# Real-Time Implementation of Fault-Tolerant Control Systems With Performance Optimization

Shen Yin, *Member, IEEE*, Hao Luo, and Steven X. Ding

***Abstract*—In this paper, two online schemes for an integrated design of fault-tolerant control (FTC) systems with application to Tennessee Eastman (TE) benchmark are proposed. Based on the data-driven design of the proposed fault-tolerant architecture whose core is an observer/residual generator based realization of the Youla parameterization of all stabilization controllers, FTC is achieved by an adaptive residual generator for the online identification of the fault diagnosis relevant vectors, and an iterative optimization method for system performance enhancement. The performance and effectiveness of the proposed schemes are demonstrated through the TE benchmark model.**

***Index Terms*—Adaptive system, fault-tolerant control (FTC), optimization, signal processing.**

## I. Introduction

DUE to increasing demands on system performance, product quality as well as economic benefits, modern technical processes have become more complicated. As a result, the safety and reliability issues on the complicated process, which include either fault detection or both fault detection and appropriate fault-tolerant strategies, receive more attention and become the most critical factors in process monitoring nowadays [1]. As a possible solution to ensure safe and reliable operation of the system, the model-based fault detection and isolation (FDI) and fault tolerant control (FTC) techniques have received more and more attention [2]–[5]. Especially, a trend of integrating FDI into the feedback control scheme as a possible FTC solution with low needs for computation can be recently observed [6]–[9]. Following the industrial requirements, the FDI systems are usually designed to enhance the sensitivity for the faults [10]–[12], while the fault-tolerant controller is designed such that the resulting closed-loop system is asymptotically stable and satisfies a prescribed performance cost [13]–[18]. Differing from the above work, a so-called generalized internal model control (GIMC) structure is proposed in [19], and similarly in [20], [21], an observer based realization of the Youla parameterization controller [22] has been proposed as a fault-tolerant architecture, which could be constructed by only using process data. The core of the fault-tolerant architecture is a residual generator based realization of the Youla parameterization of all stabilizing controllers with an embedded residual generator for the purpose of fault detection. In this architecture, the setting of the state feedback and observer gains build the basic design tasks, while the determination of the parameter matrix will be done depending on the design demands for the system performance.

In contrast to model-based approach, which requires reliable *a priori* quantitative or qualitative knowledge about the process, the data-driven approach becomes an alternative way to obtain the process model from the process history data, which is simpler than to construct the process model based on first principles. 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, subspace identification techniques that directly identify the complete state space matrices have gained more attention in the last two decades and have been successfully implemented in many industrial applications [23]–[25], provided the identified process model, observer and parity space based FDI schemes can be designed [26], [27]. In the present work an alternative procedure is proposed [28]; the central idea behind the method is the design of the parameters of diagnostic observer (DO) based on the primary form of residual generators, whose parameters can be directly identified from the process data. In this way, the system identification becomes a part of the fault diagnosis system that leads to a shorter, easier and faster design procedure.

Strongly motivated by aforementioned studies, in this paper, we propose two online schemes for the fault-tolerant architecture proposed in [21] for the purpose of FTC. One is a gradient based iterative tuning scheme for the online optimization of the system performance, the other is an adaptive residual generator scheme for the online identification of the abnormal change of the system parameters. Due to the nature of the fault-tolerant architecture that, it can be equivalently reformed to any stabilizing controller and every part of the fault-tolerant architecture has its own purpose, the realization of advanced FDI/FTC schemes on this fault-tolerant architecture is the overall goal of this paper.

The paper is organized as follows. The needed preliminaries are reviewed in Section II. In Section III, for the purpose of FTC, two online algorithms are proposed for the fault-tolerant architecture, which consist of the online optimization of the system performance and the adaptive residual generator. In order to illustrate the application of the fault-tolerant architecture for the realization of FTC schemes, a benchmark study on TE process will be presented and discussed in Section IV. Finally, some concluding remarks are given in Section V.

