

2 Basics of Process Monitoring Techniques

As discussed in the introduction, residuals are essential and play a key role in process monitoring systems. By observing the deviation between the actual measurement and its redundancy, real-time monitoring of the status of the industrial automation processes can be performed.

During the past several decades, the design of model-based process monitoring systems has been a remarkable research topic. In contrast, the data-driven process monitoring techniques serve as an efficient alternative way, which have gained lots of attention from both the academical and the industrial field. A straightforward way is to utilize the process history data for model identification and based on it, the well-established model-based techniques can be used to design efficient fault diagnosis system. For this purpose, SIM that directly identifies the complete state-space matrices has gained more attention in the last two decades and has been successfully implemented in many industrial applications (Favoreel et al., 2000; Qin, 2006; Overschee and Moor, 1996). Parallel to the SIM, an alternative data-driven process monitoring approach is proposed in Ding et al. (2009b), which is based on the data-driven realization of the Stable Kernel Representation (SKR). Thereby the whole design procedure of the process monitoring systems becomes much simpler, easier and efficient.

The objective of this chapter is to briefly introduce the basics of process monitoring techniques, which serve as the fundamentals of this thesis.

2.1 Mathematical description of automatic control processes

2.1.1 Description of nominal system behavior

Among different system descriptions, the Linear Time-Invariant (LTI) system model is of simplest form and widely used in the theoretical study and the industrial applications. There are two standard mathematical descriptions for LTI systems: the transfer function matrix and the state-space representation.

Generally, a transfer function matrix represents the input-output dynamic behavior of an LTI system in the frequency domain. Throughout this thesis, the notation $\mathbf{G}_{\mathbf{y}\mathbf{u}}(z)$ is used for presenting the transfer function matrix of the LTI discrete-time system with the input vector $\mathbf{u} \in \mathcal{R}^l$ and the output vector $\mathbf{y} \in \mathcal{R}^m$, where z denotes the complex variable of z -transform for discrete-time signals. The input-output behavior of the dynamic system can thus be presented as:

$$\mathbf{y}(z) = \mathbf{G}_{\mathbf{y}\mathbf{u}}(z)\mathbf{u}(z). \quad (2.1)$$

The state-space representation serves as the second description of an LTI system, whose standard form of a discrete-time LTI system is given by:

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k, \quad \mathbf{x}_0, \quad (2.2)$$

$$\mathbf{y}_k = \mathbf{C}\mathbf{x}_k + \mathbf{D}\mathbf{u}_k, \quad (2.3)$$

where $\mathbf{x} \in \mathcal{R}^n$ is the state vector, $\mathbf{x}_0 \in \mathcal{R}^n$ is the initial condition of the system, and $\mathbf{u} \in \mathcal{R}^l$ is the input vector and $\mathbf{y} \in \mathcal{R}^m$ is the output vector. The subscript k is an integer indicates the discrete-time sample. The system matrices $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$ and $\mathbf{D}$ are real constant matrices with appropriate dimensions. A dynamic system with single input ($l = 1$) and single output ($m = 1$) is called a Single-Input Single-Output (SISO) system. Similarly, Single-Input Multi-Output (SIMO) system, Multi-Input Single-Output (MISO) system, as well as Multi-Input Multi-Output (MIMO) system can be defined.

The state-space model can either be directly achieved by modelling or derived based on a transfer matrix. The corresponding transfer function matrix is:

$$\mathbf{G}_{\mathbf{y}\mathbf{u}}(z) = \mathbf{C}(z\mathbf{I} - \mathbf{A})^{-1}\mathbf{B} + \mathbf{D}. \quad (2.4)$$

For the sake of simplicity, the following notation is used throughout this thesis:

$$\left[\begin{array}{c|c} \mathbf{A} & \mathbf{B} \\ \hline \mathbf{C} & \mathbf{D} \end{array} \right] := \mathbf{C}(z\mathbf{I} - \mathbf{A})^{-1}\mathbf{B} + \mathbf{D}$$

Since for a given industrial automation process there are infinite state-space realizations (not necessarily the same dimension), in this thesis, only the minimal realizations (Chen, 2013) are considered. Namely, $(\mathbf{A}, \mathbf{B})$ is controllable and $(\mathbf{A}, \mathbf{C})$ is observable.

2.1.2 Coprime factorization technique

Coprime factorization provides an alternative representation of system transfer function matrix. Generally speaking, coprime factorization over $\mathcal{RH}_\infty$ is a factorization of system transfer matrix by two proper and real-rational stable transfer matrices.

Definition 2.1: Two transfer matrices $\hat{\mathbf{M}}(z)$ and $\hat{\mathbf{N}}(z)$ in $\mathcal{RH}_\infty$ are called *left coprime* over $\mathcal{RH}_\infty$ if they have the same number of rows and if there exist transfer matrices $\hat{\mathbf{X}}(z)$ and $\hat{\mathbf{Y}}(z)$ in $\mathcal{RH}_\infty$ such that

$$\left[\begin{array}{cc} \hat{\mathbf{M}}(z) & \hat{\mathbf{N}}(z) \end{array} \right] \left[\begin{array}{c} \hat{\mathbf{X}}(z) \\ \hat{\mathbf{Y}}(z) \end{array} \right] = \mathbf{I}.$$

Similarly, two transfer matrices $\mathbf{M}(z)$ and $\mathbf{N}(z)$ in $\mathcal{RH}_\infty$ are called *right coprime* over $\mathcal{RH}_\infty$ if they have the same number of columns and if there exist transfer matrices $\mathbf{X}(z)$ and $\mathbf{Y}(z)$ in $\mathcal{RH}_\infty$ such that

$$\left[\begin{array}{cc} \mathbf{X}(z) & \mathbf{Y}(z) \end{array} \right] \left[\begin{array}{c} \mathbf{M}(z) \\ \mathbf{N}(z) \end{array} \right] = \mathbf{I}.$$

Definition 2.2: Let $\mathbf{G}(z)$ be a proper real-rational transfer function matrix. The *Left Coprime Factorization (LCF)* of $\mathbf{G}(z)$ is a factorization $\mathbf{G}(z) = \hat{\mathbf{M}}^{-1}(z)\hat{\mathbf{N}}(z)$ where $\hat{\mathbf{M}}(z)$ and $\hat{\mathbf{N}}(z)$ are left coprime over $\mathcal{RH}_\infty$. Similarly, the *Right Coprime Factorization (RCF)* of $\mathbf{G}(z)$ is a factorization $\mathbf{G}(z) = \mathbf{N}(z)\mathbf{M}^{-1}(z)$ where $\mathbf{M}(z)$ and $\mathbf{N}(z)$ are right coprime over $\mathcal{RH}_\infty$.

