f-1}}, \dots$. The tree in Fig. 5.1 illustrates the performed computations for a system with $m_f = 3$ and $p_e - r = 3$.

If we denote with S the matrix formed of all achievable specifications, then, for the considered example, we have $S = [S_{F^0}^T \ S_{F^1}^T \ S_{F^{2,1}}^T \ S_{F^{3,1}}^T \ S_{F^2}^T \ S_{F^{3,2}}^T \ S_{F^3}^T]^T$, where each S_{F^i} has the i -th column zero, while each $S_{F^{j,i}}$ has the i -th and j -th columns zero. Note that in nongeneric cases, other elements may also be zero. It can be observed that the computation of $F^{1,2}(\lambda)$ is not necessary because the same information is provided by $F^{2,1}(\lambda)$. Similarly, the computation of both $F^{1,3}(\lambda)$ and $F^{2,3}(\lambda)$ is not necessary, because the corresponding information is provided by $F^{3,1}(\lambda)$ and $F^{3,2}(\lambda)$, respectively.

The computational process can be easily formulated as a recursive procedure, which for the given matrices $G(\lambda)$ and $F(\lambda)$, computes the maximally achievable structure matrix S . This procedure can be formally called as $S = \text{GENSPEC}(G, F)$.

For example, the maximally achievable structure matrix for the system (2.1) can be computed with $G(\lambda)$ and $F(\lambda)$ defined in (5.41).

Procedure GENSPEC: Generation of achievable fault specifications

Inputs : $G(\lambda) \in \mathbb{R}(\lambda)^{p_e \times m}$, $F(\lambda) \in \mathbb{R}(\lambda)^{p_e \times m_f}$

Output: $S \in \mathbb{R}^{q \times m_f}$

Function $S = \text{GENSPEC}(G, F)$

- 1) Compute a left nullspace basis $N_l(\lambda)$ of $G(\lambda)$.
Exit with $S = \emptyset$ if $N_l(\lambda)$ is empty.
- 2) Compute $N_f(\lambda) = N_l(\lambda)F(\lambda)$.
- 3) Compute the structure matrix S of $N_f(\lambda)$. **Exit** if $N_f(\lambda)$ is a row vector.
- 4) **For** $i = 1, \dots, m_f$
 - 4.1) Form $\tilde{G}_i(\lambda)$ as column i of $N_f(\lambda)$.
 - 4.2) Form $\tilde{F}_i(\lambda)$ from columns $1, \dots, i-1, i+1, \dots, m_f$ of $N_f(\lambda)$.
 - 4.3) **Call** $\tilde{S} = \text{GENSPEC}(\tilde{G}_i, \tilde{F}_i)$.
 - 4.4) Partition $\tilde{S} = [\tilde{S}_1 \ \tilde{S}_2]$ such that \tilde{S}_1 has $i-1$ columns.
 - 4.5) Define $\hat{S} = [\tilde{S}_1 \ 0 \ \tilde{S}_2]$ and update $S \leftarrow \begin{bmatrix} S \\ \hat{S} \end{bmatrix}$.

The **Procedure GENSPEC** performs the minimum number of nullspace computations and updating. This number is given by $k_S = \sum_{i=0}^{i_{\max}} \binom{m_f}{i}$, where $i_{\max} = \min(m_f, p_e - r) - 1$ and r is the rank of the initial $G(\lambda)$. As it can be observed, k_S depends on the number of initial basis vectors $p_e - r$ and the number of faults m_f , and, although the number of distinct specifications can be relatively low, still k_S can be a large number. For the example considered above, $m_f = 3$ and $p_e - r = 3$, thus the maximum number $k_S = 7 (= 2^{m_f} - 1)$ nullspace computations are necessary. However, in contrast to the brute force approach, all but one of nullspace computations are performed for rational matrices with a single column (and varying number of rows), and therefore a substantial saving in the computation effort can be expected.

Example 5.10 Consider a continuous-time system with triplex sensor redundancy on one of its measured output components, which we denote by y_1, y_2 and y_3 . Each output is related to the control and disturbance inputs by the input–output relation

$$\mathbf{y}_i(s) = G_u(s)\mathbf{u}(s) + G_d(s)\mathbf{d}(s), \quad i = 1, 2, 3,$$

where $G_u(s)$ and $G_d(s)$ are $1 \times m_u$ and $1 \times m_d$ TFMs, respectively. We assume all three outputs are susceptible to additive sensor faults. Thus, the input–output model of the system has the form

$$\mathbf{y}(s) := \begin{bmatrix} \mathbf{y}_1(s) \\ \mathbf{y}_2(s) \\ \mathbf{y}_3(s) \end{bmatrix} = \begin{bmatrix} G_u(s) \\ G_u(s) \\ G_u(s) \end{bmatrix} \mathbf{u}(s) + \begin{bmatrix} G_d(s) \\ G_d(s) \\ G_d(s) \end{bmatrix} \mathbf{d}(s) + \begin{bmatrix} \mathbf{f}_1(s) \\ \mathbf{f}_2(s) \\ \mathbf{f}_3(s) \end{bmatrix}.$$

The maximal achievable structure matrix obtained by applying the **Procedure GENSPEC** is

$$S_{max} = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{bmatrix}.$$

If we can assume that no simultaneous sensor failures occur, then we can target to solve a EFDIP for the structure matrix

$$S = \begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{bmatrix},$$

where the columns of S codify the desired fault signatures.

By using the **Procedure EFDI**, we compute first a left nullspace basis $N_l(s)$ of

$$G(s) = \begin{bmatrix} G_u(s) & G_d(s) \\ G_u(s) & G_d(s) \\ G_u(s) & G_d(s) \\ 1 & 0 \end{bmatrix},$$

in a product form similar to (5.5). We obtain

$$N_l(s) = \begin{bmatrix} 1 & -1 & 0 \\ 0 & 1 & -1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & -G_u(s) \\ 0 & 1 & 0 & -G_u(s) \\ 0 & 0 & 1 & -G_u(s) \end{bmatrix} = \begin{bmatrix} 1 & -1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & -1 & 0 & \cdots & 0 \end{bmatrix}. \quad (5.42)$$

We set $Q_1(s) = N_l(s)$ and

$$R_f(s) = \begin{bmatrix} 1 & -1 & 0 \\ 0 & 1 & -1 \end{bmatrix}. \quad (5.43)$$

For example, to achieve the first specification $[0 \ 1 \ 1]$, we redefine f_1 as a disturbance $d^{(1)} := f_1$ to be decoupled, $f^{(1)} := [f_2 \ f_3]^T$, $\overline{G}_d^{(1)}(s)$ as the first column of $R_f(s)$ and $\overline{G}_f^{(1)}(s)$ as the last two columns of $R_f(s)$. With **Procedure EFD** we obtain $\overline{Q}_1^{(1)}(s) = [0 \ 1]$ (as a constant basis of the left nullspace of $\overline{G}_d^{(1)}(s)$). Thus, the first row of the overall filter $Q(s)$ is given by

$$Q^{(1)}(s) = \overline{Q}_1^{(1)}(s)Q_1(s) = [0 \ 1 \ -1 \ 0 \ \cdots \ 0].$$

The corresponding residual component is simply

$$r_1 = y_2 - y_3 = f_2 - f_3,$$

which is fully decoupled from f_1 . Similarly, with $\overline{Q}_1^{(2)}(s) = [-1 \ -1]$ and $\overline{Q}_1^{(3)}(s) = [1 \ 0]$ we obtain

$$Q^{(2)}(s) = \overline{Q}_1^{(2)}(s)Q_1(s) = [-1 \ 0 \ 1 \ 0 \ \cdots \ 0]$$

and

$$Q^{(3)}(s) = \overline{Q}_1^{(3)}(s)Q_1(s) = [1 \ -1 \ 0 \ 0 \ \cdots \ 0].$$

The TFM of the overall FDI filter is

$$Q(s) = \begin{bmatrix} Q^{(1)}(s) \\ Q^{(2)}(s) \\ Q^{(3)}(s) \end{bmatrix} = \begin{bmatrix} 0 & 1 & -1 & 0 & \cdots & 0 \\ -1 & 0 & 1 & 0 & \cdots & 0 \\ 1 & -1 & 0 & 0 & \cdots & 0 \end{bmatrix} \quad (5.44)$$

and the overall residual vector is

$$r = \begin{bmatrix} r_1 \\ r_2 \\ r_3 \end{bmatrix} := \begin{bmatrix} y_2 - y_3 \\ y_3 - y_1 \\ y_1 - y_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 & -1 \\ -1 & 0 & 1 \\ 1 & -1 & 0 \end{bmatrix} \begin{bmatrix} f_1 \\ f_2 \\ f_3 \end{bmatrix}.$$

This fault detection filter implements the widely employed voting-based fault isolation scheme for the case when the assumption of a single sensor fault at a time is fulfilled. Its main appeal is its independence of the system dynamics. Thus the constant filter (5.44) can be applied even in the case of a system with nonlinear dynamics. Since parametric variations have no effects on the residuals, a perfect robustness of this scheme is guaranteed. However, for applications to safety critical systems, the voting scheme is potentially unreliable, in the (improbable) case of two simultaneous failures with a common value of faults (e.g., $f_2 = f_3 \neq 0$). In such a case, the faults remain undetected, and often the common fault value is wrongly used as the “valid” measurement.

The script **Ex5_10** in Listing 5.3 solves the EFDIP considered in this example. The script **Ex5_10c** (not listed) is a compact version of this script, which calls the function **efdismyn**, a prototype implementation of **Procedure EFDI**. \diamond

Listing 5.3 Script **Ex5_10** to solve the EFDIP of Example 5.10 using **Procedure EFDI**

```
% Uses the Control Toolbox and the Descriptor System Tools

% enter output and fault vector dimensions
p = 3; mf = 3;
% generate random dimensions for system order and input vectors
nu = floor(1+4*rand); mu = floor(1+4*rand);
nd = floor(1+4*rand); md = floor(1+4*rand);
% define random G_u(s) and G_d(s) with triplex sensor redundancy
% and G_f(s) for triplex sensor faults
Gu = ones(3,1)*rss(nu,1,mu); % enter G_u(s) in state-space form
Gd = ones(3,1)*rss(nd,1,md); % enter G_d(s) in state-space form
Gf = eye(3); % enter G_f(s) for sensor faults
tol = 1.e-7; % tolerance for rank tests

% build model with faults
sysf = [Gu Gd Gf];

% set input groups
sysf.InputGroup.controls = 1:mu; % controls
sysf.InputGroup.disturbances = mu+(1:md); % disturbances
sysf.InputGroup.faults = mu+md+(1:mf); % faults

S = [ 0 1 1; 1 0 1; 1 1 0]; % enter structure matrix

% Procedure EFDI

% Step 1): compute Q_1, the left nullspace of [G_u G_d; I_0]
% R_{f,1} = Q_1 [G_f; 0]; the resulting QRF contains [Q_1 R_{f,1}]
options_glnull=struct('tol',tol,'m2',mf);
QRf = glnull([sysf; eye(mu,mu+md+mf)],options_glnull);

% Step 2): determine the filters Q^{(i)} and corresponding R_f^{(i)}

% initialization
nb = size(S,1); % number of necessary filters n_b
Qt = cell(nb,1); Rft = cell(nb,1);
% options for EFDISYN for the synthesis of scalar output filters
options = struct('tol',tol,'rdim',1);
QRf.InputGroup.aux = 1:p+mu+mf; % indices of [Q_1 R_{f,1}]
for i = 1:nb
    % Step 2.1): Define \bar{G}_{d^{(i)}} as R_{f,1}(:,indd) and \bar{G}_{f^{(i)}} as R_{f,1}(:,indf)
```

```

indd = find(S(i,:) == 0); indf = find(S(i,:) ~= 0);
QRf.InputGroup.disturbances = p+mu+indd; % indices of  $\overline{G}_{d^{(i)}}$ 
QRf.InputGroup.faults = p+mu+indf; % indices of  $\overline{G}_{f^{(i)}}$ 

% Step 2.2): Apply Procedure EFD to  $\{0, \overline{G}_{d^{(i)}}, \overline{G}_{f^{(i)}}\} [Q_1 R_{f,1}]$ 
% to determine a least-order  $Q_1^{(i)}$  such that  $Q_1^{(i)} \overline{G}_{d^{(i)}} = 0$ 
% and the DC-gain of  $Q_1^{(i)} \overline{G}_{f^{(i)}}$  has all columns nonzero;
% the resulting QRfauxi contains:  $[Q_1^{(i)} \overline{G}_{f^{(i)}} Q_1^{(i)} Q_1 Q_1^{(i)} R_{f,1}]$ 
[~, QRfauxi] = efdsyn(QRf, options);
QRfi = QRfauxi(:, 'aux'); % extract  $[Q_1^{(i)} Q_1 Q_1^{(i)} R_{f,1}]$ 
QRfi.InputGroup.aux = []; % clear auxiliary input group

% Step 2.3): extract  $Q^{(i)} = Q_1^{(i)} Q_1$  and  $R_f^{(i)} = Q_1^{(i)} R_{f,1}$ 
Qt{i} = QRfi(:, 1:p+mu);
Rft{i} = QRfi(:, p+mu+(1:mf));
end

% normalize  $Q^{(i)}$  and  $R_f^{(i)}$  to match example
scale = sign([Rft{1}.d(1,2) Rft{2}.d(1,3) Rft{3}.d(1,1)]);
for i = 1:3, Qt{i} = scale(i)*Qt{i}; Rft{i} = scale(i)*Rft{i}; end
Q = [Qt{1};Qt{2};Qt{3}], Rf = [Rft{1};Rft{2};Rft{3}]

