

# USING MINIMAL POLYNOMIAL BASES FOR FAULT DIAGNOSIS

Erik Frisk and Mattias Nyberg

Department of Electrical Engineering, Linköping University

SE-581 83 Linköping, Sweden

Email: frisk@isy.liu.se, matny@isy.liu.se

## Abstract

A fundamental part of a fault diagnosis system is the residual generator. Design of residual generators to achieve perfect decoupling in linear systems is considered. A new method, the *minimal polynomial basis approach*, is presented, where the residual generation problem is transformed into the problem of finding polynomial bases for null-spaces of polynomial matrices. This is a standard problem in established linear systems theory, which means that numerically efficient computational tools are generally available. It is shown that the minimal polynomial basis approach can find all possible residual generators, including those of minimal degree, and the solution has a minimal parameterization.

**Keywords:** fault detection, diagnosis, polynomial bases, residual generation, perfect decoupling, disturbance decoupling

## 1 Introduction

The task of fault diagnosis is to, from known signals, i.e. measurements and control signals, detect and locate any faults acting on the system being supervised. A fundamental part of a *model based* diagnosis system is the *residual generator*. Here a new method for residual-generator design is presented, the *minimal polynomial basis approach*. It is based on finding polynomial bases for null-spaces to polynomial matrices.

An advantage with the minimal polynomial basis approach is that it can find all possible residual generators, including the ones of minimal degree, and the solution has a minimal parameterization. Many other methods, for residual generation design, do not have these properties [12]. A clear advantage with the minimal polynomial basis approach is also that, since it is based on established linear systems theory, design tools [5] are already available. Also, the algorithms that are used are well studied and have good numerical properties.

In Section 2, the fault diagnosis problem will be formulated. Thereafter, the minimal polynomial basis approach is presented in Section 3 and 4. In Section 5, the minimal

polynomial basis approach is applied to the design of a residual generator to be used in a diagnosis system for an aircraft.

## 2 Fault Diagnosis

A *model based* diagnosis system commonly consists of a *residual generator* followed by thresholds and some decision logic. The residual generator filters the known signals and generates a signal, the *residual*, that should be “small” (ideally 0) in the fault-free case and “large” when a fault is acting on the system. In Figure 1, it is illustrated how the residual generator is connected to the real system. The figure also shows that not only the control signal  $u$  influence the system, but also disturbances  $d$  and the faults  $f$  that we wish to detect. In order to *not* make the residual sensitive to the disturbances  $d$ , the disturbances must be *decoupled*.

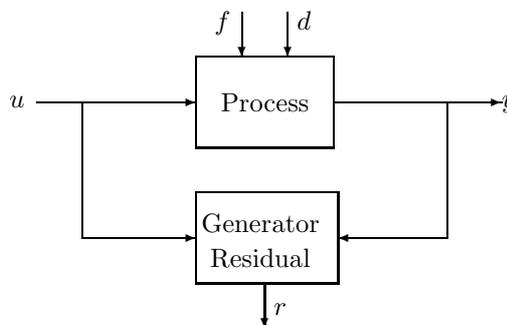


Figure 1: A residual generator.

By using several residuals, or a vector-valued residual, responding differently to different faults it is possible to achieve isolation, i.e. to distinguish between different faults. This is the basic idea of a diagnosis system using the principle of *structured residuals* [4] or the more general principle of *structured hypothesis tests* [11]. To understand the purpose of this paper, it is enough to be familiar with the principle of structured residuals which is described next.

	$f_1$	$f_2$	$f_3$
$r_1$	1	1	0
$r_2$	1	0	1
$r_3$	1	1	1

Figure 2: Example of a residual structure.

## 2.1 Structured Residuals

When *structured residuals* are used, the goal is to make different scalar residuals sensitive to different subsets of faults. What residuals that are sensitive to what faults are described by the *residual structure* [13]. An example of a residual structure is shown in Figure 2. A number 1 in the  $i$ :th row and the  $j$ :th column represents the fact that residual  $r_i$  is sensitive to fault  $f_j$ . For the example in Figure 2, it can be seen that residual  $r_1$  is sensitive to fault  $f_1$  and  $f_2$ , but not to fault  $f_3$ . The isolation is performed by matching fault columns to the actual values of the residuals. As an example, consider again Figure 2 and assume that residuals  $r_1$  and  $r_3$  become large, and  $r_2$  is small. Then it is probable that fault  $f_2$  has occurred.