Below, a lemma (Zhou, 1996) for the state-space computation of LCF and RCF is given, which forms the foundation of the subsequent studies.

Lemma 2.1: Suppose $\mathbf{G}_{\mathbf{y}\mathbf{u}}(z)$ is a proper real-rational transfer function matrix with a minimal state-space realization

$$\mathbf{G}_{\mathbf{y}\mathbf{u}}(z) = \left[\begin{array}{c|c} \mathbf{A} & \mathbf{B} \\ \hline \mathbf{C} & \mathbf{D} \end{array} \right].$$

Let $\mathbf{F}$ and $\mathbf{L}$ be such that $\mathbf{A} + \mathbf{B}\mathbf{F}$ and $\mathbf{A} - \mathbf{L}\mathbf{C}$ are both stable, and define

$$\left[\begin{array}{cc} \mathbf{M}(z) & -\hat{\mathbf{Y}}(z) \\ \mathbf{N}(z) & \hat{\mathbf{X}}(z) \end{array} \right] = \left[\begin{array}{cc|cc} \mathbf{A} + \mathbf{B}\mathbf{F} & \mathbf{B} & \mathbf{L} & \\ \hline \mathbf{F} & \mathbf{I} & \mathbf{0} & \\ \mathbf{C} + \mathbf{D}\mathbf{F} & \mathbf{D} & \mathbf{I} & \end{array} \right], \quad (2.5)$$

$$\left[\begin{array}{cc} \mathbf{X}(z) & \mathbf{Y}(z) \\ -\hat{\mathbf{N}}(z) & \hat{\mathbf{M}}(z) \end{array} \right] = \left[\begin{array}{cc|cc} \mathbf{A} - \mathbf{L}\mathbf{C} & -(\mathbf{B} - \mathbf{L}\mathbf{D}) & -\mathbf{L} & \\ \hline \mathbf{F} & \mathbf{I} & \mathbf{0} & \\ \mathbf{C} & -\mathbf{D} & \mathbf{I} & \end{array} \right]. \quad (2.6)$$

Then

$$\mathbf{G}_{\mathbf{y}\mathbf{u}}(z) = \hat{\mathbf{M}}^{-1}(z)\hat{\mathbf{N}}(z) = \mathbf{N}(z)\mathbf{M}^{-1}(z) \quad (2.7)$$

are the LCF and RCF of $\mathbf{G}_{\mathbf{y}\mathbf{u}}(z)$, respectively. Moreover, the so-called Bezout identity holds

$$\left[\begin{array}{cc} \mathbf{X}(z) & \mathbf{Y}(z) \\ -\hat{\mathbf{N}}(z) & \hat{\mathbf{M}}(z) \end{array} \right] \left[\begin{array}{cc} \mathbf{M}(z) & -\hat{\mathbf{Y}}(z) \\ \mathbf{N}(z) & \hat{\mathbf{X}}(z) \end{array} \right] = \left[\begin{array}{cc} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{array} \right]. \quad (2.8)$$

2.1.3 Description of systems with disturbances

In reality, the system is always subject to unknown inputs caused by, *e.g.* exogenous disturbances, measurement and process noises, etc. In order to consider the effects of the unknown inputs, the system input-output model (2.2)-(2.3) is extended as follows:

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k + \mathbf{E}_d\mathbf{d}_k + \boldsymbol{\xi}_k, \quad \mathbf{x}_0, \quad (2.9)$$

$$\mathbf{y}_k = \mathbf{C}\mathbf{x}_k + \mathbf{D}\mathbf{u}_k + \mathbf{F}_d\mathbf{d}_k + \mathbf{v}_k. \quad (2.10)$$

where $\mathbf{E}_d$ and $\mathbf{F}_d$ are disturbance distribution matrices of compatible dimensions, $\mathbf{d} \in \mathcal{R}^{k_d}$ represents the deterministic unknown input vector, $\boldsymbol{\xi} \in \mathcal{R}^n$ and $\mathbf{v} \in \mathcal{R}^m$ denote the process and measurement noise sequence, that are assumed to be normally distributed, white and statistically independent of $\mathbf{u}$ and $\mathbf{x}$.

2.1.4 Description of systems with faults

The faults in the LTI systems could be modeled in several ways. One widely adopted modeling of the additive faults is to extend the system model in Eqs.(2.9)-(2.10) to:

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k + \mathbf{E}_d\mathbf{d}_k + \boldsymbol{\xi}_k + \mathbf{E}_f\mathbf{f}_k, \quad \mathbf{x}_0, \quad (2.11)$$

$$\mathbf{y}_k = \mathbf{C}\mathbf{x}_k + \mathbf{D}\mathbf{u}_k + \mathbf{F}_d\mathbf{d}_k + \mathbf{v}_k + \mathbf{F}_f\mathbf{f}_k. \quad (2.12)$$

where $\mathbf{f} \in \mathcal{R}^{k_f}$ is the additive fault vector that are independent of $\mathbf{u}$ and $\mathbf{x}$, while $\mathbf{E}_f$ and $\mathbf{F}_f$ are fault distribution matrices of appropriate dimensions. Generally, by choosing

proper distribution matrices, *i.e.* $\mathbf{E}_f$ and $\mathbf{F}_f$, different additive faults in the system can be represented. In case of the fault vector $\mathbf{f}$ is a function of the system state and input variables, the above representation could also describe multiplicative faults and the system stability may be affected. Another common description of the multiplicative fault is to use system parameter changes, which is shown as follows:

$$\mathbf{x}_{k+1} = (\mathbf{A} + \mathbf{A}_f)\mathbf{x}_k + (\mathbf{B} + \mathbf{B}_f)\mathbf{u}_k + \mathbf{E}_d\mathbf{d}_k + \boldsymbol{\xi}_k, \quad \mathbf{x}_0, \quad (2.13)$$

$$\mathbf{y}_k = (\mathbf{C} + \mathbf{C}_f)\mathbf{x}_k + (\mathbf{D} + \mathbf{D}_f)\mathbf{u}_k + \mathbf{F}_d\mathbf{d}_k + \mathbf{v}_k. \quad (2.14)$$

where $\mathbf{A}_f$, $\mathbf{B}_f$, $\mathbf{C}_f$ and $\mathbf{D}_f$ represent the faults on the system matrices.

2.2 Model-based residual generation techniques

Inspired by the early work of Beard (1971) and Jones (1973), model-based process monitoring techniques have been remarkably developed. A great number of industrial applications of the model-based process monitoring techniques have been carried out and their efficiency for fault diagnosis has been abundantly demonstrated (Frank, 1990; Gertler, 1998; Chen and Patton, 1999; Patton et al., 2000; Blanke et al., 2006; Ding, 2013). Among the well-developed model-based process monitoring schemes, the Fault Detection Filter (FDF), Diagnostic Observer (DO) and Parity Space (PS) based residual generation schemes have received great attention during last two decades. Brief introductions of the related techniques will be included in this section.