```

5.5 Solving the Approximate Fault Detection and Isolation Problem

Let S be a given $n_b \times m_f$ structure matrix targeted to be achieved by the fault detection filter $Q(\lambda)$. Using the factorized representation $Q(\lambda) = \overline{Q}_1(\lambda)Q_1(\lambda)$ in (5.9), it follows that, to solve the *approximate fault detection and isolation problem* (AFDIP) formulated in Sect. 3.5.4, the same S has to be targeted by any $\overline{Q}_1(\lambda)$, which solves the AFDIP for the reduced system (5.11). For this, we consider $\overline{Q}_1(\lambda)$ partitioned with n_b block rows, in the form (5.37), where the i -th block row $\overline{Q}_1^{(i)}(\lambda)$ generates the i -th component $r^{(i)}$ of the residual vector r according to (5.38) and targets to achieve the i -th specification contained in the i -th row of S .

The solvability conditions of the AFDIP given in Theorems 3.12 and 3.13 can be replaced by simpler conditions for the reduced system (5.11). This comes down to checking for $i = 1, \dots, n_b$, the solvability conditions for the i -th specification contained in the i -th row of S . To determine the filter $\overline{Q}_1^{(i)}(\lambda)$, an AFDP can be formulated for each i , by suitably redefining the disturbance, fault and noise inputs of the reduced system (5.11).

The reduced system (5.11) can be rewritten for each $i = 1, \dots, n_b$, in the form

$$\overline{\mathbf{y}}(\lambda) = \overline{G}_d^{(i)}(\lambda)\mathbf{d}^{(i)}(\lambda) + \overline{G}_f^{(i)}(\lambda)\mathbf{f}^{(i)}(\lambda) + \overline{G}_w(\lambda)\mathbf{w}(\lambda), \quad (5.45)$$

where $d^{(i)}$ contains those components f_j of f for which $S_{ij} = 0$, $f^{(i)}$ contains those components f_j of f for which $S_{ij} \neq 0$, while $\overline{G}_d^{(i)}(\lambda)$ and $\overline{G}_f^{(i)}(\lambda)$ are formed from

the corresponding sets of columns of $\overline{G}_f(\lambda)$, respectively. The vector $f^{(i)}$ contains all faults which need to be detected in the i -th component $r^{(i)}$ of the residual.

In the case when the AFDIP is formulated to fulfill the weaker conditions (3.28), $d^{(i)}$ contains all fault components which have to be *approximately* decoupled in the i -th component $r^{(i)}$ of the residual by the i -th filter $\overline{Q}_1^{(i)}(\lambda)$, and therefore, $d^{(i)}$ have to be treated as additional noise inputs. The following corollary to Theorem 3.12 provides the solvability conditions of the AFDIP in terms of the reduced system (5.11) for an arbitrary structure matrix S (see also Remark 3.10):

Corollary 5.7 *For the system (3.2) and a given $n_b \times m_f$ structure matrix S with columns $S_j, j = 1, \dots, m_f$, the AFDIP is solvable with conditions (3.28) if and only if the reduced system (5.11) is fault input observable for all faults f_j corresponding to nonzero columns of S , or equivalently,*

$$\overline{G}_{f_j}(\lambda) \neq 0 \quad \forall j, \quad S_j \neq 0.$$

In the case when the AFDIP is formulated to fulfill the stronger conditions (3.29), $d^{(i)}$ contains all fault components to be *exactly* decoupled in the i -th component $r^{(i)}$ of the residual by the i -th filter $\overline{Q}_1^{(i)}(\lambda)$. The following corollary to Theorem 3.13 provides the solvability conditions of the AFDIP in terms of the reduced system (5.11):

Corollary 5.8 *For the system (3.2) and a given structure matrix S , the AFDIP is solvable with conditions (3.29) if and only if the reduced system (5.11) is S -fault isolable, or equivalently, for $i = 1, \dots, n_b$*

$$\text{rank} [\overline{G}_d^{(i)}(\lambda) \overline{G}_{f_j}(\lambda)] > \text{rank} \overline{G}_d^{(i)}(\lambda), \quad \forall j, \quad S_{ij} \neq 0,$$

where $\overline{G}_d^{(i)}(\lambda)$ is formed from the columns $\overline{G}_{f_j}(\lambda)$ of $\overline{G}_f(\lambda)$ for which $S_{ij} = 0$.

To determine $\overline{Q}^{(i)}(\lambda)$ we can always try first to achieve the i -th specification *exactly*, by applying the **Procedure AFD** (see Sect. 5.3) to solve the AFDP for the reduced system (5.45), and determine a least-order fault detection filter $\overline{Q}^{(i)}(\lambda)$ in (5.38) which fully decouples $d^{(i)}(t)$. If the AFDP for the reduced system (5.45) is not solvable, then the **Procedure AFD** can be applied to solve the AFDP for the same reduced system (5.45), but with the disturbance inputs $d^{(i)}(t)$ redefined as additional noise inputs.

The **Procedure AFDI**, given below, determines for a given $n_b \times m_f$ structure matrix S , a bank of n_b least-order fault detection filters $Q^{(i)}(\lambda), i = 1, \dots, n_b$, which solve the AFDIP. Additionally, the block rows of $R_f(\lambda)$ and $R_w(\lambda)$ corresponding to $Q^{(i)}(\lambda)$ are determined as

$$R_f^{(i)}(\lambda) := Q^{(i)}(\lambda) \begin{bmatrix} G_f(\lambda) \\ 0 \end{bmatrix}, \quad R_w^{(i)}(\lambda) := Q^{(i)}(\lambda) \begin{bmatrix} G_w(\lambda) \\ 0 \end{bmatrix}.$$

The existence conditions for the solvability of the AFDIP are implicitly tested when applying the **Procedure AFD** to solve the appropriate AFDP for the system (5.45), with specified number of components q_i of $r^{(i)}$ and noise signal gain level γ . For each filter $Q^{(i)}(\lambda)$, the achieved fault sensitivity level β_i is also computed by the **Procedure AFD**.

Procedure AFDI: Approximate synthesis of FDI filters

Inputs : $\{G_u(\lambda), G_d(\lambda), G_w(\lambda), G_f(\lambda)\}, S \in \mathbb{R}^{n_b \times m_f}, \{q_1, \dots, q_{n_b}\}, \gamma$

Outputs: $Q^{(i)}(\lambda), R_f^{(i)}(\lambda), R_w^{(i)}(\lambda), \beta_i, i = 1, \dots, n_b$

- 1) Compute a $(p - r_d) \times (p + m_u)$ minimal basis matrix $Q_1(\lambda)$ for the left nullspace of $G(\lambda)$ defined in (5.2), where $r_d := \text{rank } G_d(\lambda)$;
 set $Q(\lambda) = Q_1(\lambda)$ and compute $[R_f(\lambda) \ R_w(\lambda)] = Q_1(\lambda) \begin{bmatrix} G_f(\lambda) & G_w(\lambda) \\ 0 & 0 \end{bmatrix}$.
- 2) For $i = 1, \dots, n_b$
 - 2.1) Form $\overline{G}_d^{(i)}(\lambda)$ from the columns $R_{f_j}(\lambda)$ for which $S_{ij} = 0$ and $\overline{G}_f^{(i)}(\lambda)$ from the columns $R_{f_j}(\lambda)$ for which $S_{ij} \neq 0$.
 - 2.2) Apply **Procedure AFD** to the system described by the quadruple $\{0, \overline{G}_d^{(i)}(\lambda), \overline{G}_f^{(i)}(\lambda), R_w(\lambda)\}$ to obtain the $q_i \times (p - r_d)$ least-order filter $\overline{Q}_1^{(i)}(\lambda)$ and β_i . **Go to** Step 2.4) if successful.
 - 2.3) Apply **Procedure AFD** to the system described by the quadruple $\{0, 0, \overline{G}_f^{(i)}(\lambda), [R_w(\lambda) \ \overline{G}_d^{(i)}(\lambda)]\}$ to obtain the $q_i \times (p - r_d)$ least-order filter $\overline{Q}_1^{(i)}(\lambda)$ and β_i . **Exit** if no solution exists.
 - 2.4) Compute $Q^{(i)}(\lambda) = \overline{Q}_1^{(i)}(\lambda)Q(\lambda)$, $R_f^{(i)}(\lambda) = \overline{Q}_1^{(i)}(\lambda)R_f(\lambda)$ and $R_w^{(i)}(\lambda) = \overline{Q}_1^{(i)}(\lambda)R_w(\lambda)$.

Remark 5.11 For the selection of the threshold τ_i for the component $r^{(i)}(t)$ of the residual vector we can use a similar approach to that described in Remark 5.9. To determine the false alarm bound we can use the corresponding internal representation of the resulting i -th fault detection filter in the form

$$\mathbf{r}^{(i)}(\lambda) = R_f^{(i)}(\lambda)\mathbf{f}(\lambda) + R_w^{(i)}(\lambda)\mathbf{w}(\lambda). \quad (5.46)$$

If we assume, for example, a bounded energy noise input $w(t)$ such that $\|w\|_2 \leq \delta_w$, then the false alarm bound $\tau_f^{(i)}$ for the i -th residual vector component $r^{(i)}(t)$ can be computed as

$$\tau_f^{(i)} = \sup_{\|w\|_2 \leq \delta_w} \|R_w^{(i)}(\lambda)\mathbf{w}(\lambda)\|_2 = \|R_w^{(i)}(\lambda)\|_\infty \delta_w. \quad (5.47)$$

However, by simply setting $\tau_i = \tau_f^{(i)}$, we can only detect the presence of a fault in any of the components of f , but we ignore the additional structural information needed for fault isolation. Therefore, we need to take into account the partition of the components of f into two distinct vectors, namely $f^{(i)}$, which contains those

components f_j of f for which $S_{ij} = 1$ (i.e., the faults to be detected in $r^{(i)}$) and $\bar{f}^{(i)}$, which contains those components f_j of f for which $S_{ij} = 0$ (i.e., the faults to be decoupled from $r^{(i)}$). By denoting $R_{f^{(i)}}^{(i)}(\lambda)$ and $R_{\bar{f}^{(i)}}^{(i)}(\lambda)$ the columns of $R_f^{(i)}(\lambda)$ corresponding to $f^{(i)}$ and $\bar{f}^{(i)}$, respectively, we can rewrite (5.46) in the form

$$\mathbf{r}^{(i)}(\lambda) = R_{f^{(i)}}^{(i)}(\lambda)\mathbf{f}^{(i)}(\lambda) + R_{\bar{f}^{(i)}}^{(i)}(\lambda)\bar{\mathbf{f}}^{(i)}(\lambda) + R_w^{(i)}(\lambda)\mathbf{w}(\lambda). \quad (5.48)$$

If we assume, for example, a bounded energy noise input $w(t)$ such that $\|w\|_2 \leq \delta_w$ and, similarly, a bounded energy fault input $\bar{f}^{(i)}(t)$ such that $\|\bar{f}^{(i)}\|_2 \leq \delta_{\bar{f}^{(i)}}$, then the false alarm bound for isolation $\tau_{fi}^{(i)}$ for the i -th residual vector component $r^{(i)}(t)$ can be bounded as follows:

$$\begin{aligned} \tau_{fi}^{(i)} &= \sup_{\substack{\|w\|_2 \leq \delta_w \\ \|\bar{f}^{(i)}\|_2 \leq \delta_{\bar{f}^{(i)}}}} \|R_{\bar{f}^{(i)}}^{(i)}(\lambda)\bar{\mathbf{f}}^{(i)}(\lambda) + R_w^{(i)}(\lambda)\mathbf{w}(\lambda)\|_2 \\ &\leq \|R_{\bar{f}^{(i)}}^{(i)}(\lambda)\|_\infty \delta_{\bar{f}^{(i)}} + \|R_w^{(i)}(\lambda)\|_\infty \delta_w := \tilde{\tau}_{fi}^{(i)}. \end{aligned} \quad (5.49)$$

The setting of the threshold $\tau_i = \tilde{\tau}_{fi}^{(i)}$ ensures no false isolation alarms due to faults occurring in $\bar{f}^{(i)}$. A somewhat smaller (i.e., less conservative) threshold can be used if additionally the information on the maximum number of faults which simultaneously may occur is included in bounding $\|R_{\bar{f}^{(i)}}^{(i)}(\lambda)\bar{\mathbf{f}}^{(i)}(\lambda)\|_2$. Note that if the i -th specification (coded in the i -th row of the structure matrix S) has been exactly achieved at Step 2.2) of the **Procedure AFDI**, then $R_{\bar{f}^{(i)}}^{(i)}(\lambda) = 0$ and therefore $\tau_f^{(i)} = \tau_{fi}^{(i)} = \tilde{\tau}_{fi}^{(i)}$. In this case we can set the threshold to the lowest value $\tau_i = \tau_f^{(i)}$ (i.e., the false alarm bound).