To distinguish between faults that are to be decoupled and faults that the residual is made sensitive to, the terms *non-monitored faults* and *monitored faults* are used respectively. For example, if residual  $r_1$  in Figure 2 is considered, faults  $f_1$  and  $f_2$  are monitored faults and fault  $f_3$  is a non-monitored fault. From the perspective of residual generator design, there is no difference between non-monitored faults and disturbances; both must be decoupled. From now on, we will make no distinction between non-monitored faults and disturbances, and the term disturbances will also include non-monitored faults.

## 2.2 The Residual Generator

This work is a study of *linear* residual generation for *linear* systems with no model uncertainties. A general linear residual generator can be written

$$r = Q(s) \begin{pmatrix} y \\ u \end{pmatrix} \quad (1)$$

i.e.  $Q(s)$  is a multi-dimensional transfer-matrix with known signals  $y(t)$  and  $u(t)$  as inputs and a scalar *residual* as output. The requirement on the residual generator, i.e.  $Q(s)$ , is that it is sensitive to monitored faults and not sensitive to disturbances (including non-monitored faults).

A number of design methods for designing linear residual generators, have been proposed in literature, see for example [16, 19, 18, 9, 10, 1, 14]. All these methods are methods to design the transfer matrix  $Q(s)$ . Three questions that have not gained very much attention before are the following:

- Does the method find all possible residual generators?

- Does the method explicitly find residual generators of minimal McMillan degree?
- Does the solution represent a minimal parameterization or is it over parameterized?

These three questions are naturally handled by formulating the residual generation problem in the standard framework of polynomial matrices. The outcome of this is a new method, *the minimal polynomial basis approach*, presented in the following section. This approach is an intuitive solution to the residual generator problem and an additional advantage is its good numerical properties. With the minimal polynomial basis approach, it is shown that the decoupling problem is transformed into finding a minimal basis for a null-space of a polynomial matrix.

## 3 The Minimal Polynomial Basis Approach

This section introduces the minimal polynomial basis approach to the design of linear residual generators. The approach does not adopt an observer view, e.g. like the unknown input observer and eigenstructure assignment design methods. This is because the primary issues of this paper is to handle minimality and completeness of solution which are more easily addressed in a polynomial basis framework. All derivations are performed in the continuous case but the corresponding results for the time-discrete case can be obtained by substituting  $s$  by  $z$  and *improper* by *non-causal*.

### 3.1 Problem Formulation

The systems studied in this work are assumed to be on the form

$$y = G(s)u + H(s)d + L(s)f \quad (2)$$

where  $y(t)$  is measurements,  $u(t)$  is known inputs to the system,  $d(t)$  is unknown disturbances including non-monitored faults, and  $f(t)$  is the monitored faults. The filter  $Q(s)$  in (1) is a residual generator if and only if  $r(t) = 0$  for all  $d(t)$  and  $u(t)$  when  $f(t) = 0$ . To be able to detect faults, it is also required that  $r(t) \neq 0$  when  $f(t) \neq 0$ .

Inserting (2) into (1) gives

$$r = Q(s) \begin{bmatrix} G(s) & H(s) \\ I & 0 \end{bmatrix} \begin{bmatrix} u \\ d \end{bmatrix} + Q(s) \begin{bmatrix} L(s) \\ 0 \end{bmatrix} f$$