2.2.1 Kernel representation and fault detection filter

Definition 2.3: Given system (2.2)-(2.3), a stable linear system $\mathcal{K}$ driven by $\mathbf{u}(z)$, $\mathbf{y}(z)$ and satisfying

$$\forall \mathbf{u}(z), \quad \mathbf{r}(z) = \mathcal{K} \begin{bmatrix} \mathbf{u}(z) \\ \mathbf{y}(z) \end{bmatrix} = \mathbf{0} \quad (2.15)$$

is called *SKR* of (2.2)-(2.3).

As initially proposed in Beard (1971) and Jones (1973), FDF is the first type of observer-based residual generator for the purpose of FDI. Considering the LTI system described by Eqs.(2.11)-(2.12), a full-order state observer could be realized as:

$$\hat{\mathbf{x}}_{k+1} = \mathbf{A}\hat{\mathbf{x}}_k + \mathbf{B}\mathbf{u}_k + \mathbf{L}(\mathbf{y}_k - \hat{\mathbf{y}}_k), \quad \hat{\mathbf{x}}_0, \quad (2.16)$$

$$\hat{\mathbf{y}}_k = \mathbf{C}\hat{\mathbf{x}}_k + \mathbf{D}\mathbf{u}_k, \quad (2.17)$$

$$\mathbf{r}_k = \mathbf{y}_k - \hat{\mathbf{y}}_k, \quad (2.18)$$

where the matrix $\mathbf{L}$ is the so-called observer gain such that $\mathbf{A} - \mathbf{L}\mathbf{C}$ being Schur matrix, *i.e.* a square matrix with real entries and with eigenvalues of absolute value less than one. $\mathbf{r}$ is called the residual signal. Recall the LCF of the system $\mathbf{G}_{\mathbf{y}\mathbf{u}}(z) = \hat{\mathbf{M}}^{-1}(z)\hat{\mathbf{N}}(z)$. Following the definition given in (2.6), it is easy to see that, $\hat{\mathbf{M}}(z)$ and $\hat{\mathbf{N}}(z)$ respectively correspond to the transfer matrices from $\mathbf{r}_k$ to $\mathbf{y}_k$ and $\mathbf{u}_k$. Therefore, in the fault- and disturbance-free case

$$\mathbf{r}(z) = \begin{bmatrix} -\hat{\mathbf{N}}(z) & \hat{\mathbf{M}}(z) \end{bmatrix} \begin{bmatrix} \mathbf{u}(z) \\ \mathbf{y}(z) \end{bmatrix} = \mathbf{y}(z) - \hat{\mathbf{y}}(z) = \mathbf{0}, \quad (2.19)$$

which indicates that the LCF of the system $\mathbf{G}_{\mathbf{y}\mathbf{u}}(z)$ can be realized as an FDF. Furthermore, the stable linear system $\begin{bmatrix} -\hat{\mathbf{N}}(z) & \hat{\mathbf{M}}(z) \end{bmatrix}$ forms an SKR of the system $\mathbf{G}_{\mathbf{y}\mathbf{u}}(z)$.

Consider the case when disturbance and fault are involved, by introducing the estimation error of the state variable, $\mathbf{e}_k = \mathbf{x}_k - \hat{\mathbf{x}}_k$, the dynamics of the FDF becomes:

$$\mathbf{e}_{k+1} = (\mathbf{A} - \mathbf{L}\mathbf{C})\mathbf{e}_k + (\mathbf{E}_d - \mathbf{L}\mathbf{F}_d)\mathbf{d}_k + (\mathbf{E}_f - \mathbf{L}\mathbf{F}_f)\mathbf{f}_k + \boldsymbol{\xi}_k - \mathbf{L}\mathbf{v}_k, \quad \mathbf{e}_0, \quad (2.20)$$

$$\mathbf{r}_k = \mathbf{C}\mathbf{e}_k + \mathbf{F}_d\mathbf{d}_k + \mathbf{F}_f\mathbf{f}_k + \mathbf{v}_k. \quad (2.21)$$

Note that in the fault- and disturbance-free case, $\lim_{k \rightarrow \infty} \mathbf{e}_k = 0$ due to $\mathbf{A} - \mathbf{L}\mathbf{C}$ being Schur matrix which implies $\lim_{k \rightarrow \infty} \mathbf{r}_k = 0$. When a fault happens, $\mathbf{r}_k \neq 0$ can be used to indicate the occurrence of the fault. However, in actual industrial practice, the disturbances are inevitable which means $\mathbf{r}_k \neq 0$ can not be easily used to make any decision. To tackle this problem, the residual generator can be extended to

$$\mathbf{r}_k = \mathbf{V}(\mathbf{y}_k - \hat{\mathbf{y}}_k) \quad (2.22)$$

by introducing the so-called post-filter $\mathbf{V}$, which can be designed to increase the sensitivity to faults and enhance the robustness against disturbances.

The full-order state observer serves as the core of the FDF, whose online computational cost is much more expensive than a reduced order observer. In contrast, a reduced order observer can provide similar estimation performance but with much less online computation. This is one of the motivations to develop Luenberger type residual generators.

2.2.2 Diagnostic observer

The DO is one of the mostly studied model-based residual generators due to its flexible structure and similarity to the Luenberger type (output) observer. The core of a DO is a Luenberger type (output) observer that is described by

$$\mathbf{x}_{\mathbf{o},k+1} = \mathbf{G}\mathbf{x}_{\mathbf{o},k} + \mathbf{H}\mathbf{u}_k + \mathbf{L}\mathbf{y}_k, \quad \mathbf{x}_{\mathbf{o},0}, \quad (2.23)$$

$$\hat{\mathbf{y}}_k = \bar{\mathbf{W}}\mathbf{x}_{\mathbf{o},k} + \bar{\mathbf{V}}\mathbf{y}_k + \bar{\mathbf{Q}}\mathbf{u}_k \quad (2.24)$$

where the state vector of the observer $\mathbf{x}_{\mathbf{o}} \in \mathcal{R}^s$, s denotes the observer order and it could be different from the system order n . The matrices $\mathbf{G}, \mathbf{H}, \mathbf{L}, \bar{\mathbf{W}}, \bar{\mathbf{V}}$ and $\bar{\mathbf{Q}}$ together with a matrix, $\mathbf{T} \in \mathcal{R}^{s \times n}$, have to satisfy the following Luenberger conditions,

$$\text{I.} \quad \mathbf{G} \text{ is stable,} \quad (2.25)$$

$$\text{II.} \quad \mathbf{T}\mathbf{A} - \mathbf{G}\mathbf{T} = \mathbf{L}\mathbf{C}, \quad \mathbf{H} = \mathbf{T}\mathbf{B} - \mathbf{L}\mathbf{D}, \quad (2.26)$$

$$\text{III.} \quad \mathbf{C} = \bar{\mathbf{W}}\mathbf{T} + \bar{\mathbf{V}}\mathbf{C}, \quad \bar{\mathbf{Q}} = \mathbf{D} - \bar{\mathbf{V}}\mathbf{D}, \quad (2.27)$$

under which, the system described by Eqs.(2.23)-(2.24) provides an unbiased estimation for output $\mathbf{y}$, *i.e.*

$$\lim_{k \rightarrow \infty} (\mathbf{y}_k - \hat{\mathbf{y}}_k) = 0. \quad (2.28)$$