The least size $\delta_{f_j}^{(i)}$ of the fault f_j which can be detected in $r^{(i)}$ for $S_{ij} = 1$, can be estimated similarly as done in Remark 5.9 (see (5.33))

$$\delta_{f_j}^{(i)} = \frac{2\|R_w^{(i)}(\lambda)\|_\infty \delta_w}{\|R_{f_j}^{(i)}(\lambda)\|_{\Omega-}}, \quad (5.50)$$

where Ω is a given set of relevant frequency values. Overall, δ_{f_j} , the least size of the isolable fault f_j , can be defined as

$$\delta_{f_j} := \min_{i \in \mathcal{I}_j} \delta_{f_j}^{(i)},$$

where $\mathcal{I}_j := \{i : i \in \{1, \dots, n_b\} \wedge S_{ij} = 1\}$. □

Example 5.11 Consider the solution of the AFDIP for the system

$$G_u(s) = \begin{bmatrix} \frac{s+1}{s+2} \\ \frac{s+2}{s+3} \end{bmatrix}, \quad G_d(s) = 0, \quad G_w(s) = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \quad G_f(s) = \begin{bmatrix} \frac{s+1}{s+2} & 0 \\ 0 & 1 \end{bmatrix}$$

used also in Examples 5.5 and 5.8. At Step 1) of the **Procedure AFDI**, we compute a minimal left nullspace basis of $G(\lambda)$ defined in (5.2) as $Q_1(s) = [I_2 \ -G_u(s)]$, which leads to $R_w(s) = G_w(s)$ and $R_f(s) = G_f(s)$. By inspecting $R_f(s)$ it follows that the strong isolability condition is fulfilled (i.e., $\text{rank } R_f(s) = 2$), thus we can target to solve an AFDIP with $S = I_2$.

To achieve the specification in the first row of S , we define the reduced model (5.45) with $\bar{G}_d^{(1)}(s) = R_{f_2}(s)$ and $\bar{G}_f^{(1)}(s) = R_{f_1}(s)$. We can apply the **Procedure AFD** to solve the AFDP for the quadruple $\{0, \bar{G}_d^{(1)}(s), \bar{G}_f^{(1)}(s), R_w(s)\}$. At Step 1) we compute a left nullspace basis of $\bar{G}_d^{(1)}(s)$ as $Q_1^{(1)}(s) = [1 \ 0]$ and determine

$$R_w^{(1)}(s) := Q_1^{(1)}(s)R_w(s) = \frac{1}{s+2}, \quad R_f^{(1)}(s) := Q_1^{(1)}(s)R_{f_1}(s) = \frac{s+1}{s+2}.$$

Since $R_f^{(1)}(s) \neq 0$, it follows that the EFDP, and therefore also the AFDP has a solution according to Theorem 3.9. At Step 2) we take $Q_2^{(1)}(s) = 1$ and at Step 3), the quasi-co-outer factor $G_{wo}(s)$ is simply $R_w^{(1)}(s)$, which is strictly proper and, thus, has a zero at infinity. With $Q_3^{(1)}(s) = (R_w^{(1)}(s))^{-1}$, the resulting $\bar{Q}^{(1)}(s) := Q_3^{(1)}(s)Q_2^{(1)}(s)Q_1^{(1)}(s)$ and $\bar{R}_f^{(1)}(s) := Q_3^{(1)}(s)Q_2^{(1)}(s)Q_1^{(1)}(s)R_{f_1}(s)$ are therefore improper. At Step 4), we choose $Q_4^{(1)}(s)$ of unity \mathcal{H}_∞ -norm of the form $Q_4^{(1)}(s) = a/(s+a)$ with $a \geq 1$, such that $Q_4^{(1)}(s)[\bar{Q}^{(1)}(s) \ \bar{R}_f^{(1)}(s)]$ is stable and proper. For $\gamma = 1$ we obtain at Step 5) with $Q_5^{(1)} = 1$ the final $\bar{Q}^{(1)}(s)$ as

$$\bar{Q}^{(1)}(s) = Q_5^{(1)}(s)Q_4^{(1)}(s)Q_3^{(1)}(s)Q_2^{(1)}(s)Q_1^{(1)}(s) = \begin{bmatrix} \frac{a(s+2)}{s+a} & 0 \end{bmatrix}.$$

At Step 2.4) of **Procedure AFDI** we obtain

$$Q^{(1)}(s) = \bar{Q}^{(1)}(s)Q_1(s) = \begin{bmatrix} \frac{a(s+2)}{s+a} & 0 & -\frac{a(s+1)}{s+a} \end{bmatrix},$$

$$R_f^{(1)}(s) = \bar{Q}_1^{(1)}(s)R_f(s) = \begin{bmatrix} \frac{a(s+1)}{s+a} & 0 \end{bmatrix}, \quad R_w^{(1)}(s) = \bar{Q}_1^{(1)}(s)R_w(s) = \frac{a}{s+a}.$$

Since $\beta_1 = \|R_{f_1}^{(1)}(s)\|_\infty = a$ can be arbitrarily large, the underlying $\mathcal{H}_{\infty-}/\mathcal{H}_\infty$ problem has no optimal solution. Still, the resulting $Q^{(1)}(s)$ is completely satisfactory, by providing an arbitrary large gap $\beta_1/\gamma = a$.

To achieve the specification in the second row of S , we define $\bar{G}_d^{(2)}(s) = R_{f_1}(s)$ (the first column of $R_f(s)$) and $\bar{G}_f^{(2)}(s) = R_{f_2}(s)$. Again, we apply **Procedure AFD** to solve the AFDP for the quadruple $\{0, \bar{G}_d^{(2)}(s), \bar{G}_f^{(2)}(s), R_w(s)\}$. At Step 1) we compute a left nullspace basis of $\bar{G}_d^{(2)}(s)$ as $Q_1^{(2)}(s) = [0 \ -1]$ and determine

$$R_w^{(2)}(s) := Q_1^{(2)}(s)R_w(s) = 0, \quad R_f^{(2)}(s) := Q_1^{(2)}(s)R_{f_2}(s) = -1.$$

Observe that we actually solved the AFDP as an EFDP, by obtaining $\bar{Q}_1^{(2)}(s) = Q_1^{(2)}(s)$. At Step 2.4) of **Procedure AFDI** we obtain

$$Q^{(2)}(s) = \bar{Q}^{(2)}(s)Q_1(s) = \begin{bmatrix} 0 & -1 & \frac{s+2}{s+3} \end{bmatrix},$$

$$R_f^{(2)}(s) = \bar{Q}_1^{(2)}(s)R_f(s) = [0 \ -1], \quad R_w^{(2)}(s) = \bar{Q}_1^{(2)}(s)R_w(s) = 0.$$

Although $\beta_2 = \|R_f^{(2)}(s)\|_\infty = 1$, but β_2 can be arbitrary large by suitably rescaling $Q^{(2)}(s)$.

The script **Ex5_11** in Listing 5.4 solves the AFDIP considered in this example. \diamond

Listing 5.4 Script **Ex5_11** to solve the AFDIP of Example 5.11 using Procedure AFDI

```

% Uses the Control Toolbox and the Descriptor System Tools

% define s as an improper transfer function
s = tf('s');
% define G_u(s), G_w(s), G_f(s)
Gu = [(s+1)/(s+2); (s+2)/(s+3)]; % enter G_u(s)
Gw = [1/(s+2); 0]; % enter G_w(s)
Gf = [(s+1)/(s+2) 0; 0 1]; % enter G_f(s)
p = 2; mu = 1; mw = 1; mf = 2; % enter dimensions
S = eye(mf); % enter structure matrix

% Step 1): choose the left nullspace as Q_1 = [I - G_u] and
% form R_{f,1} = Q_1[G_f; 0] and R_{w,1} = Q_1[G_w; 0]
Q1 = [eye(p) -Gu]; Rf1 = Gf; Rw1 = Gw;

% Step 2): determine Q^{(i)} and the corresponding R_f^{(i)} and R_w^{(i)}

% initialization
nb = size(S,1); % number of necessary filters n_b
Q = cell(nb,1); Rf = cell(nb,1); Rw = cell(nb,1);

% set options for coprime factorizations;
% enforce stability degree -3
opts = struct('sdeg',-3,'smarg',-3);
for i = 1:nb
    % perform Procedure AFD to compute Q^{(i)}
    indd = (S(i,:) == 0);
    Q1 = glnull(ss(Rf1(:,indd))); % compute Q_1^{(i)}
    % initialize Q^{(i)}, R_f^{(i)}, and R_w^{(i)}
    Qi = Q1*Q1; Rfi = Q1*Rf1; Rwi = Q1*Rw1;

    % compute optimal solution if R_w^{(i)} nonzero
    if norm(evalfr(Rwi,rand)) > 0.0001
        % compute the quasi-co-outer-co-inner factorization
        [Rwi,Rwo]=goifac(Rwi,1.e-7); % R_w^{(i)} = R_{wo}^{(i)}R_{wi}^{(i)}; R_w^{(i)} \leftarrow R_{wi}^{(i)}
        % update Q^{(i)} \leftarrow (R_{wo}^{(i)})^{-1}Q^{(i)} and R_f^{(i)} \leftarrow (R_{wo}^{(i)})^{-1}R_f^{(i)}
        Qi = Rwo\Qi; Rfi = Rwo\Rfi;
    end

    % update the solution if [Q^{(i)} R_f^{(i)} R_w^{(i)}] is improper or unstable
    [Qi_Rfi_Rwi,M] = glcf([Qi Rfi Rwi],opts);
    % adjust denominator to unit infinity norm to match example
    scale = norm(M,inf);
    Q{i} = tf(Qi_Rfi_Rwi(:,1:p+mu)/scale);
    Rf{i} = tf(Qi_Rfi_Rwi(:,p+mu+1:p+mu+mf)/scale);
    Rw{i} = tf(Qi_Rfi_Rwi(:,p+mu+mf+1:end)/scale);
end

```

5.6 Solving the Exact Model-Matching Problem

Let $M_r(\lambda)$ be a given $q \times m_f$ TFM of a stable and proper reference model specifying the desired input–output behaviour from the faults to residuals as $\mathbf{r}(\lambda) = M_r(\lambda)\mathbf{f}(\lambda)$. Using the factorized representation $Q(\lambda) = \bar{Q}_1(\lambda)Q_1(\lambda)$ in (5.9), it follows that the *exact model-matching problem* (EMMP) formulated in Sect. 3.5.5 is solvable for the system (3.2) with $w \equiv 0$ if it is solvable for the reduced system (5.11) with $w \equiv 0$. The following corollary to Theorem 3.14 provides the solvability conditions of the EMMP in terms of the reduced system (5.11):

Corollary 5.9 *For the system (3.2) with $w \equiv 0$ and a given reference model $M_r(\lambda)$, the EMMP is solvable if and only if the EMMP is solvable for the reduced system (5.11) with $w \equiv 0$, or equivalently, the following condition is fulfilled:*

$$\text{rank } \bar{G}_f(\lambda) = \text{rank} \begin{bmatrix} \bar{G}_f(\lambda) \\ M_r(\lambda) \end{bmatrix}. \quad (5.51)$$

The case when $M_r(\lambda)$ is diagonal and invertible corresponds to a strong FDI requirement. The solvability condition for this case is the same as the solvability condition resulting from (5.51) for the case when $M_r(\lambda)$ has full column rank m_f .