To make  $r(t) = 0$  when  $f(t) = 0$ , it is required that disturbances and the control signal are *decoupled*, i.e. for  $Q(s)$  to be a residual generator, it must hold that

$$Q(s) \begin{bmatrix} G(s) & H(s) \\ I & 0 \end{bmatrix} = 0$$

This implies that  $Q(s)$  must belong to the left null-space of

$$M(s) = \begin{bmatrix} G(s) & H(s) \\ I & 0 \end{bmatrix} \quad (3)$$

This null-space is denoted  $\mathcal{N}_L(M(s))$ . The matrix  $Q(s)$  need to fulfill two requirements: belong to the left null-space of  $M(s)$  and have good fault sensitivity properties. If, in a first step of the design, *all*  $Q(s)$  that fulfills the first requirement is found, then a  $Q(s)$  with good fault sensitivity properties can be selected. Thus, in a first step of the design of the residual generator  $Q(s)$  we need not consider  $f$  or  $L(s)$ . The problem is then to find *all* rational  $Q(s) \in \mathcal{N}_L(M(s))$ . Of special interest are the residual generators with least McMillan degree, i.e. the number of states in a minimal realization.

This can be done by finding a minimal basis for the rational vector-space  $\mathcal{N}_L(M(s))$ . A minimal basis for a rational vector-space is a *polynomial* basis [2]. In Section 4, a *computationally simple, efficient, and numerically stable* method, to find a *polynomial* basis for the left null-space of  $M(s)$  is presented. Let this basis be the rows of a matrix denoted  $N_M(s)$ . By inspection of (3), it can be realized that the dimension of  $\mathcal{N}_L(M(s))$  (i.e. the number of rows of  $N_M(s)$ ) is

$$\dim \mathcal{N}_L(M(s)) = m - \text{rank } H(s) = m - k_d$$

where  $m$  is the number of outputs, i.e. the dimension of  $y(t)$ , and  $k_d$  is the number of disturbances, i.e. the dimension of  $d(t)$ . The last equality holds only if  $\text{rank } H(s) = k_d$ , but this should be the normal case.

### 3.2 Forming the Residual Generator

When a polynomial basis  $N_M(s)$  have been obtained, the second and final step in the residual generator design is to shape fault-to-residual responses as described next.

The minimal polynomial basis  $N_M(s)$  is irreducible and because of this [6, 12], all decoupling *polynomial* vectors  $F(s)$  can be parameterized as

$$F(s) = \phi(s)N_M(s) \quad (4)$$

where  $\phi(s)$  is a polynomial vector of suitable dimensions. This parameterization vector  $\phi(s)$  can e.g. be used to shape the fault-to-residual response or simply to select one row in  $N_M(s)$ . Since  $N_M(s)$  is a basis, the parameterization vector  $\phi(s)$  have minimal number of elements, i.e. a minimal parameterization.

When a decoupling polynomial vector  $F(s)$  has been selected for implementation to form a residual generator, it must be made realizable since a polynomial vector is improper and thus not realizable. For this, we need to know the *row-degree* of  $F(s)$ , i.e. the largest polynomial degree in the row-vector  $F(s)$ . Then a realizable rational

transfer function  $Q(s)$ , i.e. the residual generator, can be found as

$$Q(s) = d_F^{-1}(s)F(s)$$

where the polynomial  $d_F(s)$  has greater or equal degree compared to the row-degree of  $F(s)$ . The degree constraint is the only constraint on  $d_F(s)$ . This means that the dynamics, i.e. poles, of the residual generator  $Q(s)$  can be chosen freely. This also means that the minimal order of a realization of a decoupling filter is determined by the row-degrees of the *minimal* polynomial basis  $N_M(s)$ .

## 4 Methods to find a Minimal Polynomial Basis to $\mathcal{N}_L(M(s))$

The problem of finding a minimal polynomial basis to the left null-space of the rational matrix  $M(s)$  can be solved by a transformation to a problem of finding a minimal polynomial basis to the left null space of a polynomial matrix. This transformation can be done in several different ways. In this section, two possibilities are demonstrated, where one is used if the model is given on transfer function form and the other if the model is given in state-space form. Also included is a short description on how to compute a basis for the null-space of a polynomial matrix.

The motivation for this transformation to a polynomial problem, is that there exists well established theory [6] regarding polynomial matrices. In addition, the generally available Polynomial Toolbox [5] for MATLAB contains an extensive set of tools for numerical handling of polynomial matrices.