Considering the LTI system described by Eqs.(2.11)-(2.12) and denoting the error vector $\mathbf{e}_k = \mathbf{T}\mathbf{x}_k - \mathbf{x}_{\mathbf{o},k}$, then the error dynamics becomes

$$\mathbf{e}_{k+1} = \mathbf{G}\mathbf{e}_k + (\mathbf{T}\mathbf{E}_d - \mathbf{L}\mathbf{F}_d)\mathbf{d}_k + (\mathbf{T}\mathbf{E}_f - \mathbf{L}\mathbf{F}_f)\mathbf{f}_k + \mathbf{T}\boldsymbol{\xi}_k - \mathbf{L}\mathbf{v}_k, \quad \mathbf{e}_0, \quad (2.29)$$

$$\mathbf{r}_k = \bar{\mathbf{W}}\mathbf{e}_k + (\mathbf{F}_d - \bar{\mathbf{V}}\mathbf{F}_d)\mathbf{d}_k + (\mathbf{F}_f - \bar{\mathbf{V}}\mathbf{F}_f)\mathbf{f}_k + (\mathbf{I} - \bar{\mathbf{V}})\mathbf{v}_k, \quad (2.30)$$

which ensures (2.28) in the fault- and disturbance-free case. To increase the degree of design freedom, a post-filter $\mathbf{V}^*$ is introduced

$$\mathbf{r}_k = \mathbf{V}^*(\mathbf{y}_k - \hat{\mathbf{y}}_k), \quad (2.31)$$

which provides a residual vector, whose dynamics can be described by

$$\mathbf{x}_{\mathbf{o},k+1} = \mathbf{G}\mathbf{x}_{\mathbf{o},k} + \mathbf{H}\mathbf{u}_k + \mathbf{L}\mathbf{y}_k, \quad \mathbf{x}_{\mathbf{o},0}, \quad (2.32)$$

$$\mathbf{r}_k = \mathbf{V}\mathbf{y}_k - \mathbf{W}\mathbf{x}_{\mathbf{o},k} - \mathbf{Q}\mathbf{u}_k \quad (2.33)$$

where

$$\mathbf{V} = \mathbf{V}^*(\mathbf{I} - \bar{\mathbf{V}}), \quad \mathbf{W} = \mathbf{V}^*\bar{\mathbf{W}}, \quad \mathbf{Q} = \mathbf{V}^*\bar{\mathbf{Q}}.$$

Therefore, for residual generation, the third Luenberger condition (2.27) shall be replaced by

$$\text{III.} \quad \mathbf{VC} - \mathbf{WT} = \mathbf{0}, \quad \mathbf{Q} = \mathbf{VD}. \quad (2.34)$$

In comparison with FDF scheme introduced in the last subsection, DO scheme may lead to a reduced order residual generator with less on-line computation, which is desirable and useful for online implementation.

2.2.3 Parity space approach

Initiated by the early work of Chow and Willsky (1984), the parity space approach has received much attention in last two decades. The parity space approach is generally recognized as one of the most important model-based residual generation approaches, which results an easy way for FDI.

Consider the LTI system described by Eqs.(2.11)-(2.12). In order to construct the residual generator, for a given parity space of order s , the system can be expressed as follows:

$$\mathbf{y}_{s,k} = \mathbf{\Gamma}_s \mathbf{x}_k + \mathbf{H}_{u,s} \mathbf{u}_{s,k} + \mathbf{H}_{d,s} \mathbf{d}_{s,k} + \mathbf{H}_{f,s} \mathbf{f}_{s,k} + \mathbf{H}_{\xi,s} \boldsymbol{\xi}_{s,k} + \mathbf{v}_{s,k} \quad (2.35)$$

where $\mathbf{y}_{s,k}$, $\mathbf{u}_{s,k}$, $\mathbf{d}_{s,k}$, $\mathbf{f}_{s,k}$, $\boldsymbol{\xi}_{s,k}$ and $\mathbf{v}_{s,k}$ are constructed as $\boldsymbol{\lambda}_{s,k}$ with the following data structure:

$$\boldsymbol{\lambda}_{s,k} = \begin{bmatrix} \lambda_k \\ \lambda_{k+1} \\ \vdots \\ \lambda_{k+s} \end{bmatrix} \in \mathcal{R}^{(s+1)k\lambda}, \quad \lambda_k \in \mathcal{R}^{k\lambda} \quad (2.36)$$

and

$$\mathbf{\Gamma}_s = \begin{bmatrix} \mathbf{C} \\ \mathbf{CA} \\ \vdots \\ \mathbf{CA}^s \end{bmatrix}, \quad \mathbf{H}_{u,s} = \begin{bmatrix} \mathbf{D} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{CB} & \mathbf{D} & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{CA}^{s-1}\mathbf{B} & \cdots & \mathbf{CB} & \mathbf{D} \end{bmatrix}, \quad (2.37)$$

$$\mathbf{H}_{d,s} = \begin{bmatrix} \mathbf{F}_d & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{CE}_d & \mathbf{F}_d & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{CA}^{s-1}\mathbf{E}_d & \cdots & \mathbf{CE}_d & \mathbf{F}_d \end{bmatrix}, \quad \mathbf{H}_{f,s} = \begin{bmatrix} \mathbf{F}_f & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{CE}_f & \mathbf{F}_f & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{CA}^{s-1}\mathbf{E}_f & \cdots & \mathbf{CE}_f & \mathbf{F}_f \end{bmatrix}, \quad (2.38)$$

$$\mathbf{H}_{\xi,s} = \begin{bmatrix} \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{C} & \mathbf{0} & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{CA}^{s-1} & \cdots & \mathbf{C} & \mathbf{0} \end{bmatrix}. \quad (2.39)$$