Corollary 5.10 *For the system (3.2) with $w \equiv 0$ and a given reference model $M_r(\lambda)$ with $\text{rank } M_r(\lambda) = m_f$, the EMMP is solvable if and only if the reduced system (5.11) with $w \equiv 0$ is strongly isolable, or equivalently, the following condition is fulfilled:*

$$\text{rank } \bar{G}_f(\lambda) = m_f. \quad (5.52)$$

Remark 5.12 For a strongly isolable system (3.2) with $w \equiv 0$, the left invertibility condition (5.52) is a necessary and sufficient condition for the solvability of the EMMP for an arbitrary $M_r(\lambda)$. \square

For the solution of the EMMP, we present a synthesis procedure which employs the factorized representation $Q(\lambda) = \bar{Q}_1(\lambda)Q_1(\lambda)$ in (5.9), where $Q_1(\lambda)$ is a minimal proper left nullspace basis of $G(\lambda)$ defined in (5.2). The factor $\bar{Q}_1(\lambda)$ can be determined in the product form

$$\bar{Q}_1(\lambda) = Q_3(\lambda)Q_2(\lambda),$$

where $Q_2(\lambda)$ is a solution, possibly of least McMillan degree, of the linear rational matrix equation

$$Q_2(\lambda)\bar{G}_f(\lambda) = M_r(\lambda), \quad (5.53)$$

while the diagonal updating factor $Q_3(\lambda) := M(\lambda)$ is determined such that

$$Q(\lambda) = Q_3(\lambda)Q_2(\lambda)Q_1(\lambda)$$

is stable and proper. The computation of $Q_3(\lambda)$ is necessary only if $Q_2(\lambda)Q_1(\lambda)$ is not proper or is unstable. The **Procedure EMM**, given below, summarizes the main computational steps for solving the EMMP.

Procedure EMM: Exact model-matching synthesis of FDI filters

Inputs : $\{G_u(\lambda), G_d(\lambda), G_f(\lambda)\}, M_r(\lambda)$

Outputs: $Q(\lambda), M(\lambda)$

- 1) Compute a minimal proper basis $Q_1(\lambda)$ for the left nullspace of $G(\lambda)$ defined in (5.2); set $Q(\lambda) = Q_1(\lambda)$ and compute $R_f(\lambda) = Q_1(\lambda) \begin{bmatrix} G_f(\lambda) \\ 0 \end{bmatrix}$.
- 2) Solve, for the least McMillan degree solution $Q_2(\lambda)$, the linear rational matrix equation $Q_2(\lambda)R_f(\lambda) = M_r(\lambda)$. **Exit** if no solution exists. Otherwise update $Q(\lambda) \leftarrow Q_2(\lambda)Q(\lambda)$.
- 3) Determine a diagonal, stable, proper and invertible $Q_3(\lambda) := M(\lambda)$ such that $M(\lambda)Q(\lambda)$ is stable and proper; update $Q(\lambda) \leftarrow Q_3(\lambda)Q(\lambda)$.

To perform the computation at Step 2), a state-space realization based algorithm to compute least McMillan degree solutions of linear rational matrix equations is described in Sect. 10.3.7. For the determination of the diagonal updating factor $M(\lambda)$ at Step 3), coprime factorization techniques can be used, as described in Sect. 9.1.6. The underlying state-space realization based algorithms are presented in Sect. 10.3.5.

Remark 5.13 The solution of the EMMP can be alternatively performed by determining $Q(\lambda)$ as $Q(\lambda) = Q_2(\lambda)Q_1(\lambda)$, where $Q_1(\lambda)$ is a least McMillan degree solution of the linear rational matrix equation

$$Q_1(\lambda) \begin{bmatrix} G_u(\lambda) & G_d(\lambda) & G_f(\lambda) \\ I_{m_u} & 0 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 & M_r(\lambda) \end{bmatrix}. \quad (5.54)$$

The diagonal updating factor $Q_2(\lambda) := M(\lambda)$ is determined to ensure that $Q(\lambda)$ is proper and stable. \square

Example 5.12 In Example 5.10 we solved an EFDIP for a system with triplex sensor redundancy. To solve an EMMP for the same system, we use the resulting $R_f(s)$ to define the reference model

$$M_r(s) := R_f(s) = \begin{bmatrix} 0 & 1 & -1 \\ -1 & 0 & 1 \\ 1 & -1 & 0 \end{bmatrix}.$$

Using **Procedure EMM**, we determine first a left nullspace basis $Q_1(s) = N_l(s)$, with $N_l(s)$ given in (5.42). The corresponding $R_f(s)$ (given in (5.43)) is

$$R_f(s) = \begin{bmatrix} 1 & -1 & 0 \\ 0 & 1 & -1 \end{bmatrix}.$$

The solvability condition can be easily checked

$$\text{rank } R_f(s) = \text{rank} \begin{bmatrix} R_f(s) \\ M_r(s) \end{bmatrix} = 2.$$

We solve for $Q_2(s)$

$$Q_2(s)R_f(s) = M_r(s)$$

and obtain

$$Q_2(s) = \begin{bmatrix} 0 & 1 \\ -1 & -1 \\ 1 & 0 \end{bmatrix}.$$

Finally, we have

$$Q(s) = Q_2(s)Q_1(s) = \begin{bmatrix} 0 & 1 & -1 & 0 & \cdots & 0 \\ -1 & 0 & 1 & 0 & \cdots & 0 \\ 1 & -1 & 0 & 0 & \cdots & 0 \end{bmatrix}.$$

We obtain the same result by solving directly (5.54) for $Q(s) = Q_1(s)$.

The script **Ex5_12** in Listing 5.5 solves the EMMP considered in this example. \diamond

Listing 5.5 Script **Ex5_12** to solve the EMMP of Example 5.12 using Procedure EMM

```
% Uses the Control Toolbox and the Descriptor System Tools

% enter output and fault vector dimensions
p = 3; mf = 3;
% generate random dimensions for system order and input vectors
nu = floor(1+4*rand); mu = floor(1+4*rand);
nd = floor(1+4*rand); md = floor(1+4*rand);
% define random Gu(s) and Gd(s) with triplex sensor redundancy
% and Gf(s) for triplex sensor faults
Gu = ones(3,1)*rss(nu,1,mu); % enter Gu(s) in state-space form
Gd = ones(3,1)*rss(nd,1,md); % enter Gd(s) in state-space form
Gf = eye(3); % enter Gf(s) for sensor faults
% enter reference model Mr
Mr = ss([ 0 1 -1; -1 0 1; 1 -1 0]);

% two step solution using Procedure EMM
% 1. Compute left nullspace Q1 of [Gu Gd; I 0] and
% the reduced system Rf = Q1[Gf; 0].
Q_Rf = glnull([Gu Gd Gf; eye(mu,mu+md+mf)], struct('m2',mf));
Q1 = Q_Rf(:,1:p+mu); Rf = Q_Rf(:,p+mu+1:end);

% 2. Solve Q2Rf = Mr, and form Q = Q2Q1.
Q2 = glsol(Rf,Mr,struct('tol',1.e-7));
Q = Q2*Q1;

% one step solution
% solve QGe = Me, where Ge = [Gu Gd Gf; I 0] and Me = [0 0 Mr].
Ge = [Gu Gd Gf; eye(mu,mu+md+mf)]; Me = [zeros(p,mu+md) Mr];
Qbar = glsol(Ge,Me,struct('tol',1.e-7));

% compare solutions by computing ||Q - Qbar||_inf
norm(Q-Qbar,inf)
```

In what follows, we discuss the solution of the EMMP for strongly isolable systems. According to Remark 5.12, the solvability of the EMMP is automatically guaranteed in this case, regardless the choice of the reference model $M_r(\lambda)$. An important

particular case in practical applications is when $M_r(\lambda)$ is diagonal, stable, proper and invertible. In this case, the solution of the EMMP allows the detection and isolation of up to m_f simultaneous faults, and thus is also a solution of a strong EFDIP (i.e., for an identity structure matrix). Fault reconstruction (or fault estimation) problems can be addressed in this way by choosing $M_r(\lambda) = I_{m_f}$. For the solution of the EMMP for strongly isolable systems we develop a specialized synthesis procedure, which also addresses the least-order synthesis aspect for a regularity-enforcing admissibility condition.

Using the factorized representation $Q(\lambda) = \bar{Q}_1(\lambda)Q_1(\lambda)$ in (5.9), the factor $\bar{Q}_1(\lambda)$ can be determined in the product form

$$\bar{Q}_1(\lambda) = \bar{Q}_2(\lambda)Q_2(\lambda),$$

where $Q_2(\lambda)$ is computed such that

$$\tilde{G}_f(\lambda) := Q_2(\lambda)\bar{G}_f(\lambda)$$

is invertible. This regularization step is always possible, since, for a strongly isolable system, $\bar{G}_f(\lambda)$ is left invertible (see Remark 5.12). The simplest choice of $Q_2(\lambda)$ is a constant (e.g., orthogonal) projection matrix which simply selects m_f linearly independent rows of $\bar{G}_f(\lambda)$. A more involved choice is based on an *admissibility condition*, which enforces the invertibility of $\tilde{G}_f(\lambda)$ simultaneously with the least dynamical orders of $Q_2(\lambda)Q_1(\lambda)$ and $\tilde{G}_f(\lambda)$. Such a choice of $Q_2(\lambda)$ is possible using minimal dynamic cover techniques (see Sect. 7.5).

The factor $\bar{Q}_2(\lambda)$ can be determined in the form

$$\bar{Q}_2(\lambda) = Q_4(\lambda)Q_3(\lambda),$$

where $Q_3(\lambda) = M_r(\lambda)\tilde{G}_f^{-1}(\lambda)$ and $Q_4(\lambda) := M(\lambda)$ is chosen a diagonal, stable, proper and invertible TFM, to ensure that the resulting final filter

$$Q(\lambda) = Q_4(\lambda)Q_3(\lambda)Q_2(\lambda)Q_1(\lambda)$$

is stable and proper. The updating factor $M(\lambda)$ can be determined using stable and proper coprime factorization techniques (see Sects. 9.1.6 and 10.3.5).

The above synthesis method is sometimes called in the literature the *inversion-based method*. The **Procedure EMMS**, given in what follows, formalizes the computational steps of the inversion-based synthesis method to solve the EMMP for strongly isolable systems.

Procedure EMMS: Exact model-matching synthesis of strong FDI filters**Inputs :** $\{G_u(\lambda), G_d(\lambda), G_f(\lambda)\}, M_r(\lambda)$ **Outputs:** $Q(\lambda), M(\lambda)$

- 1) Compute a proper minimal basis $Q_1(\lambda)$ for the left nullspace of $G(\lambda)$ defined in (5.2); set $Q(\lambda) = Q_1(\lambda)$ and compute $R_f(\lambda) = Q_1(\lambda) \begin{bmatrix} G_f(\lambda) \\ 0 \end{bmatrix}$.
Exit if $\text{rank } R_f(\lambda) < m_f$ (no solution).
- 2) Choose $Q_2(\lambda)$ such that $Q_2(\lambda)R_f(\lambda)$ is invertible and $Q_2(\lambda)[R_f(\lambda) \ Q(\lambda)]$ has least-order; update $Q(\lambda) \leftarrow Q_2(\lambda)Q(\lambda)$ and $R_f(\lambda) \leftarrow Q_2(\lambda)R_f(\lambda)$.
- 3) With $Q_3(\lambda) = M_r(\lambda)R_f^{-1}(\lambda)$, update $Q(\lambda) \leftarrow Q_3(\lambda)Q(\lambda)$.
- 4) Determine diagonal, stable, proper and invertible $Q_4(\lambda) := M(\lambda)$ such that $M(\lambda)Q(\lambda)$ is stable and proper; update $Q(\lambda) \leftarrow Q_4(\lambda)Q(\lambda)$.

Example 5.13 Consider a continuous-time system with the transfer function matrices

$$G_u(s) = \begin{bmatrix} \frac{s}{s^2 + 3s + 2} & \frac{1}{s + 2} \\ \frac{s}{s + 1} & 0 \\ 0 & \frac{1}{s + 2} \end{bmatrix}, \quad G_d(s) = 0, \quad G_f(s) = \begin{bmatrix} \frac{s}{s^2 + 3s + 2} & \frac{1}{s + 2} \\ \frac{s}{s + 1} & 0 \\ 0 & \frac{1}{s + 2} \end{bmatrix}$$

for which we want to solve the EMMP with the reference model

$$M_r(s) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}.$$

Using **Procedure EMMS**, we choose at Step 1) the left nullspace basis $Q_1(s) = [I - G_u(s)]$ and initialize $Q(s) = Q_1(s)$ for which the corresponding $R_f(s)$ is simply $R_f(s) = G_f(s)$. $R_f(s)$ has full column rank (thus is left invertible) and therefore the EMMP has a solution. Since $R_f(s)$ has zeros in the origin and at infinity, the existence condition of Lemma 9.5 for a stable solution $\bar{Q}_1(s)$ of $\bar{Q}_1(s)R_f(s) = M_r(s)$ is not fulfilled.