Manuscript received December 9, 2012; revised March 26, 2013 and June 2, 2013; accepted June 14, 2013. Date of publication July 16, 2013; date of current version October 18, 2013. This work was supported by the National Natural Science Foundation of China (Grant 61304102) and by the Natural Science Foundation of Liaoning Province of China (Grant 2013020002).

S. Yin is with the Research Institute of Mechatronics and Automation, Bohai University, Jinzhou 121013, China, and also with the Research Center of Intelligent Control and Systems, Harbin Institute of Technology, Harbin 150001, China (e-mail: shen.yin2011@googlemail.com).

H. Luo and S. X. Ding are with the Institute for Automatic Control and Complex Systems (AKS), University of Duisburg-Essen, Duisburg 47057, Germany (e-mail: hao.luo@uni-due.de; steven.ding@uni-due.de).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIE.2013.2273477

*Notation:* The notation adopted throughout this paper is fairly standard. $\mathcal{R}^n$ denotes the $n$-dimensional Euclidean space and $\mathcal{R}^{n\times m}$ the set of all $n\times m$ real matrices. $\mathcal{RH}_\infty$ represents the set of all stable transfer function matrices. The superscript "$T$" stands for the transpose of a matrix. $col(P)$ denotes a column-wise re-ordering of a matrix $P$. "$T$" and "0" denote the identity and zero matrix with appropriate dimension, respectively.

## II. Preliminaries

### A. Plant Model and Left Coprime Factorization

In this paper, we consider the process modeled by

$$x(k+1) = Ax(k) + Bu(k) + \xi(k) \quad (1)$$
$$y(k) = Cx(k) + Du(k) + v(k) \quad (2)$$

where $u(k)\in\mathcal{R}^l$, $y(k)\in\mathcal{R}^m$ and $x(k)\in\mathcal{R}^n$ represent the process input, output and state variable vectors, respectively. $\xi(k)\in\mathcal{R}^n$ and $v(k)\in\mathcal{R}^m$ denote noise sequences that are normally distributed and statistically independent of $u(k)$ and $x(0)$. We assume that the above process model is controllable and observable. Let $G_u(z)$ denote the system transfer function matrix $G_u(z)=C(zI-A)^{-1}B+D$. All stabilizing controllers, in the context of a standard feedback control configuration, can be parameterized based on the so-called Youla parameterization [22], [29] as follows:

$$K(z)=\left(\hat{X}(z)-Q_c(z)\hat{N}(z)\right)^{-1}\left(\hat{Y}(z)-Q_c(z)\hat{M}(z)\right) \quad (3)$$

in which $Q_c(z)\in\mathcal{RH}_\infty$ is the parameter matrix. $\hat{M}(z)$, $\hat{N}(z)$, $\hat{X}(z)$ and $\hat{Y}(z)\in\mathcal{RH}_\infty$ and their definitions are referred to [20]. The pair $\hat{M}(z)$, $\hat{N}(z)$ builds a left coprime factorization (LCF) of $G_u(z)$, i.e., $G_u(z)=\hat{M}^{-1}(z)\hat{N}(z)$. A fundamental property of the LCF is that in the fault- and noise-free case

$$\forall\, u,\quad [-\hat{N}(z)\quad \hat{M}(z)]\begin{bmatrix}u(z)\\ y(z)\end{bmatrix}=0. \quad (4)$$

For this reason, the LCF, also known as kernel representation of system (1) and (2), is used for the parameterization of residual generators, which can be described by

$$r(z)=R(z)[-\hat{N}(z)\quad \hat{M}(z)]\begin{bmatrix}u(z)\\ y(z)\end{bmatrix} \quad (5)$$
$$=R(z)(y(z)-\hat{y}(z)) \quad (6)$$

where $R(z)(\neq 0)$ is called post-filter, $\hat{y}$ is delivered by a full order observer as an estimate of $y$ [30].