Since $\mathbf{\Gamma}_s \in \mathcal{R}^{m(s+1) \times n}$, for $s \geq n$, the following rank condition holds:

$$\text{rank}(\mathbf{\Gamma}_s) = n \quad (2.40)$$

which implies that there exists at least a row vector $\mathbf{v}_s (\neq 0) \in \mathcal{R}^{1 \times m(s+1)}$ such that

$$\mathbf{v}_s \mathbf{\Gamma}_s = \mathbf{0}. \quad (2.41)$$

The vectors satisfying Eq.(2.41) are called parity vectors, the set of the parity vectors, *i.e.*

$$\mathbf{P}_s = \{ \mathbf{v}_s \mid \mathbf{v}_s \mathbf{\Gamma}_s = \mathbf{0} \} \quad (2.42)$$

is called the parity space of the s -th order. Consequently, a parity relation based residual generator can be constructed by

$$\mathbf{r}_k = \mathbf{v}_s (\mathbf{y}_{s,k} - \mathbf{H}_{u,s} \mathbf{u}_{s,k}). \quad (2.43)$$

Therefore, considering (2.35), in the fault- and disturbance-free case, the residual signal becomes

$$\mathbf{r}_k = \mathbf{v}_s (\mathbf{y}_{s,k} - \mathbf{H}_{u,s} \mathbf{u}_{s,k}) = \mathbf{v}_s \mathbf{\Gamma}_s \mathbf{x}_k = 0.$$

In case that the system is corrupted by faults and disturbances, the dynamics of the residual signal is given by

$$\mathbf{r}_k = \mathbf{v}_s (\mathbf{H}_{d,s} \mathbf{d}_{s,k} + \mathbf{H}_{f,s} \mathbf{f}_{s,k} + \mathbf{H}_{\xi,s} \xi_{s,k} + \mathbf{v}_{s,k}), \quad \mathbf{v}_s \in \mathbf{P}_s. \quad (2.44)$$

The main task of the parity relation based residual generator is to select the orthogonal subspace of the subspace spanned by the columns of $\mathbf{\Gamma}_s$. However, its online implementation requires not only the temporal but also the past input and measurement data that have to be recorded. In actual industrial practice, the parity vector $\mathbf{v}_s$ is expected to be selected in such a way that the residual signal is completely decoupled from the disturbances, while the fault information is fully preserved. If the perfect decoupling is not feasible, then the parity vector should be selected such that the influence of the disturbances on the residual signal is minimized while the influence of the fault on the residual signal is maximized.

2.2.4 Interconnections between DO and PS schemes

It has been proven in Ding et al. (1999) and Ding (2013) that there exists an one-to-one mapping between the design parameters of DO and parity relation based residual generators. The interconnections are shown through the following two lemmata (Ding, 2013).

Lemma 2.2: *Given system model (2.11)-(2.12) and a parity vector $\mathbf{v}_s = [\mathbf{v}_{s,0} \ \mathbf{v}_{s,1} \ \cdots \ \mathbf{v}_{s,s}]$, then matrices $\mathbf{G}, \mathbf{T}, \mathbf{L}, \mathbf{H}, \mathbf{q}, \mathbf{v}, \mathbf{w}$ defined by*

$$\begin{aligned} \mathbf{G} &= [\mathbf{G}_0 \ \mathbf{g}], \quad \mathbf{G}_0 = \begin{bmatrix} 0 & 0 & \cdots & 0 \\ 1 & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & 1 & 0 \\ 0 & \cdots & 0 & 1 \end{bmatrix} \in \mathcal{R}^{s \times (s-1)}, \quad \mathbf{g} = \begin{bmatrix} g_1 \\ \vdots \\ g_s \end{bmatrix} \in \mathcal{R}^s, \\ \mathbf{T} &= \begin{bmatrix} \mathbf{v}_{s,1} & \mathbf{v}_{s,2} & \cdots & \mathbf{v}_{s,s-1} & \mathbf{v}_{s,s} \\ \mathbf{v}_{s,2} & \cdots & \cdots & \mathbf{v}_{s,s} & 0 \\ \vdots & \cdots & \cdots & \vdots & \vdots \\ \mathbf{v}_{s,s} & 0 & \cdots & \cdots & 0 \end{bmatrix} \begin{bmatrix} \mathbf{C} \\ \mathbf{CA} \\ \vdots \\ \mathbf{CA}^{s-1} \end{bmatrix}, \quad \mathbf{L} = - \begin{bmatrix} \mathbf{v}_{s,0} \\ \mathbf{v}_{s,1} \\ \vdots \\ \mathbf{v}_{s,s-1} \end{bmatrix} - \mathbf{g}\mathbf{v}_{s,s}, \\ \mathbf{H} &= \begin{bmatrix} \mathbf{v}_{s,0} + g_1\mathbf{v}_{s,s} & \mathbf{v}_{s,1} & \cdots & \cdots & \mathbf{v}_{s,s} \\ \mathbf{v}_{s,1} + g_2\mathbf{v}_{s,s} & \mathbf{v}_{s,2} & \cdots & \mathbf{v}_{s,s} & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ \mathbf{v}_{s,s-1} + g_s\mathbf{v}_{s,s} & \mathbf{v}_{s,s} & 0 & \cdots & 0 \end{bmatrix} \begin{bmatrix} \mathbf{D} \\ \mathbf{CB} \\ \vdots \\ \mathbf{CA}^{s-1}\mathbf{B} \end{bmatrix}, \\ \mathbf{q} &= \mathbf{v}_{s,s}\mathbf{D}, \quad \mathbf{v} = \mathbf{v}_{s,s}, \quad \mathbf{w} = [0 \ \cdots \ 0 \ 1] \in \mathcal{R}^s, \end{aligned}$$

in which $\mathbf{g}$ ensures the stability of $\mathbf{G}$, satisfy the Luenberger conditions (2.25)-(2.26),(2.34).

Lemma 2.3: *Given system model (2.11)-(2.12) and observer-based residual generator (2.32)-(2.33) with matrices $\mathbf{G}, \mathbf{T}, \mathbf{L}, \mathbf{H}, \mathbf{q}, \mathbf{v}, \mathbf{w}$ satisfying the Luenberger conditions (2.25)-(2.26),(2.34) while $\mathbf{G}, \mathbf{w}$ are of following form*

$$\begin{aligned} \mathbf{G} &= [\mathbf{G}_0 \ \mathbf{g}], \quad \mathbf{G}_0 = \begin{bmatrix} 0 & 0 & \cdots & 0 \\ 1 & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & 1 & 0 \\ 0 & \cdots & 0 & 1 \end{bmatrix} \in \mathcal{R}^{s \times (s-1)}, \\ \mathbf{g} &= \begin{bmatrix} g_1 \\ \vdots \\ g_s \end{bmatrix} \in \mathcal{R}^s, \quad \mathbf{w} = [0 \ \cdots \ 0 \ 1] \in \mathcal{R}^s. \end{aligned}$$

Then vector $\mathbf{v}_s = [\mathbf{v}_{s,0} \ \mathbf{v}_{s,1} \ \cdots \ \mathbf{v}_{s,s}]$ with

$$\mathbf{v}_{s,s} = \mathbf{v}, \quad \begin{bmatrix} \mathbf{v}_{s,0} \\ \mathbf{v}_{s,1} \\ \vdots \\ \mathbf{v}_{s,s-1} \end{bmatrix} = -\mathbf{L} - \mathbf{g}\mathbf{v}$$

belongs to the parity space $\mathbf{P}_s$.