At Step 2), we choose $Q_2(s)$ such that $Q_2(s)[R_f(s) \ Q(s)]$ has a least-order. This can be achieved with the simple choice

$$Q_2(s) = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

and, after updating $Q(s) \leftarrow Q_2(s)Q(s)$ and $R_f(s) \leftarrow Q_2(s)R_f(s)$, we obtain

$$Q(s) = \begin{bmatrix} 0 & 1 & 0 & -\frac{s}{s+1} & 0 \\ 0 & 0 & 1 & 0 & -\frac{1}{s+2} \end{bmatrix}, \quad R_f(s) = \begin{bmatrix} \frac{s}{s+1} & 0 \\ 0 & \frac{1}{s+2} \end{bmatrix}.$$

At Step 3), the resulting

$$Q_3(s) := M_r(s)R_f^{-1}(s) = \begin{bmatrix} \frac{s+1}{s} & 0 \\ 0 & s+2 \end{bmatrix}$$

is improper and unstable, and, therefore, the updated $Q(s) \leftarrow \tilde{Q}(s) := Q_3(s)Q(s)$

$$\tilde{Q}(s) = \begin{bmatrix} 0 & \frac{s+1}{s} & 0 & -1 & 0 \\ 0 & 0 & s+2 & 0 & -1 \end{bmatrix}$$

is improper and unstable as well.

Finally, at Step 4) we determine a diagonal

$$Q_4(s) = M(s) = \begin{bmatrix} \frac{s}{s+1} & 0 \\ 0 & \frac{1}{s+1} \end{bmatrix},$$

which ensures the properness and stability of the solution. The final $Q(s) = Q_4(s)\tilde{Q}(s)$ is

$$Q(s) = \begin{bmatrix} 0 & 1 & 0 & -\frac{s}{s+1} & 0 \\ 0 & 0 & \frac{s+2}{s+1} & 0 & -\frac{1}{s+1} \end{bmatrix}.$$

The McMillan degree of this fault detection filter is 2 and is the least achievable one among all stable and proper filters which solve the EMMP. Note that the presence of the zero $s = 0$ in $M(s)M_r(s)$ is unavoidable for the existence of a stable solution. It follows, that while a constant fault f_2 is strongly detectable, a constant fault f_1 is only detectable during transients.

An alternative way, see Remark 5.13, to determine a least-order solution of the considered EMMP is to directly solve, for the least-order solution $Q_1(s)$, the linear rational matrix equation

$$Q_1(s) \begin{bmatrix} G_u(s) & G_f(s) \\ I & 0 \end{bmatrix} = [0 \ M_r(s)]$$

and then determine $Q_2(s)$ (as above $Q_4(s)$ at Step 4) of **Procedure EMMS**) to obtain a stable and proper $Q(s) := Q_2(s)Q_1(s)$.

The script **Ex5_13** in Listing 5.6 solves the EMMP considered in this example. The alternative direct approach is implemented in the script **Ex5_13a** (not listed). \diamond

Listing 5.6 Script **Ex5_13** to solve the EMMP of Example 5.13 using **Procedure EMMS**

% Uses the Control Toolbox and the Descriptor System Tools

% define s as an improper transfer function

s = tf('s');

% enter $G_u(s)$, $G_f(s)$ and $M_r(s)$

Gu = [s/(s^2+3*s+2) 1/(s+2);

s/(s+1) 0;

0 1/(s+2)];

Gf = [s/(s^2+3*s+2) 1/(s+2);

s/(s+1) 0;

0 1/(s+2)];

Mr = tf(eye(2));

[p,mf] = size(Gf); mu = size(Gu,2);

% compute left nullspace basis as $Q_1(s) = [I - G_u(s)]$; set $R_f(s) = G_f(s)$.

Q1 = [eye(p) -Gu]; Rf = Gf;

% check solvability condition

if rank(evalfr(Rf,rand)) ~= mf

error('No solution')

end

```

% check for unstable or infinite zeros
gzero(ss(Rf)) % zeros at infinity and in the origin exist

tol = 1.e-7; % set tolerance
sdeg = -1; % set stability degree

% solve a minimum dynamic cover problem
% select rows [2 3] of [Rf Q1] to combine them with row 1
% Rf_Q contains [Rf Q] ← Q2[Rf Q1] with Rf full row rank
cinp = [ 2 3 1];
Rf_Q = glmcover1(ss([Rf(cinp,:) Q1(cinp,:) ]),mf,tol);

% compute irreducible realization of  $\tilde{Q} = M_r R_f^{-1} Q$  by first solving
% the linear rational matrix equation  $R_f X = Q$ 
X = grsol(Rf_Q,p+mu,struct('tol',tol));
Qtilde = gir(Mr*X,tol);

% compute stable and proper  $Q = Q_4 \tilde{Q}$  with suitable diagonal  $Q_4 = M$ 
Q = ss(zeros(0,p+mu)); M = ss(zeros(0,0));
opt_glcf = struct('tol',tol,'sdeg',sdeg);
for i=1:mf
    [Qi,Mi] = glcf(Qtilde(i,:),opt_glcf);
    scale = get(zpk(Mi),'k'); % scale with gain to fit example
    Q = [Q;Qi/scale]; M = append(M,Mi/scale);
end

% convert to standard state-space representation
Q = gss2ss(Q); M = gss2ss(M);
% display results
minreal(tf(Q), tf(M))

% check solution
G = [Gu Gf;eye(mu,mu+mf)]; F = [zeros(mf,mu) M*Mr];
norm(Q*G-F,inf)

```

5.7 Solving the Approximate Model-Matching Problem

Using the factorized representation $Q(\lambda) = \bar{Q}_1(\lambda)Q_1(\lambda)$ in (5.9), with $Q_1(\lambda)$ stable and proper, allows to reformulate the *approximate model-matching problem* (AMMP) formulated in Sect. 3.5.6 for the system (3.2) in terms of the reduced system (5.11), with both $\bar{G}_f(\lambda)$ and $\bar{G}_w(\lambda)$ assumed to be stable and proper (this can be always enforced by a suitable choice of $Q_1(\lambda)$). The following corollary to Proposition 3.1 gives a sufficient condition for the solvability of the AMMP in terms of the reduced system (5.11):

Corollary 5.11 *For the system (3.2) and a given $M_r(\lambda)$, the AMMP is solvable if the EMMP is solvable for the reduced system (5.11).*

According to Remark 5.12, for a strongly isolable system (3.2), the left invertibility condition (5.52) (i.e., $\text{rank } \overline{G}_f(\lambda) = m_f$), is, therefore, a sufficient condition for the solvability of the AMMP.

To solve the AMMP for the reduced system (5.11), a standard model-matching problem can be formulated to determine the optimal stable and proper solution $\overline{Q}_1(\lambda)$ of the norm-minimization problem

$$\|\overline{Q}_1(\lambda) [\overline{G}_f(\lambda) \overline{G}_w(\lambda)] - [M_r(\lambda) 0]\| = \text{minimum} . \quad (5.55)$$

With $\overline{F}(\lambda) := [M_r(\lambda) 0]$, $\overline{G}(\lambda) := [\overline{G}_f(\lambda) \overline{G}_w(\lambda)]$, and the error function

$$\overline{\mathcal{E}}(\lambda) := \overline{F}(\lambda) - X(\lambda)\overline{G}(\lambda) , \quad (5.56)$$

a solution of the AMMP can be aimed by solving either a \mathcal{H}_∞ - or \mathcal{H}_2 -*model-matching problem* (MMP) (see Sect. 9.1.10) to determine $\overline{Q}_1(\lambda)$ as the stable and proper optimal solution $X(s)$ which minimizes $\|\overline{\mathcal{E}}(\lambda)\|_\infty$ or $\|\overline{\mathcal{E}}(\lambda)\|_2$, respectively. Sufficient conditions for the solvability of the \mathcal{H}_∞ -MMP and \mathcal{H}_2 -MMP are given in Lemmas 9.6 and 9.7, respectively. These sufficient conditions require that $\overline{G}(\lambda)$ has no zeros in $\partial\mathbb{C}_s$. However, these conditions are not necessary for the solvability of the AMMP, and, therefore, we define the *standard case*, when $\overline{G}(\lambda)$ has no zeros in $\partial\mathbb{C}_s$, and the *nonstandard case*, when $\overline{G}(\lambda)$ has zeros in $\partial\mathbb{C}_s$.

Solution procedures for the standard case are presented in Sect. 9.1.10 and determine optimal solutions which are stable and proper. The same procedures applied in the nonstandard case, determine “optimal” solutions, which, in general, have poles in $\partial\mathbb{C}_s$, and thus are unstable or improper. If $X(\lambda)$ is such a solution, then a diagonal, stable, proper and invertible updating factor $M(\lambda)$ can be determined such that the filter $\overline{Q}_1(\lambda) := M(\lambda)X(\lambda)$ is stable and proper, and achieves the (suboptimal) performance level $\gamma_{sub} := \|M(\lambda)\overline{\mathcal{E}}(\lambda)\|$. Let $\overline{X}(\lambda)$ be an “optimal” solution (possibly unstable or improper) which minimizes the weighted error norm $\|M(\lambda)\overline{\mathcal{E}}(\lambda)\|$ and let γ_{opt} be the corresponding optimal performance level. Since $\gamma_{opt} \leq \gamma_{sub}$, the difference $\gamma_{sub} - \gamma_{opt}$ is an indicator of the achieved degree of suboptimality of the resulting filter $\overline{Q}_1(\lambda)$ for the weighted norm-minimization problem corresponding to the updated reference model $M(\lambda)M_r(\lambda)$. The choice of a diagonal $M(\lambda)$ is instrumental to preserve the zero–nonzero structure of $M_r(\lambda)$.

Example 5.14 Consider the \mathcal{H}_∞ -MMP in a continuous-time setting with

$$\overline{G}(s) = [\overline{G}_f(s) \mid \overline{G}_w(s)] := \left[\frac{1}{s+1} \mid \frac{1}{s+2} \right], \quad \overline{F}(s) = [M_r(s) \mid 0] = \left[\frac{1}{s+3} \mid 0 \right].$$

This problem is nonstandard, because $\overline{G}(s)$ has a zero at infinity. Ignoring momentarily this aspect, we can formally use the solution approach in Sect. 9.1.10 relying on the quasi-co-outer–inner factorization of $\overline{G}(s)$ followed by the solution of a 2-block \mathcal{H}_∞ -least distance problem. We obtain the \mathcal{H}_∞ -optimal solution

$$X_\infty(s) = \frac{0.041587(s+13.65)(s+2)(s+1)}{(s+3)(s+1.581)} ,$$

which is improper. The optimal error norm is $\gamma_{\infty, opt} := \|\bar{F}(\lambda) - X_{\infty}(s)\bar{G}(s)\|_{\infty} = 0.1745$, thus finite. With $M(s) = \frac{1}{s+1}$, we obtain a proper candidate filter

$$\bar{Q}_1(s) = M(s)X_{\infty}(s) = \frac{0.041587(s + 13.65)(s + 2)}{(s + 3)(s + 1.581)},$$

for which $\gamma_{sub} := \|M(\lambda)\bar{F}(\lambda) - \bar{Q}_1(s)\bar{G}(s)\|_{\infty} = 0.1522$. The optimal solution $\bar{X}_{\infty}(s)$ of the \mathcal{H}_{∞} -MMP, which minimizes $\|M(s)\bar{F}(s) - X(s)\bar{G}(s)\|_{\infty}$, leads to an optimal value of $\gamma_{opt} = \|M(\lambda)\bar{F}(\lambda) - \bar{X}_{\infty}(\lambda)\bar{G}(\lambda)\|_{\infty} = 0.1491$. As expected, the optimal solution $\bar{X}_{\infty}(s)$ is improper. Since $\gamma_{sub} - \gamma_{opt} = 0.0031$, the degree of suboptimality of the proper and stable filter $M(s)X_{\infty}(s)$ with respect to the optimal (but improper) solution $\bar{X}_{\infty}(s)$ appears to be acceptable. \diamond

Example 5.15 We can also solve the \mathcal{H}_2 -MMP for Example 5.14. Although this problem is non-standard, still cancelations of infinite poles and zeros make that the resulting \mathcal{H}_2 -optimal solution is proper

$$X_2(s) = \frac{0.54572(s + 2)(s + 1)}{(s + 3)(s + 1.581)}.$$

The corresponding optimal performance is $\gamma_{opt} = \|\bar{F}(\lambda) - X_2(\lambda)\bar{G}(\lambda)\|_2 = 0.2596$. Interestingly, the \mathcal{H}_{∞} error norm of the \mathcal{H}_2 -optimal solution is $\|\bar{F}(\lambda) - X_2(s)\bar{G}(s)\|_{\infty} = 0.1768$, which is only marginally worse than $\gamma_{\infty, opt}$, the optimal performance of the improper \mathcal{H}_{∞} -optimal solution $X_{\infty}(\lambda)$. Thus, $X_2(s)$ can be considered an acceptable \mathcal{H}_{∞} -suboptimal solution.