The discussion will cover the general problem with disturbances included. However, in the no-disturbance case, the finding of a minimal polynomial basis to  $M(s)$  is particularly simple. Let  $\{\bar{D}_G(s), \bar{N}_G(s)\}$  be an irreducible left MFD of  $G(s)$ , i.e.  $G(s) = \bar{D}_G^{-1}(s)\bar{N}_G(s)$ . Then a minimal polynomial basis for  $M(s)$ , in the no-disturbance case, is  $N_M(s) = [\bar{D}_G(s) \quad -\bar{N}_G(s)]$  [12].

### 4.1 Frequency Domain Solution

One way of transforming the rational problem to a polynomial problem is to perform a right MFD on  $M(s)$ , i.e.

$$M(s) = \widetilde{M}_1(s)\widetilde{D}^{-1}(s) \quad (5)$$

One simple example is

$$M(s) = \widetilde{M}_1(s)d^{-1}(s)$$

where  $d(s)$  is the least common multiple of all denominators. By finding a polynomial basis for the left null-space of the *polynomial* matrix  $\widetilde{M}_1(s)$ , a basis is found also for the left null-space of  $M(s)$ . No solutions are missed because  $\widetilde{D}(s)$  (e.g.  $d(s)$ ) is of full normal rank. Thus the problem of finding a minimal polynomial basis to  $\mathcal{N}_L(M(s))$  has been transformed into finding a minimal polynomial basis to  $\mathcal{N}_L(\widetilde{M}_1(s))$ .

## 4.2 State-Space Solution

Assume that the system is described in state-space form,

$$\dot{x}(t) = Ax(t) + B_u u(t) + B_d d(t) \quad (6a)$$

$$y(t) = Cx(t) + D_u u(t) + D_d d(t) \quad (6b)$$

where  $x$  is the  $n$ -dimensional state. Then it is convenient to use the *system matrix* in state-space form [17] to find the left null-space to  $M(s)$ . The system matrix has been used before in the context of fault diagnosis, see e.g. [10, 8]. Denote the system matrix  $M_s(s)$ , describing the system with disturbances as inputs:

$$M_s(s) = \begin{bmatrix} C & D_d \\ -(sI - A) & B_d \end{bmatrix}$$

Lets define the matrix  $P$  as

$$P = \begin{bmatrix} I & -D_u \\ 0 & -B_u \end{bmatrix}$$

Then the following theorem gives a direct method on how to find a minimal polynomial basis to  $\mathcal{N}_L(M(s))$  via the system matrix.

**Theorem 1.** *Let  $V(s)$  be a minimal polynomial basis for  $\mathcal{N}_L(M_s(s))$  and let the pair  $\{A, [B_u \ B_d]\}$  be controllable. Then  $W(s) = V(s)P$  is a minimal polynomial basis for  $\mathcal{N}_L(M(s))$ .*

The proof of this theorem can be found in [3]. In conclusion, as in the previous section, the problem of finding a minimal polynomial basis to  $\mathcal{N}_L(M(s))$  has been transformed into finding a minimal polynomial basis to a polynomial matrix, in this case the system matrix  $M_s(s)$ .

## 4.3 Finding a Minimal Polynomial Basis for the null-space of a General Polynomial Matrix

The only remaining problem is how to find a minimal polynomial basis to a polynomial matrix. This is a well-known problem in the general literature on linear systems. At least two different algorithms exists. The first is based on the *Hermite form* [6] and a second algorithm is based on the *polynomial echelon form* [6]. Both methods are implemented in the Polynomial Toolbox [5]<sup>1</sup> for MATLAB and a detailed description can be found in [3].

The two algorithms have very different numerical properties. Although the algorithm based on Hermite form is easy to understand, it has poor numerical properties. However the algorithm based on polynomial echelon form is both fast and numerically stable and should therefore be the preferred choice.

<sup>1</sup>In version 1.6, the command is `xab`.