The above lemmata reveal the one-to-one relationship between PS and DO schemes. It is worth to notice that, the FDF and DO based residual generators are of closed-loop configuration, since a feedback of the residual signal is embedded in the system realization and the computation is realized in a recursive manner. In contrast, the PS-based residual generator is in an open-loop form. As a result, in order to achieve a numerically stable and online effective procedure, a residual generator can be firstly designed using PS approach and then online implemented as a DO. This strategy is also known as *PS-based design, DO-based implementation* (Ding, 2013).

2.3 Data-driven residual generation techniques

Although model-based process monitoring techniques have been well studied in literature, their applications to industrial processes are often problematic due to the essential requirement on highly accurate process models, which leads to a sophisticated and time-consuming modelling procedure. In contrast, in modern industrial processes, many kinds of advanced sensors are used and a rich set of process data is often available or easy to be measured. Driven by this fact, many data-driven process monitoring techniques using process I/O data are studied in literature (Huang and Kadali, 2008; Ding, 2014).

2.3.1 SIM-aided process monitoring

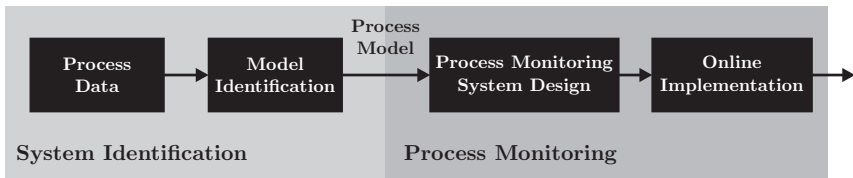


Figure 2.1: SIM-aided process monitoring

Among data-driven techniques, the SIM (Favoreel et al., 2000; Qin, 2006; Overschee and Moor, 1996) has gained great attention in the last two decades due to its numerical reliability and simplicity, by which, the system model and the associated system matrices can be identified using the process history I/O data. Based on it, the well-established model-based techniques for the purpose of process monitoring and control can be used to design an efficient process monitoring systems. A typical subspace identification algorithm includes two steps:

- identification of the extended observability matrix Γ_s and $\mathbf{H}_{u,s}$ as defined in Eq.(2.37), and
- calculation of system matrices $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$ and $\mathbf{D}$.

The design procedure of SIM-aided process monitoring is schematically sketched in Fig. 2.1.

2.3.2 Data-driven design of residual generator

Different from the conventional SIM-aided techniques, recently, a novel data-driven design procedure is proposed in Ding et al. (2009b) and later extended in Ding et al. (2014). In this work, instead of the identification of the system matrices, the SKR of the system is directly identified from process I/O data, which results in a direct link from process data to the design of process monitoring and FTC systems. The whole design procedure becomes much simpler, easier and efficient. A brief comparison between the conventional SIM-aided design and this novel data-driven method is sketched in Fig. 2.2. In order to understand this novel method, a brief introduction is given in this subsection.

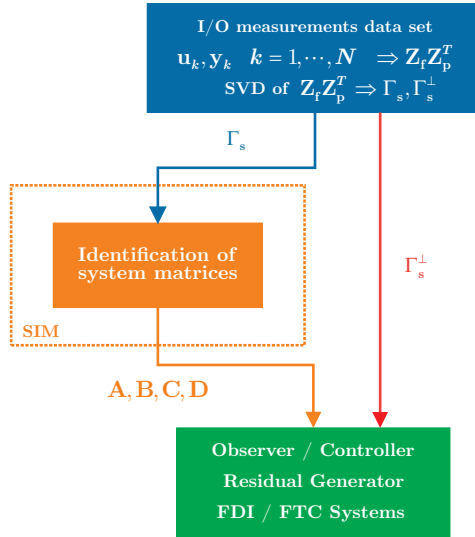


Figure 2.2: Conventional SIM-aided design and the novel data-driven approach.

Consider the I/O data model (2.35) in the fault- and disturbance-free case:

$$\mathbf{y}_{s,k} = \Gamma_s \mathbf{x}_k + \mathbf{H}_{u,s} \mathbf{u}_{s,k},$$

which can be rewritten into

$$\begin{bmatrix} \mathbf{u}_{s,k} \\ \mathbf{y}_{s,k} \end{bmatrix} = \Psi_s \begin{bmatrix} \mathbf{u}_{s,k} \\ \mathbf{x}_k \end{bmatrix}, \quad \Psi_s = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{H}_{u,s} & \Gamma_s \end{bmatrix} \in \mathcal{R}^{(s+1)(m+l) \times ((s+1)l+n)}.$$

For $s \geq n$:

$$\text{rank}(\Psi_s) = (s+1)l + n < (s+1)(m+l).$$

Therefore, the rows of Ψ_s are not independent and there exists at least a non-zero row vector $\alpha \neq \mathbf{0}$ such that $\alpha \Psi_s = \mathbf{0}$. Recall the notation (2.36) and introduce

$$\Lambda_k = [\lambda_k \quad \cdots \quad \lambda_{k+N-1}] \in \mathcal{R}^{k_\lambda \times N}, \quad (2.45)$$

$$\Lambda_{k,s} = [\lambda_{s,k} \quad \cdots \quad \lambda_{s,k+N-1}] = \begin{bmatrix} \Lambda_k \\ \vdots \\ \Lambda_{k+s} \end{bmatrix} \in \mathcal{R}^{(s+1)k_\lambda \times N}, \quad (2.46)$$

where N is sufficiently large. Considering the fault-free system (2.9)-(2.10) that is only affected by noise sequences ($\mathbf{d} = \mathbf{0}$), it follows that

$$\mathbf{Y}_{k,s} = \Gamma_s \mathbf{X}_k + \mathbf{H}_{u,s} \mathbf{U}_{k,s} + \mathbf{H}_{\xi,s} \Xi_{k,s} + \mathbf{V}_{k,s}, \quad (2.47)$$

which can be further written into

$$\begin{bmatrix} \mathbf{U}_{k,s} \\ \mathbf{Y}_{k,s} \end{bmatrix} = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{H}_{u,s} & \Gamma_s \end{bmatrix} \begin{bmatrix} \mathbf{U}_{k,s} \\ \mathbf{X}_k \end{bmatrix} + \begin{bmatrix} \mathbf{0} \\ \mathbf{H}_{\xi,s} \Xi_{k,s} + \mathbf{V}_{k,s} \end{bmatrix}.$$