In what follows we develop a general synthesis procedure for solving AMMPs relying on the solution of $\mathcal{H}_{2/\infty}$ -MMPs. We assume that the reference model $M_r(\lambda)$ has been chosen to capture a fault estimation or, equivalently, a strong fault isolation setup. Often, $M_r(\lambda)$ is chosen diagonal, and even equal to the identity matrix, when trying to solve a fault estimation problem. Therefore, $M_r(\lambda)$ will be assumed to be a stable and invertible TFM. In the case of an EMMP (when $w \equiv 0$), a necessary and sufficient condition for the existence of a proper and stable solution (possibly with an updated reference model $M(\lambda)M_r(\lambda)$, with $M(\lambda)$ a diagonal, stable and invertible factor) is that $\bar{G}_f(\lambda)$ has full column rank (i.e., left invertible) (see Corollary 5.10). For simplicity, we will assume that this condition is fulfilled and provide a synthesis procedure which computes an optimal solution in the standard case or a suboptimal solution of a weighted problem in the nonstandard case. As it will be apparent, the final fault detection filter intrinsically results in a factored form as in (5.1), which automatically leads to a synthesis procedure relying on successive updating of partially synthesized filters.

Let $\ell \geq m_f$ be the rank of the $(p - r_d) \times (m_f + m_w)$ TFM $\bar{G}(\lambda)$. We take $\bar{Q}_1(\lambda) = \bar{Q}_2(\lambda)Q_2(\lambda)$, where $Q_2(\lambda)$ is an $\ell \times (p - r_d)$ proper TFM chosen to ensure that $Q_2(\lambda)\bar{G}(\lambda)$ has full row rank ℓ . If $\ell < p - r_d$ (i.e., $\bar{G}(\lambda)$ has not a full row rank), a possible choice of $Q_2(\lambda)$ is one which simultaneously minimizes the McMillan degree of $Q_2(\lambda)Q_1(\lambda)$ (see Sect. 7.5). A simpler choice with $Q_2(\lambda)$ a constant (e.g., orthogonal) matrix is also always possible. If $\ell = p - r_d$, then $Q_2(\lambda) = I_{\ell}$ can be chosen.

The next step is standard in solving $\mathcal{H}_{2/\infty}$ -MMPs and consists in compressing the full row rank TFM $\bar{G}(\lambda)$ to a full column rank (thus invertible) TFM. For this, we compute an extended quasi-co-outer-co-inner factorization in the form

$$Q_2(\lambda)\overline{G}(\lambda) = [G_o(\lambda) \ 0] \begin{bmatrix} G_{i,1}(\lambda) \\ G_{i,2}(\lambda) \end{bmatrix} := [G_o(\lambda) \ 0]G_i(\lambda), \quad (5.57)$$

where the quasi-co-outer part $G_o(\lambda)$ is invertible and has only zeros in $\overline{\mathbb{C}}_s$, and $G_i(\lambda)$ is a square co-inner factor (i.e., $G_i(\lambda)G_i^*(\lambda) = I$). The factor $\overline{Q}_2(\lambda)$ is determined in the product form

$$\overline{Q}_2(\lambda) = Q_5(\lambda)Q_4(\lambda)Q_3(\lambda),$$

with $Q_3(\lambda) = G_o^{-1}(\lambda)$ and $Q_4(\lambda)$, the optimal solution which minimizes the error norm $\|\tilde{\mathcal{E}}(\lambda)\|_{2/\infty}$, with $\tilde{\mathcal{E}}(\lambda)$ defined as

$$\tilde{\mathcal{E}}(\lambda) := \overline{\mathcal{E}}(\lambda)G_i^*(\lambda) = [\tilde{F}_1(\lambda) - Q_4(\lambda) \tilde{F}_2(\lambda)], \quad (5.58)$$

where $\tilde{F}_1(\lambda) := \overline{F}(\lambda)G_{i,1}^*(\lambda)$ and $\tilde{F}_2(\lambda) := \overline{F}(\lambda)G_{i,2}^*(\lambda)$. The factor $Q_5(\lambda) := M(\lambda)$ is chosen to enforce the stability and properness of the final filter

$$Q(\lambda) = Q_5(\lambda)Q_4(\lambda)Q_3(\lambda)Q_2(\lambda)Q_1(\lambda). \quad (5.59)$$

The determination of a stable and proper $Q_4(\lambda)$ which minimizes $\|\tilde{\mathcal{E}}(\lambda)\|_{2/\infty} = \|\overline{\mathcal{E}}(\lambda)\|_{2/\infty}$ is a $\mathcal{H}_{2/\infty}$ -least distance problem ($\mathcal{H}_{2/\infty}$ -LDP), for which solution methods are given in Sect. 9.1.10.

The overall filter $Q(\lambda)$ in (5.59) can be alternatively expressed in the form $Q(\lambda) = Q_5(\lambda)Q_4(\lambda)\overline{Q}(\lambda)$, where $\overline{Q}(\lambda) := Q_3(\lambda)Q_2(\lambda)Q_1(\lambda)$ can be interpreted as a partial synthesis. The TFMs of the internal form corresponding to this filter are

$$\begin{aligned} [\overline{R}_f(\lambda) \ \overline{R}_w(\lambda)] &:= Q_3(\lambda)Q_2(\lambda)[\overline{G}_f(\lambda) \ \overline{G}_w(\lambda)] \\ &= [I_\ell \ 0] \begin{bmatrix} G_{i,1}(\lambda) \\ G_{i,2}(\lambda) \end{bmatrix} = G_{i,1}(\lambda) \end{aligned} \quad (5.60)$$

and thus, are parts of the (stable) co-inner TFM $G_{i,1}(\lambda)$.

Generally, $\overline{Q}(\lambda)$ contains among its poles the zeros of $G_o(\lambda)$. This is also true for the product $Q_4(\lambda)\overline{Q}(\lambda)$, where $Q_4(\lambda)$ is the stable and proper solution of the $\mathcal{H}_{2/\infty}$ -LDP. In the standard case (i.e., when $\overline{G}(\lambda)$ has no zeros in $\partial\mathbb{C}_s$), $G_o(\lambda)$ has only stable finite zeros and no infinite zeros, and therefore, $\overline{Q}(\lambda)$ results stable, provided $Q_2(\lambda)Q_1(\lambda)$ is stable. In this case, we take simply $Q_5(\lambda) = I$ and the updating factor $M(\lambda) = I$. In the nonstandard case (i.e., when $\overline{G}(\lambda)$ has zeros in $\partial\mathbb{C}_s$), the quasi-outer factor $G_o(\lambda)$ will have these zeros in $\partial\mathbb{C}_s$ too. Therefore, $\overline{Q}(\lambda)$ results unstable or improper, and we choose a diagonal, stable, proper and invertible $M(\lambda) := Q_5(\lambda)$, such that, the final $Q(\lambda)$ is proper and stable.

The computation of suitable $M(\lambda)$ can be done using LCF-based techniques as described in Sect. 9.1.6. The choice of $M(\lambda)$ can be performed such that $\|M(\lambda)\tilde{\mathcal{E}}(\lambda)\|_{2/\infty} \approx \|\tilde{\mathcal{E}}(\lambda)\|_{2/\infty}$ and $M(\lambda)$ has the least possible McMillan degree. For example, to ensure properness or strict properness, $M(\lambda)$ can be chosen diagonal with the diagonal terms $M_j(\lambda)$, $j = 1, \dots, m_f$ having the form

$$M_j(s) = \frac{1}{(\tau s + 1)^{k_j}} \quad \text{or} \quad M_j(z) = \frac{1}{z^{k_j}},$$

for continuous- or discrete-time settings, respectively. Notice that both above factors have unit \mathcal{H}_∞ -norm.

The **Procedure AMMS**, given below, formalizes the computational steps of the described synthesis method for a strongly isolable system and an invertible reference model $M_r(\lambda)$. This procedure can be also interpreted an enhanced version of **Procedure EMMS**.

Procedure AMMS: Approximate model-matching synthesis of FDI filters

Inputs : $\{G_u(\lambda), G_d(\lambda), G_w(\lambda), G_f(\lambda)\}$, invertible $M_r(\lambda)$

Outputs: $Q(\lambda), R_f(\lambda), R_w(\lambda), M(\lambda)$

- 1) Compute a minimal proper basis $Q_1(\lambda)$ for the left nullspace of $G(\lambda)$ defined in (5.2); set $Q(\lambda) = Q_1(\lambda)$ and compute

$$\left[R_f(\lambda) \middle| R_w(\lambda) \right] = Q_1(\lambda) \begin{bmatrix} G_f(\lambda) & G_w(\lambda) \\ 0 & 0 \end{bmatrix}$$

Exit if $\text{rank } R_f(\lambda) < m_f$ (no solution).

- 2) Choose $Q_2(\lambda)$ such that $Q_2(\lambda) \left[R_f(\lambda) \middle| R_w(\lambda) \right]$ has maximal full row rank and $Q_2(\lambda)Q(\lambda)$ has least McMillan degree; update $Q(\lambda) \leftarrow Q_2(\lambda)Q(\lambda)$, $R_f(\lambda) \leftarrow Q_2(\lambda)R_f(\lambda)$ and $R_w(\lambda) \leftarrow Q_2(\lambda)R_w(\lambda)$.
- 3) Compute the extended quasi-co-outer-co-inner factorization

$$\left[R_f(\lambda) \middle| R_w(\lambda) \right] = \left[G_o(\lambda) \ 0 \right] \begin{bmatrix} G_{i,1}(\lambda) \\ G_{i,2}(\lambda) \end{bmatrix}.$$

With $Q_3(\lambda) = G_o^{-1}(\lambda)$, update $Q(\lambda) \leftarrow Q_3(\lambda)Q(\lambda)$ and compute

$$\tilde{F}_1(\lambda) = \left[M_r(\lambda) \ 0 \right] G_{i,1}^{-1}(\lambda), \quad \tilde{F}_2(\lambda) = \left[M_r(\lambda) \ 0 \right] G_{i,2}^{-1}(\lambda).$$

Set $\left[R_f(\lambda) \middle| R_w(\lambda) \right] = G_{i,1}(\lambda)$.

- 4) Compute the solution $Q_4(\lambda)$ of the $\mathcal{H}_{2/\infty}$ -LDP

$$\min_{Q_4(\lambda) \in \mathcal{H}_\infty} \left\| \left[\tilde{F}_1(\lambda) - Q_4(\lambda) \tilde{F}_2(\lambda) \right] \right\|_{2/\infty};$$

update $Q(\lambda) \leftarrow Q_4(\lambda)Q(\lambda)$, $R_f(\lambda) \leftarrow Q_4(\lambda)R_f(\lambda)$ and $R_w(\lambda) \leftarrow Q_4(\lambda)R_w(\lambda)$.

- 5) Determine diagonal, stable, proper and invertible $Q_5(\lambda) := M(\lambda)$ such that $M(\lambda)Q(\lambda)$ is stable and proper; update $Q(\lambda) \leftarrow Q_5(\lambda)Q(\lambda)$, $R_f(\lambda) \leftarrow Q_5(\lambda)R_f(\lambda)$ and $R_w(\lambda) \leftarrow Q_5(\lambda)R_w(\lambda)$.