## 5 Design Example: Aircraft Dynamics

This model, taken from [7], represents a linearized model of vertical-plane dynamics of an aircraft. The inputs and outputs of the model are

Inputs	
$u_1$ :	spoiler angle [tenth of a degree]
$u_2$ :	forward acceleration [ $\text{ms}^{-2}$ ]
$u_3$ :	elevator angle [degrees]
Outputs	
$y_1$ :	relative altitude [m]
$y_2$ :	forward speed [ $\text{ms}^{-1}$ ]
$y_3$ :	Pitch angle [degrees]

The model has state-space matrices:

$$A = \begin{bmatrix} 0 & 0 & 1.132 & 0 & -1 \\ 0 & -0.0538 & -0.1712 & 0 & 0.0705 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0.0485 & 0 & -0.8556 & -1.013 \\ 0 & -0.2909 & 0 & 1.0532 & -0.6859 \end{bmatrix}$$

$$B = \begin{bmatrix} 0 & 0 & 0 \\ -0.12 & 1 & 0 \\ 0 & 0 & 0 \\ 4.419 & 0 & -1.665 \\ 1.575 & 0 & -0.0732 \end{bmatrix}$$

$$C = [I_3 \ 0] \quad D = 0_{3 \times 3}$$

Suppose the faults of interest are sensor-faults (denoted  $f_1$ ,  $f_2$ , and  $f_3$ ), and actuator-faults (denoted  $f_4$ ,  $f_5$ , and  $f_6$ ). Also, assume that the faults are modeled with additive fault models. The total model, including fault models then becomes:

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} = G(s) \left( \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} + \begin{bmatrix} f_4 \\ f_5 \\ f_6 \end{bmatrix} \right) + \begin{bmatrix} f_1 \\ f_2 \\ f_3 \end{bmatrix}$$

where  $G(s) = C(sI - A)^{-1}B + D$ . Thus, the transfer function from fault vector  $f$  to measurement vector  $y$  becomes,  $G_{yf}(s) = [I_3 \ G(s)]$ .

### 5.1 Decoupling of faults in the elevator angle actuator

The first design example is intended to illustrate the design procedure and also illustrate how available design freedom can be utilized, e.g. when selecting the residual structure but also when selecting dynamics of the residual generator.

The design example is to design a residual generator  $Q_1(s)$  that decouples faults in the elevator angle actuator, i.e.  $f_6$ . The matrix  $H(s)$  from Equation 2 correspond to all signals that are to be decoupled, i.e. considered disturbances. In this case,  $H(s)$  becomes the column in

$G_{yf}(s)$  corresponding to  $f_6$ . Matrix  $L(s)$  corresponds to the monitored faults and therefore  $L(s)$  becomes the other columns.

Since the model is given in state-space form, Theorem 1 is used to extract  $N_M(s)$ . Calculations in MATLAB give

$$N_M(s) = \begin{bmatrix} 0.0705s & s + 0.0538 & \dots \\ 22.7459s^2 + 14.5884s & -6.6653 & \dots \\ 0.091394 & 0.12 & -1 & 0 \\ s^2 - 0.93678s - 16.5141 & 31.4058 & 0 & 0 \end{bmatrix}$$

This gives that the dimension of the null-space  $\mathcal{N}_L(M(s))$  is 2, i.e. there exists two linearly independent numerators that decouples  $f_6$ . The row-degrees of the basis is 1 and 2. From this it is clear that the filter of least McMillan-degree, which decouples  $f_6$ , is a first order filter corresponding to row 1 in the basis. The polynomial  $d_F(s)$  need to have degree  $\geq 1$  to make the filter realizable since the row-degree of the first row in  $N_M(s)$  is 1. By setting  $\phi$  in (4) to  $\phi = [1 \ 0]$  and  $d_F(s)$  to  $d_F(s) = 1 + s$  the filter is made realizable and results in the following 1:st order filter.

$$Q_1(s) = \frac{1}{1+s} \begin{bmatrix} 0.0705s & s + 0.0538 & 0.091394 & \dots \\ & 0.12 & -1 & 0 \end{bmatrix} \quad (7)$$

In this example, a polynomial with degree 1 is chosen to get a minimal order realization. The cut-off frequency in  $d_F(s)$  is set so that the residual generator detects faults with energy in frequency ranges up to  $\approx 1$  rad/s.