Construct

$$\mathbf{Z}_f = \begin{bmatrix} \mathbf{U}_{k,s} \\ \mathbf{Y}_{k,s} \end{bmatrix}, \quad \mathbf{Z}_p = \begin{bmatrix} \mathbf{U}_{k-s_p-1,s_p} \\ \mathbf{Y}_{k-s_p-1,s_p} \end{bmatrix},$$

then the data-driven realization of the SKR can be summarized into the following algorithm (Ding et al., 2014):

Algorithm 2.1 Identification of SKR ($\mathcal{K}_s$)

Step 1 Collect process data and build $\mathbf{Z}_p$, $\mathbf{U}_{k,s}$, $\mathbf{Y}_{k,s}$.

Step 2 Do following QR-decomposition:

$$\begin{bmatrix} \mathbf{Z}_p \\ \mathbf{U}_{k,s} \\ \mathbf{Y}_{k,s} \end{bmatrix} = \begin{bmatrix} \mathbf{R}_{11} & \mathbf{0} & \mathbf{0} \\ \mathbf{R}_{21} & \mathbf{R}_{22} & \mathbf{0} \\ \mathbf{R}_{31} & \mathbf{R}_{32} & \mathbf{R}_{33} \end{bmatrix} \begin{bmatrix} \mathbf{Q}_1 \\ \mathbf{Q}_2 \\ \mathbf{Q}_3 \end{bmatrix}.$$

Step 3 Do following Singular Value Decomposition (SVD):

$$\begin{bmatrix} \mathbf{R}_{21} & \mathbf{R}_{22} \\ \mathbf{R}_{31} & \mathbf{R}_{32} \end{bmatrix} = [\mathbf{U}_1 \quad \mathbf{U}_2] \begin{bmatrix} \Sigma_1 & \mathbf{0} \\ \mathbf{0} & \Sigma_2 (\approx \mathbf{0}) \end{bmatrix} \begin{bmatrix} \mathbf{V}_1^T \\ \mathbf{V}_2^T \end{bmatrix}.$$

Step 4 $\mathcal{K}_s = \mathbf{U}_2^T \in \mathcal{R}^{((s+1)m-n) \times (s+1)(l+m)}$.

Based on the above algorithm, a state observer can thus be constructed by selecting rows from the identified SKR, which is summarized into the following algorithm (Ding et al., 2014):

Algorithm 2.2 *Data-driven design of observer-based residual generator*

Step 1 Run Algorithm 2.1 to obtain $\mathcal{K}_s$.

Step 2 Let $\psi_s^\perp$ be a row of $\mathcal{K}_s$ and of the form:

$$\psi_s^\perp = [\psi_{s,u}^\perp \quad \psi_{s,y}^\perp], \quad \psi_{s,u}^\perp \in \mathcal{R}^{(s+1)l}, \quad \psi_{s,y}^\perp \in \mathcal{R}^{(s+1)m}.$$

Step 3 Construct observer-based residual generator as follows:

$$\mathbf{x}_{o,k+1} = \mathbf{A}_o \mathbf{x}_{o,k} + \mathbf{B}_o \mathbf{u}_k + \mathbf{L}_o \mathbf{y}_k,$$

$$\mathbf{r}_k = \mathbf{g} \mathbf{y}_k - \mathbf{c}_o \mathbf{x}_{o,k} - \mathbf{d}_o \mathbf{u}_k,$$

where

$$\mathbf{A}_o = \begin{bmatrix} 0 & 0 & \cdots & 0 \\ 1 & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & 1 & 0 \end{bmatrix} \in \mathcal{R}^{s \times s}, \quad \mathbf{c}_o = [0 \quad \cdots \quad 0 \quad 1] \in \mathcal{R}^s,$$

$$\mathbf{L}_o = - \begin{bmatrix} \psi_{s,y}^\perp(1:m) \\ \vdots \\ \psi_{s,y}^\perp(((s-1)m+1):sm) \end{bmatrix}, \quad \mathbf{g} = \psi_{s,y}^\perp((sm+1):(s+1)m),$$

$$\mathbf{B}_o = - \begin{bmatrix} \psi_{s,u}^\perp(1:l) \\ \vdots \\ \psi_{s,u}^\perp(((s-1)l+1):sl) \end{bmatrix}, \quad \mathbf{d}_o = -\psi_{s,u}^\perp((sl+1):(s+1)l),$$

Since s is generally selected sufficiently large (much larger than system order n) during the identification of SKR, it is of practical advantage to reduce the order of the residual generator as much as possible. Therefore, after the identification, an order reduction of the parity vector $\psi_{s,y}^\perp$ is essential (Ding, 2014).

2.4 Residual evaluation and decision making

A process monitoring system mainly consists of two parts: residual generation and evaluation with decision making. The major objective of the residual evaluation is to determine a threshold based on the mathematical features of the evaluated residual signals in the fault-free case. Then, by comparing the online evaluated residual signal and the predetermined threshold, a decision is made whether a fault has occurred in the system or not.

In the fault- and disturbance-free case, the residual signal is zero. Any deviation of the residual signal from zero should indicate an occurrence of fault. However, in real industrial practice, unknown disturbance (*e.g.* measurement noise, process disturbance and model uncertainty) is inevitable, and in most cases a perfect decoupling of residual signal from unknown disturbance is infeasible. Therefore, in order to achieve a successful

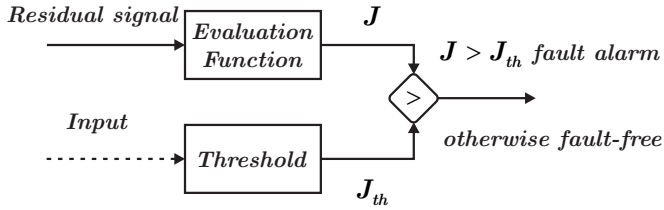


Figure 2.3: Schematic description of residual evaluation and decision making

fault detection based on the available residual signal, an evaluation process for the residual signal with threshold determination is essential. A schematic description of residual evaluation and decision making is shown in Fig. 2.3.

2.4.1 Residual evaluation strategies

In literature, depending on the type of the system under consideration, there exist two procedures for residual evaluation. One of them is statistic testing, which is mainly applied for stochastic systems (Basseville and Nikiforov, 1993; Lehmann and Romano, 2005; Ding, 2013). Another is norm-based evaluation, which is focused on the systems containing deterministic disturbance or system uncertainty (Ding, 2013). Due to its lower online computational effort and systematic threshold computation, the norm-based residual evaluation is widely applied. It is worth to notice that, an attractive combination of norm-based and statistical testing for residual signal evaluation is proposed in Ding et al. (2004) and Ding (2013).