Remark 5.14 The main advantage of the **Procedure AMMS** over alternative methods, as—for example, solving $\mathcal{H}_{2/\infty}$ filter synthesis problems using standard $\mathcal{H}_{2/\infty}$

optimization procedures, lies in the possibility to easily handle frequently encountered nonstandard cases (e.g., strictly proper systems). For such a case, the standard procedures would either fail without providing any useful result, or determine unpracticable solutions (e.g., with very fast dynamics or excessively large or small gains). In contrast, the described method produces the weighting TFM $M(\lambda)$, which allows to easily obtain a suboptimal solution of a weighted problem. \square

Remark 5.15 In the case when $M_r(\lambda)$ is an $m_f \times m_f$ invertible diagonal TFM, the solution of the AMMP targets the solution of an AFDIP with a structure matrix $S = I_{m_f}$. It follows, that we can apply the threshold selection approach described in Remark 5.11 with $R_f(\lambda)$ and $R_w(\lambda)$ being $m_f \times m_f$ and, respectively, $m_f \times m_w$ TFMs. An alternative approach can be devised for the case when $M_r(\lambda)$ is a given reference model (not assumed to be structured). To account for the achieved model-matching performance, we employ instead of the residual r , the tracking error

$$\mathbf{e}(\lambda) := \mathbf{r}(\lambda) - M(\lambda)M_r(\lambda)\mathbf{f}(\lambda) = (R_f(\lambda) - M(\lambda)M_r(\lambda))\mathbf{f}(\lambda) + R_w(\lambda)\mathbf{w}(\lambda)$$

and we set the threshold $\tau_i \geq \tau_f^{(i)}$, where $\tau_f^{(i)}$ is the false alarm bound for the i -th component e_i of the tracking error defined as

$$\tau_f^{(i)} := \sup_{\substack{\|\mathbf{w}\|_2 \leq \delta_w \\ \|\mathbf{f}\|_2 \leq \delta_f}} \|\mathbf{e}_i(\lambda)\|_2.$$

As in Remark 5.11, δ_f and δ_w are the assumed bounds for the norms of the fault and noise signals, respectively. For example, τ_i can be chosen as

$$\tau_i = \|R_f^{(i)}(\lambda) - M^{(i)}(\lambda)M_r(\lambda)\|_\infty \delta_f + \|R_w^{(i)}(\lambda)\|_\infty \delta_w,$$

where $R_f^{(i)}(\lambda)$, $M^{(i)}(\lambda)$ and $R_w^{(i)}(\lambda)$ are the i -th rows of $R_f(\lambda)$, $M(\lambda)$ and $R_w(\lambda)$, respectively. The above bound can be refined along the approach used in Remark 5.11 in the case when $M_r(\lambda)$ is a structured matrix with the corresponding structure matrix S_{M_r} . \square

Example 5.16 We use the LTI system of Example 2.2 to solve a robust fault detection and isolation problem for actuator faults by employing the \mathcal{H}_∞ -norm based version of **Procedure AMMS**. The fault system in state-space form (2.2) has a standard state-space realization with $E = I$ and

$$A = \begin{bmatrix} -0.8 & 0 & 0 \\ 0 & -0.5 & 0.6 \\ 0 & -0.6 & -0.5 \end{bmatrix},$$

$$B_u = \begin{bmatrix} 1 & 1 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad B_d = 0, \quad B_w := \begin{bmatrix} 0 & 0 \\ 0 & 0.25 \\ 0.25 & 0 \end{bmatrix}, \quad B_f = \begin{bmatrix} 1 & 1 \\ 1 & 0 \\ 0 & 1 \end{bmatrix},$$

$$C = \begin{bmatrix} 0 & 1 & 1 \\ 1 & 1 & 0 \end{bmatrix}, \quad D_u = 0, \quad D_d = 0, \quad D_w = 0, \quad D_f = 0.$$

The noise input matrix B_w accounts for the effect of parametric uncertainties in the complex conjugated eigenvalues of A and is a 0.25 times scaled version of B_w derived in Example 2.2. Let $G_u(s)$, $G_d(s) = 0$, $G_w(s)$, and $G_f(s)$ denote the TFMs defined according to (2.3). The FDI filter $Q(s)$ is aimed to provide robust fault detection and isolation of actuator faults in the presence of parametric uncertainties.

At Step 1) of **Procedure AMMS**, we choose as nullspace basis

$$Q_1(s) = [I - G_u(s)] = \left[\begin{array}{c|c} sI - A & 0 - B_u \\ \hline C & I - D_u \end{array} \right]$$

and obtain $R_f(s) = G_f(s)$ and $R_w(s) = G_w(s)$. The solvability condition is: $\text{rank } R_f(s) = 2$, and thus fulfilled. Note that $R_f(s)$ is invertible and we can choose $Q_2(s) = I$ at Step 2).

At Step 3), the extended quasi-co-outer-co-inner factorization of $\bar{G}(s) = [R_f(s) \ R_w(s)]$ in (5.57) is computed. The state-space realization of the resulting $G_o(s)$ is obtained in the form (see dual version of Theorem 9.3)

$$G_o(s) = \left[\begin{array}{c|c} A - sI & \bar{B}_o \\ \hline C & \bar{D}_o \end{array} \right],$$

with

$$\bar{B}_o = \begin{bmatrix} -1.313 & -0.48 \\ -0.9334 & 0.3602 \\ -0.398 & -0.9538 \end{bmatrix}, \quad \bar{D}_o = 0.$$

Since $\bar{G}(s)$ has two zeros at infinity, $G_o(s)$ inherits these two zeros and has an additional stable zero at -1.7772 . This stable zero is also the only pole of the first-order inner factor $G_i(s) \in \mathcal{H}(s)^{4 \times 4}$. With $Q_3(s) = G_o^{-1}(s)$, the descriptor realization of the current synthesis $\bar{Q}(s) = Q_3(s)Q_2(s)Q_1(s)$ can be explicitly computed as (see (7.80) in Sect. 7.9)

$$\bar{Q}(s) = G_o^{-1}(s)Q_1(s) = \left[\begin{array}{c|c} A - sI & \bar{B}_o \\ \hline C & \bar{D}_o \end{array} \middle| \begin{array}{c|c} 0 & -B_u \\ \hline I & -D_u \end{array} \right].$$

While the current filter $\bar{Q}(s)$ is improper (having two infinite poles), the updated $R_f(s)$ and $R_w(s)$ can also be expressed according to (5.60) as $[R_f(s) \ R_w(s)] \leftarrow Q_3(s)[R_f(s) \ R_w(s)] = G_{i,1}(s)$ and are therefore, stable systems (as parts of the inner factor).

With $M_r(s) = I_2$, we compute $\tilde{F}_1(s)$ and $\tilde{F}_2(s)$ as

$$[\tilde{F}_1(s) \ \tilde{F}_2(s)] = [I \ 0][G_{i,1}(s) \ G_{i,2}(s)] = \left[\begin{array}{c|c} \tilde{A} - sI & \tilde{B}_1 \ \tilde{B}_2 \\ \hline \tilde{C} & \tilde{D}_1 \ \tilde{D}_2 \end{array} \right],$$

where

$$\begin{aligned} \tilde{A} &= 1.7772, & \tilde{B}_1 &= [-0.01688 \ -1.129], & \tilde{B}_2 &= [4.304 \ 4.754], \\ \tilde{C} &= \begin{bmatrix} 0.04136 \\ -0.1661 \end{bmatrix}, & \tilde{D}_1 &= \begin{bmatrix} -0.9090 & 0.3542 \\ -0.4035 & -0.7796 \end{bmatrix}, & \tilde{D}_2 &= \begin{bmatrix} 0.2190 & -0.0136 \\ -0.4273 & -0.2165 \end{bmatrix}. \end{aligned}$$

Both $\tilde{F}_1(s)$ and $\tilde{F}_2(s)$ are first order systems with an unstable eigenvalue at 1.7772.

At Step 4) we solve a \mathcal{H}_∞ -LDP and determine the optimal solution

$$Q_4(s) = \begin{bmatrix} -1.017 & 0.3501 \\ -0.448 & -0.7868 \end{bmatrix},$$

which leads to the current optimal synthesis $\bar{Q}(s) = Q_4(s)\bar{Q}(s)$, which is still improper. To obtain a proper and stable FDI filter $Q(s) = Q_5(s)\bar{Q}(s)$, we take at Step 5) $Q_5(s) = M(s) = \frac{10}{s+10}I_2$. The

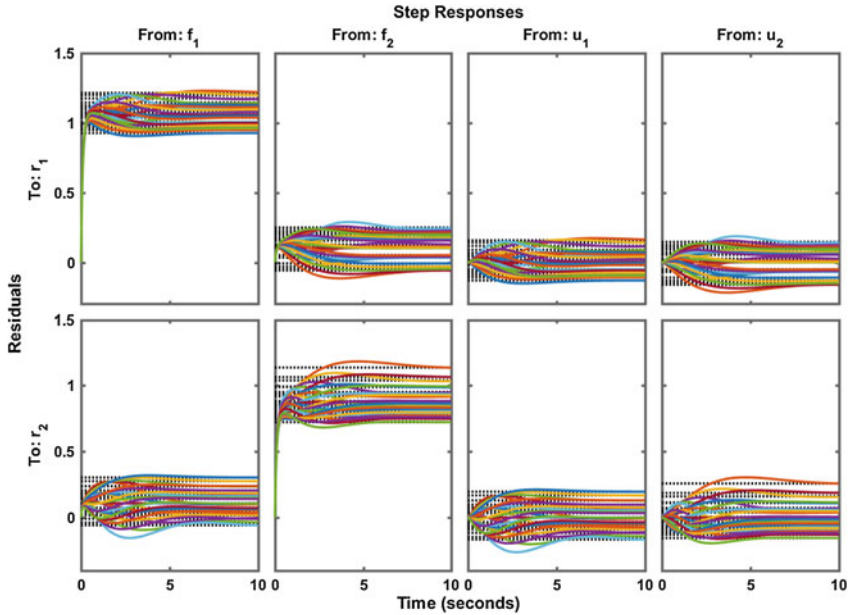


Fig. 5.2 Parametric step responses for \mathcal{H}_∞ -synthesis

resulting overall filter $Q(s)$ has order three. Note that the orders of the realizations of the individual factors $Q_1(s)$, $Q_2(s)$, $Q_3(s)$, $Q_4(s)$, and $Q_5(s)$ are respectively 2, 0, 5, 0, and 3, which sum together to 10. The corresponding (suboptimal) error norm is $\gamma_{sub} := \|M(s)\tilde{\mathcal{E}}(s)\|_\infty = 0.4521$. The minimum error norm $\gamma_{opt} := \|\bar{X}(s)\bar{G}(s) - M(s)\bar{F}(s)\|_\infty$ corresponding to the optimal improper solution $\bar{X}(s)$ (of McMillan degree 4) is $\gamma_{opt} = 0.4502$. The relatively small difference $\gamma_{sub} - \gamma_{opt} = 0.0019$ indicates that the computed $Q(s)$ is a satisfactory suboptimal proper and stable solution of the weighted problem.

We can check the robustness of the resulting $Q(s)$ by applying this FDI filter to the original system in Example 2.2 with the parameter dependent state matrix

$$A(\rho_1, \rho_2) = \begin{bmatrix} -0.8 & 0 & 0 \\ 0 & -0.5(1 + \rho_1) & 0.6(1 + \rho_2) \\ 0 & -0.6(1 + \rho_2) & -0.5(1 + \rho_1) \end{bmatrix},$$

where ρ_1 and ρ_2 take values on uniform grids with five values in their definition ranges $\rho_1 \in [-0.25, 0.25]$ and $\rho_2 \in [-0.25, 0.25]$. The simulations have been performed for all $5 \times 5 = 25$ combinations of values of ρ_1 and ρ_2 . For each combination, the step responses of the internal form of the fault detection filter have been computed. As it can be observed from Fig. 5.2, with an appropriate choice of the detection threshold, the detection and isolation of constant faults can be reliably performed in the presence of parametric uncertainties.