Figure 3 shows the singular value for

$$G_d(s) = Q_1(s) \begin{bmatrix} G(s) & H(s) \\ I & 0 \end{bmatrix}$$

This plot should theoretically be exactly 0, but because of finite word length in MATLAB it doesn't become exactly 0. The plot shows that the control signals and the decoupled fault has no significant influence on the residual. Figure 4 shows how the monitored faults influence the residual which clearly shows that fault influence is significantly larger than influence from the decoupled fault and control signals plotted in Figure 3.

The leftmost plot in Figure 4 also shows that DC-gain from fault 1 to the residual is 0. Therefore are fault 1 difficult to detect since the effect in the residual of a constant fault disappears. If a fault has zero DC-gain for any residual generator, the fault is said to be *weakly detectable*. This might be expected since the system model includes an integration. Note however that an integration is neither necessary nor sufficient for a fault to be weakly detectable.

## 5.2 Decoupling several faults

As noted in the example above, the dimension of the null-space when decoupling  $f_6$  was 2. This indicates that there

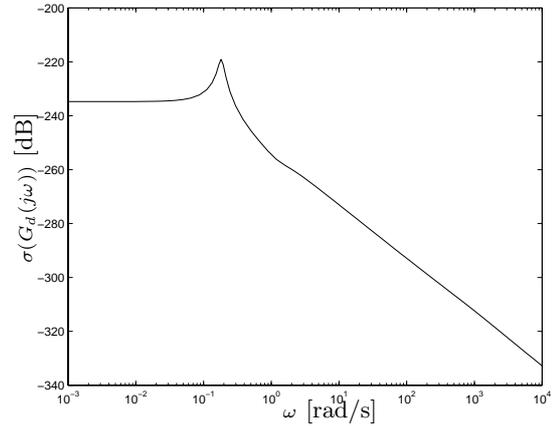


Figure 3: Singular value of the transfer function from  $u$  and  $f_6$  to  $r$ .

exists additional freedom. This freedom can e.g. be used to decouple more than one fault in each residual to shape the residual structure, e.g. to facilitate multiple-fault isolation. For example, designing a filter that decouples both  $f_1$  and  $f_4$ , i.e. faults in the first sensor and the first actuator results in a null-space of dimension 1. The polynomial  $d_F(s)$  must be at least a third order polynomial to make the filter realizable since the row-degree of the basis vector is 3. Selecting  $d_F(s)$  to be  $d_F(s) = (1 + s)^3$ , the residual generator becomes

$$Q_2(s) = \frac{1}{(1+s)^3} \begin{bmatrix} 0 \\ 36.825s^2 + 13.8953s + 1.1157 \\ s^3 + 0.61619s^2 + 4.4322s + 2.048 \\ 0 \\ -36.825s - 11.9626 \\ 1.665s - 0.28273 \end{bmatrix}^T$$

where the decoupling of faults in sensor 1 and actuator 1 is evident since  $y_1$  and  $u_1$  is not used in the calculations of the residual (compare 1).

## 6 Conclusions

Design of residual generators to achieve perfect decoupling in linear systems is considered. The goal has been to develop a design method and four issues have been addressed, namely that the method (1) is able to generate *all* possible residual generators, (2) explicitly gives the solutions with minimal McMillan degree, (3) results in a minimal parameterization of the solutions, i.e. all residual generators, and (4) has good numerical properties.

The residual generator design problem is formulated with standard notions from linear algebra and linear systems theory such as polynomial bases for rational vector spaces and it is shown that the design problem can be seen as the problem of finding *polynomial* matrices in the left

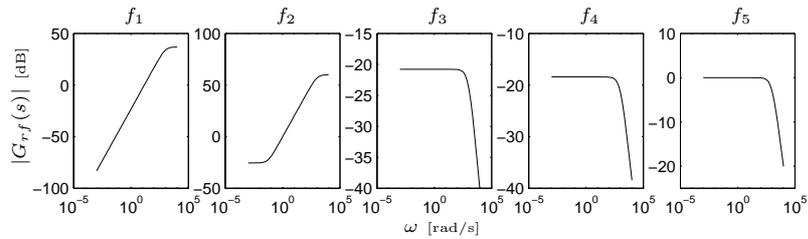


Figure 4: Magnitude bode plots for the monitored faults to the residual.

null-space of a rational matrix  $M(s)$ . Within this framework, the completeness of solution, i.e. issue (1) above, and minimality, i.e. issues (2) and (3), are naturally handled by the concept of *minimal polynomial bases*.