In linear algebra, a norm is a function that assigns a strictly positive length or size to each vector in a vector space. In FDI, $\mathcal{L}_2$ and $\mathcal{L}_\infty$ are two standard norms used for residual evaluation and threshold determination.

$\mathcal{L}_2$ -norm: As one of the popular residual evaluation functions, $\mathcal{L}_2$ -norm measures the energy of a residual signal. For a given residual signal $\mathbf{r}_k \in \mathcal{R}^m$ provided by the aforementioned residual generators, its $\mathcal{L}_2$ -norm is defined by:

$$J_2 = \|\mathbf{r}_k\|_2^2 = \sum_{k=0}^{\infty} \mathbf{r}_k^T \mathbf{r}_k \quad (2.48)$$

Since the evaluation over the infinite time range is impractical, normally, an evaluation within a time window $[k_1, k_2]$ is applied. The corresponding $\mathcal{L}_{2,[k_1,k_2]}$ -norm evaluation function is defined by:

$$J_{2,[k_1,k_2]} = \|\mathbf{r}_k\|_{2,[k_1,k_2]}^2 = \sum_{k=k_1}^{k_2} \mathbf{r}_k^T \mathbf{r}_k \quad (2.49)$$

In practice, the Root Mean Square (RMS) value is often used instead of the $\mathcal{L}_2$ -norm, which measures the average energy of the residual signal over a time interval $[1, n]$. The

RMS value evaluation function is defined as follows:

$$J_{RMS,[k,n]} = \|\mathbf{r}_k\|_{RMS}^2 = \frac{1}{n} \sum_{i=1}^n \mathbf{r}_{k+i}^T \mathbf{r}_{k+i} \quad (2.50)$$

$\mathcal{L}_\infty$ -norm: The $\mathcal{L}_\infty$ -norm (also known as peak norm, max norm) is defined as the maximum of the absolute values of its components. For the purpose of FDI, the following peak value evaluation function is normally used:

$$J_{peak} = \|\mathbf{r}_k\|_{peak}^2 := \sup_{k \geq 0} \|\mathbf{r}_k\|_2^2 \quad (2.51)$$

2.4.2 Threshold setting and decision making

The selection of the threshold greatly influences the efficiency of the monitoring system. Generally, a threshold is a tolerant limit for disturbances and model uncertainties under fault-free operation conditions. A lower threshold setting usually leads the monitoring system to be subject to higher false alarms, and a higher threshold setting normally causes higher rate of missed detection. Therefore, based on the chosen evaluation function, the threshold can be generally defined by:

$$J_{th} = \sup_{\mathbf{f}=0, \mathbf{d}, \mathbf{\Delta}} J_e \quad (2.52)$$

where $\mathbf{d}$ and $\mathbf{\Delta}$ represent the disturbance and model uncertainties, respectively. J_e represents the feature of the evaluated residual signal, which could be J_2 , $J_{2,[k_1,k_2]}$, $J_{RMS,[k,n]}$ and J_{peak} .

After the residual evaluation and threshold setting, a decision logic has to be carried out. The simplest decision logic is to compare the feature of the evaluated residual J_e with the predetermined threshold J_{th} . In this way, decision is made as follows:

$$\begin{cases} J_e \leq J_{th} & \Rightarrow \text{fault-free;} \\ J_e > J_{th} & \Rightarrow \text{faulty and alarm.} \end{cases}$$

2.5 Multivariate statistical process monitoring techniques

In the data-driven design framework, the multivariate statistical process monitoring techniques, which utilize input and output information of the process, are widely used in the research and industrial applications in recent years, for instance, PCA (Wold et al., 1987; Jolliffe, 2002), PLS (MacGregor et al., 1994; Kruger and Dimitriadis, 2008; Li et al., 2010), FDA (Chiang et al., 2000; He et al., 2005), ICA (Lee, 1998; Kano et al., 2003; Lee et al., 2004; Zhang and Zhang, 2010), SVM (Chiang et al., 2004; Widodo and Yang, 2007) etc. Generally, the basic idea of multivariate statistical process monitoring techniques is to extract the statistical features to describe the desired process behavior from huge amount of process data. The extracted statistical features are used for later process monitoring purpose. The ability to tackle large number of highly correlated variables shows the multivariate statistical methods significant advantage. Due to their simple

forms and low design efforts compared with the model-based techniques, the multivariate statistical methods are widely applied to process monitoring in numerous large-scale industrial applications (Russell et al., 2000a; Chiang et al., 2001).

Within multivariate statistical framework, approaches like PCA, PLS and their variants are widely applied for process monitoring. To successfully apply the standard PCA and PLS schemes, it is a primary assumption that a linear process runs under stationary operating conditions. However, the process dynamics and nonlinearity are very important aspects in industrial processes. In order to deal with process dynamics and nonlinearity, many variates are intensively studied in literature, for instance, Lakshminarayanan et al. (1997), Russell et al. (2000b), Chen and Liu (2002), Choi and Lee (2004), and Peng et al. (2013).

2.6 Concluding remarks

This chapter provides a brief introduction to the modern process monitoring system design, which includes model-based and data-driven process monitoring techniques.

After the mathematical description of automation processes, the model-based residual generation techniques, FDF, DO and parity space approach, as well as their interconnections are briefly introduced. The third part of this chapter addresses the data-driven residual generation techniques, namely the SIM-aided process monitoring and data-driven design of residual generator. In contrast to model-based residual generation techniques, the data-driven approaches do not rely on process model and the residual generator can be directly constructed using process history I/O data. Furthermore, the residual evaluation, threshold setting and decision making are discussed.

In the data-driven design framework, the multivariate statistical methods serve as an alternative way to perform process monitoring. Due to the assumption on stationary operating conditions, the multivariate statistical methods are incapable to deal with process dynamics. Although their dynamic and recursive variates are proposed and studied in literature, the performance is quite limited. On the other hand, the SKR and the Stable Image Representation (SIR) are dual in control theory. Therefore, data-driven realization of SKR and SIR play important roles in data-driven integrated design of process monitoring and control system. Parallel to this chapter, Chapter 3 gives a brief introduction to the fundamentals of the FTC structure.

<http://www.springer.com/978-3-658-15927-6>

Plug-and-Play Monitoring and Performance Optimization
for Industrial Automation Processes

Luo, H.

2017, XVII, 149 p. 67 illus., 49 illus. in color., Softcover

ISBN: 978-3-658-15927-6