The script `Ex5_16` in Listing 5.7 solves the AMMP considered in this example. \diamond

Listing 5.7 Script **Ex5_16** to solve the \mathcal{H}_∞ AMMP of Example 5.16 using Procedure AMMS

```

% Uses the Control Toolbox and the Descriptor System Tools

% define system with control, noise and actuator fault inputs
A = [-.8 0 0; 0 -.5 .6; 0 -.6 -.5];
Bu = [1 1; 1 0; 0 1]; Bw = 0.25*[0 0; 0 1; 1 0]; Bf = Bu;
C = [0 1 1; 1 1 0]; Du = zeros(2,2);
% define  $G_u$ ,  $G_w$ ,  $G_f$  and  $M_r$ 
Gu = ss(A,Bu,C,0); Gw = ss(A,Bw,C,0); Gf = Gu;
Mr = ss(eye(2));
[p,mu] = size(Gu); mw = size(Gw,2); mf = size(Gf,2);

% compute left nullspace basis as  $Q_1 = [I - G_u]$ 
% initialize  $R_f = Q_1[G_f; 0]$  and  $R_w = Q_1[G_w; 0]$ 
Q1 = ss(A,[zeros(n,p) -Bu],C,[eye(p) -Du]); Rf = Gf; Rw = Gw;

% check solvability condition
if rank(evalfr(Rf,rand)) ~= mf
    error('No solution')
end

% check for unstable or infinite zeros of  $[R_f R_w]$ 
Rf_Rw = ss(A,[Bu Bw],C,0);
gzero(Rf_Rw) % two infinite zeros

tol = 1.e-7; % set tolerance

% compute the quasi-co-outer-co-inner factorization of  $[R_f R_w]$ 
[Gi,Go] = goifac(Rf_Rw,tol);

% with  $Q_3 = G_o^{-1}$  form  $\bar{Q} = Q_3 Q_1$  using explicit formulas
Qbar = dss([Go.a Go.b; Go.c Go.d],[Q1.b; Q1.d],...
    [zeros(mf,n) -eye(mf)], zeros(mf,p+mu),...
    [eye(n,n+mf); zeros(mf,n+mf)]);

% compute  $[\tilde{F}_1 \tilde{F}_2] = [M_r 0]G_i^{-1}$ 
F1_F2 = [Mr zeros(mf,mw)]*Gi';

% solve the  $H_\infty$  least distance problem  $\min \|[\tilde{F}_1 - Q_4 \tilde{F}_2]\|_\infty$ 
options = struct('tol',tol,'reltol',5.e-4);
Q4 = glinfdp(F1_F2,mw,options);
Qtilde = Q4*Qbar; % form  $\tilde{Q} = Q_4 \bar{Q}$ 

% compute stable and proper  $Q = Q_5 \tilde{Q}$  with suitable diagonal  $Q_5 = M$ 
Q = ss(zeros(0,p+mu)); M = ss(zeros(0,0));
opt_glcf = struct('tol',tol,'sdeg',-10,...
    'mindeg',true,'mininf',true);
for i=1:mf
    [Qi,Mi] = glcf(Qtilde(i,:),opt_glcf);
    % normalize  $M_i$  to unit  $H$ -infinity norm to match example
    scale = norm(Mi,inf)*sign(dcgain(Mi));
    Q = [Q;Qi/scale]; M = append(M,Mi/scale);
end

% compare suboptimal and optimal (improper) solutions
% compute  $\gamma_{sub} = \|Q[G_f G_w G_u; 0 0 I] - MM_r[I 0 0]\|_\infty$ 
G = [Gf Gw Gu; zeros(mu,mf+mw) eye(mu)];
F = M*Mr*eye(mf,mu+mw+mf);
gamma_sub = norm(Q*G-F,inf)

```

```

% compute  $\gamma_{opt} = \|Q_{opt}[G_f G_w G_u; 0 0 I] - MM_r[1 0 0]\|_\infty$ 
Yopt = glinfdp(M*F1_F2,mw,tol);
Qopt = Yopt*Qbar;
gamma_opt = norm(gir(Qopt*G-F,tol),inf)

```

Example 5.17 The model used is the same as in Example 5.16, but this time we employ the \mathcal{H}_2 -norm based version of **Procedure AMMS**. Therefore, we choose $M_r(s) = \frac{10}{s+10}I_2$ which ensures that $\|\tilde{\mathcal{E}}(s)\|_2$, the \mathcal{H}_2 -norm of the error function in (5.56), is finite. Steps 1)–3) are the same as in Example 5.16. At Step 4) the solution $Q_4(s)$ of the \mathcal{H}_2 -LDP is simply the stable part of $\tilde{F}_1(s)$

$$Q_4(s) = \left[\begin{array}{cc|cc} -10-s & 0 & -9.090 & 3.582 \\ 0 & -10-s & -4.038 & -7.955 \\ \hline 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{array} \right].$$

At Step 5) we take $Q_5(s) = M(s) = I$. The resulting FDI filter $Q(s)$ has order three. Note that the orders of the realizations of the individual factors $Q_1(s)$, $Q_2(s)$, $Q_3(s)$, $Q_4(s)$, and $Q_5(s)$ are respectively 3, 0, 5, 2, and 0, which sum together to 10. The corresponding \mathcal{H}_2 -norm of the error is $\|\tilde{\mathcal{E}}(s)\|_2 = 1.1172$, while the \mathcal{H}_∞ -norm of the error is 0.4519. It follows, that $Q(s)$ can be also interpreted as a fully satisfactory suboptimal solution of the \mathcal{H}_∞ -MMP. For the resulting filter, simulation results similar to those in Fig. 5.2 have been obtained, which indicates a satisfactory robustness of the FDI filter. \diamond

5.8 Notes and References

Section 5.1. The two computational paradigms which underly the synthesis procedures presented in this chapter have been discussed for the first time in the authors' papers [144, 151]. The factorized form (5.1) of the resulting fault detection filters is the basis of numerically reliable integrated computational algorithms. The numerical aspects of these algorithms are presented in Chap. 7. The parametrization of fault detection filters given in Theorem 5.1 extends the product form parametrization proposed in [45] given in terms of a polynomial nullspace basis. An alternative less general parametrization, without including the disturbance inputs, is presented in [31, 44]. The nullspace-based characterization of strong fault detectability in Proposition 5.2 generalizes the characterization proposed in [92] based on polynomial bases.

Section 5.2. The nullspace method (without using this naming), in a state-space based formulation, has been originally employed in [101] to solve the EFDIP using structured residuals and extended in [62] to descriptor systems. The least-order synthesis problem has been apparently addressed for the first time in [45], where a minimal polynomial basis based solution has been proposed. The application of the polynomial basis method to systems with improper TFMs is done in [93]. A numerically reliable state-space approach to the least-order synthesis relying on rational nullspace bases has been proposed in [132]. The computational details of this approach, in a state-space based setting, are discussed in Sect. 7.4. The role of

the nullspace method as a universal first computational step in all synthesis algorithms has been recognized in [151] as an important computational paradigms to address the synthesis of residual generators for a range of fault detection problems. The *sensitivity condition* (5.20) has been introduced in [48, p. 353] as a criterion to be minimized for an optimal design.

Section 5.3. To solve the AFD, $\mathcal{H}_\infty/\mathcal{H}_\infty$ optimization-based methods have been suggested by several authors, as [31, 44, 105] to cite a few of them. In this context, the H_∞ -filter based solution, advocated in [37, 38], is one of several possible synthesis methods. The $\mathcal{H}_\infty/\mathcal{H}_\infty$ optimization-based problem formulation as well as similar ones (e.g., $\mathcal{H}_2/\mathcal{H}_\infty$, $\mathcal{H}_2/\mathcal{H}_2$, etc.) have a basic difficulty in enforcing the sensitivity of residual to all fault inputs. To enhance the optimization-based formulations, the $\|\cdot\|_-$ index has been introduced in [64] as a sensitivity measure covering globally all fault inputs. Based on this definition, synthesis methods to solve the AFD have been proposed in several papers [28, 66, 77, 78, 163]. The alternative fault sensitivity measures $\|\cdot\|_{\infty-}$ and $\|\cdot\|_{2-}$ have been introduced by the author in [141], where a synthesis procedure similar to **Procedure AFD** has been also proposed. The solution of several nonstandard problems has been considered in [52]. A solution approach for the nonstandard case has been described in [28], based on a special factorization of the quasi-outer factor as a product of an outer factor and a second factor containing all zeros on the boundary of stability domain. This latter approach is implicitly contained in **Procedure AFD**, where the respective zeros are dislocated as the poles of the inverse of the quasi-outer factor using coprime factorization techniques. The extended quasi-co-outer-co-inner factorization of an arbitrary rational matrix can be computed using the dual of the algorithm of [97] for the continuous-time case and the dual of the algorithm of [94] for the discrete-time case. Specialized versions of these algorithms for proper and full column rank rational matrices are presented in Sect. 10.3.6.

Section 5.4. The solution of the EFDIP was one of the most intensively investigated problems in the fault detection literature. We only mention some of the notable works in this area, by pointing out the main achievements. Historically, of fundamental importance for a range of subsequent developments was the *geometric approach* introduced by Massoumnia [81], which was the starting point of *observer-based methods*. The main limitation of this *single filter* approach is the assumed form of the fault detection filter as a full-order Luenberger observer [80], with a suitably determined output gain matrix targeting the achievement of a desired structure matrix. The strong solvability conditions can frequently not be satisfied (no single stable filter exists), even if the FDIP has a solution. The use of a *bank of filters*, as suggested in [83], appears therefore as a natural approach to solve FDIPs for a given structure matrix. Phatak and Viswanadham proposed the use of a bank of *unknown-input observers* (UIOs) as fault detection and isolation filters [103]. Although the lack of generality of this approach is well known and ways to eliminate them have been proposed by Hou and Müller [63], the UIO-based approach preserved over the years a certain popularity (e.g., being the preferred method in [20]). The extension of the observer-based approach to the case of general proper systems has been done by Patton and Hou [101] and later extended by Hou to descriptor systems in [62].

The least-order synthesis aspect, in the most general setting, has been addressed by the author in [140] and later improved in [149], where the nullspace method has been used as a first preprocessing step to reduce the complexity of the FDIP and for designing a bank of fault detection filters to provide a set of structured residuals. This improved approach underlies the **Procedure EFDI**. Similar synthesis methods can be devised using the *parity-space approach* proposed by Chow and Willsky [22], where the least-order synthesis aspect has been discussed in [30]. The synthesis of FDI schemes based on structured residuals, also including the selection of structure matrices for suitable coding sets, has been discussed in several works of Gertler [48–50]. The nullspace-based algorithm for the efficient determination of the maximally achievable structure matrix has been proposed in [145]. This algorithm underlies the **Procedure GENSPEC**.

Section 5.5. The **Procedure AFDI** represents a refined version of the approach suggested in [151].

Section 5.6. The solution of the EMMP involves the solution of a linear rational matrix equation (see Sect. 9.1.9 for existence conditions and parametrization of all solutions). General computational algorithms, based on state-space representations, have been proposed by the author in [134, 135] and are discussed in details in Sect. 10.3.7. The *inversion-based method* to solve the EFDIP with the strong fault isolability requirement goes back to Massoumnia and Vander Velde [82], where only the case without disturbance inputs is addressed. For further extensions and discussions of this method see [31, 49, 72]. A recent development, leading to the general numerically reliable computational approach in **Procedure EMMS**, has been proposed by the author in [151].

Section 5.7. The solution of the AMMP using a \mathcal{H}_∞ or \mathcal{H}_2 optimal controller synthesis setup is the method of choice in some textbooks, see for example [14, 20]. Standard software tools for controller synthesis are available (e.g., the functions **hinfsyn** or **h2syn** available in MATLAB), but their general applicability to solve the (dual) filter synthesis problems may face difficulties. Typical bottlenecks are the assumptions on stabilizability (not fulfilled for filter synthesis for unstable plants), the lack of zeros on $\partial\mathbb{C}_s$ (typically not fulfilled if only actuator faults are considered) or the need to formulate meaningful reference models for the TFM from faults to residuals. The first two aspects can be overcome with the help of stable factorization techniques and using more general computational frameworks (e.g., *linear matrix inequalities* (LMIs) based formulations). However, in spite of some efforts (see for example, [91]), there are no clear guidelines for choosing reference models able to guarantee the existence of stable solutions. This is why, a new approach has been proposed by the author in [137], where the choice of a suitable $M_r(\lambda)$ is part of the solution procedure. This procedure has been later refined in [146, 147, 150] and **Procedure AMMS** represents its final form. The main computational ingredients of this procedure are discussed in Chap. 7, in a state-space formulation based setting.

Final note: A common aspect worth to mention regarding the proposed synthesis procedures to solve the approximate synthesis problems AFDP, AFDIP and AMMP is that the main focus in developing these algorithms lies not on solving the associated optimization problems, but on obtaining “useful” solutions of these synthesis

problems, in the most general setting and using reliable numerical techniques. Although the proposed solution approaches in [141, 146, 147, 150] follow the usual solution processes to determine the optimal solutions, still the resulting filters are usually not optimal in the nonstandard cases. The assessment of the “usefulness” of the resulting filters involves the evaluation of the actual signal bounds on the contribution of noise inputs in the residual signal and the determination of the minimum detectable amplitudes of fault signals. A solution can be considered as “useful” if it is possible to choose a suitable decision threshold which allows a robust fault monitoring without false alarms and missed fault detections. For a pertinent discussion of these aspects see [48].