Finding a minimal polynomial basis for a null-space is a well-known problem and there exists computationally simple, efficient, and numerically stable algorithms, i.e. issue (4), to generate the bases. In addition, generally available implementations of these algorithms exists for MATLAB [5].

Finally, a design example shows the main steps of a design with the proposed method. The example shows how existing design freedom can be utilized and also shows how residual generators with minimal McMillan-degree is found.

## References

- [1] E.Y. Chow and A.S. Willsky. Analytical redundancy and the design of robust failure detection systems. *IEEE Trans. on Automatic Control*, 29(7):603–614, 1984.
- [2] G.D. Forney. Minimal bases of rational vector spaces, with applications to multivariable linear systems. *SIAM J. Control*, 13(3):493–520, May 1975.
- [3] E. Frisk and M. Nyberg. A description of the minimal polynomial basis approach to linear residual generation. Technical Report LiTH-R-2088, ISY, Linköping, Sweden, 1999.
- [4] J. Gertler. Analytical redundancy methods in fault detection and isolation; survey and synthesis. IFAC Fault Detection, Supervision and Safety for Technical Processes, pages 9–21, Baden-Baden, Germany, 1991.
- [5] D. Henrion, F. Kraffer, H. Kwakernaak, S. Pejchov M. Sebek, and R.C.W. Strijbos. The Polynomial Toolbox for Matlab, URL: <http://www.math.utwente.nl/polbox/> or <http://www.polyx.com>. 1997.
- [6] Thomas Kailath. *Linear Systems*. Prentice-Hall, 1980.
- [7] J.M. Maciejowski. *Multivariable Feedback Design*. Addison Wesley, 1989.
- [8] J.F. Magni and P. Mouyon. On residual generation by observer and parity space approaches. *IEEE Trans. on Automatic Control*, 39(2):441–447, 1994.
- [9] M.A. Massoumnia, G.C. Verghese, and A.S. Willsky. Failure detection and identification. *IEEE Trans. on Automatic Control*, AC-34(3):316–321, 1989.
- [10] R. Nikoukhah. Innovations generation in the presence of unknown inputs: Application to robust failure detection. *Automatica*, 30(12):1851–1867, 1994.
- [11] M. Nyberg. Framework and method for model based diagnosis with application to an automotive engine. ECC, 1997.
- [12] M. Nyberg and E. Frisk. A minimal polynomial basis solution to residual generation for fault diagnosis in linear systems. IFAC, Beijing, China, 1998.
- [13] M. Nyberg and L. Nielsen. Model based diagnosis for the air intake system of the SI-engine. *SAE Paper*, (970209), 1997.
- [14] M. Nyberg and L. Nielsen. Parity functions as universal residual generators and tool for fault detectability analysis. IEEE Conf. on Decision and Control, 1997.
- [15] R. Patton, P. Frank, and R. Clark, editors. *Fault diagnosis in Dynamic systems*. Systems and Control Engineering. Prentice Hall, 1989.
- [16] R.J. Patton and S.M. Kangethe. *Robust Fault Diagnosis using Eigenstructure Assignment of Observers*, chapter 4. In Patton et al. [15], 1989.
- [17] H.H. Rosenbrock. *State-Space and Multivariable Theory*. Wiley, New York, 1970.
- [18] J.E. White and J.L. Speyer. Detection filter design: Spectral theory and algorithms. *IEEE Trans. Automatic Control*, AC-32(7):593–603, 1987.
- [19] J. Wünnenberg. *Observer-Based Fault Detection in Dynamic Systems*. PhD thesis, University of Duisburg, 1990.