### B. Data-Driven Realization of LCF

In practice, the system matrices $A$, $B$, $C$ and $D$, as well as the order $n$ in the plant model (1) and (2) are often unknown *a priori*. Use the system input and output data to construct the Hankel matrices $U_f$, $U_p$, $Y_f$, $Y_p$. The state space model (1) and (2) can be rewritten into

$$Y_f=\Gamma_s X_k + H_{u,s}U_f + H_{\xi,s}W_f + V_f \quad (7)$$

where

$$\Gamma_s=\begin{bmatrix}C\\ CA\\ \vdots\\ CA^s\end{bmatrix},\quad H_{u,s}=\begin{bmatrix}D & 0 & \cdots & 0\\ CB & D & \cdots & 0\\ \vdots & \ddots & \ddots & \vdots\\ CA^{s-1}B & \cdots & CB & D\end{bmatrix}$$

and $H_{\xi,s}W_f+V_f$ represents the influence of the noise vectors on process output with $H_{\xi,s}$ having the same structure like $H_{u,s}$ and $W_f$, $V_f$ like $Y_f$. In the data-driven realization of LCF, it deals with the design of a dynamic system $H(z)$ satisfying

$$r(z)=H(z)\begin{bmatrix}u(z)\\ y(z)\end{bmatrix}=0 \quad (8)$$

if $\xi(z)=v(z)=0$, and $\lim_{k\to\infty}r(k)=0$ for $\xi(k)=v(k)=0$. To this end, rewrite (7) into

$$\begin{bmatrix}U_f\\ Y_f\end{bmatrix}=\Psi_s\begin{bmatrix}U_f\\ X_k\end{bmatrix}+\begin{bmatrix}0\\ H_{e,s}E_f\end{bmatrix}$$
$$\Psi_s=\begin{bmatrix}I & 0\\ H_{u,s} & \Gamma_s\end{bmatrix},\ E_f=\begin{bmatrix}W_f\\ V_f\end{bmatrix},\ H_{e,s}=[H_{\xi,s}\quad I]. \quad (9)$$

Note that for $\xi(k)=0$, $v(k)=0$ and any $y(k)$

$$\Psi_s^\perp\Psi_s=0\Rightarrow\Psi_s^\perp\begin{bmatrix}U_f\\ Y_f\end{bmatrix}=0. \quad (10)$$

Analogous to (8), $\Psi_s^\perp$ is called the data-driven realization of the LCF of (1) and (2). The design procedures are referred to [21].

### C. Data-Driven Design of Fault-Tolerant Architecture

References [20] and [22] proposed observer-based realization of the Youla parameterization, as sketched in Fig. 1. According to the Youla parameterization, all stabilizing controllers with plant model (1) and (2) could be reformed into the fault-tolerant architecture, whose core is a state observer

$$\hat{x}(k+1)=A\hat{x}(k)+Bu(k)+L(y(k)-\hat{y}(k)) \quad (11)$$
$$\hat{y}(k)=C\hat{x}(k)+Du(k). \quad (12)$$

The control law as well as residual generation are given by

$$u(k)=F\hat{x}(k)+r_c(k)+Vw(k),\ r_c(z)=R_c(z)r(z) \quad (13)$$
$$r_f(z)=R_f(z)r(z),\ r(k)=y(k)-\hat{y}(k) \quad (14)$$

where $w(k)$ denotes the reference signal, $V$ and $F$, respectively represent the pre-filter and the feedback gain. $R_c(z)=-Q_c(z)\in\mathcal{RH}_\infty$ is the parameter matrix. $f_A$, $f_P$ and $f_S$ represent actuator faults, process faults and sensor faults, respectively. In this control structure, residual vector $r_f(k)$ is available for the FDI purpose with a (stable) post-filter $R_f(z)$. The post-filter $R_f(z)$ is related to the FDI performance, more detailed design schemes are referred to [30]. In this paper, $R_f(z)$ is simply chosen as identity matrix.

As shown in Fig. 1, the realization of the proposed fault-tolerant architecture mainly contains following parts: 1) a state observer which delivers the state estimate and the residual signal; 2) the feedback gain $F$ which ensures the closed-loop stability; 3) a pre-filter $V$ which eliminates the static tracking

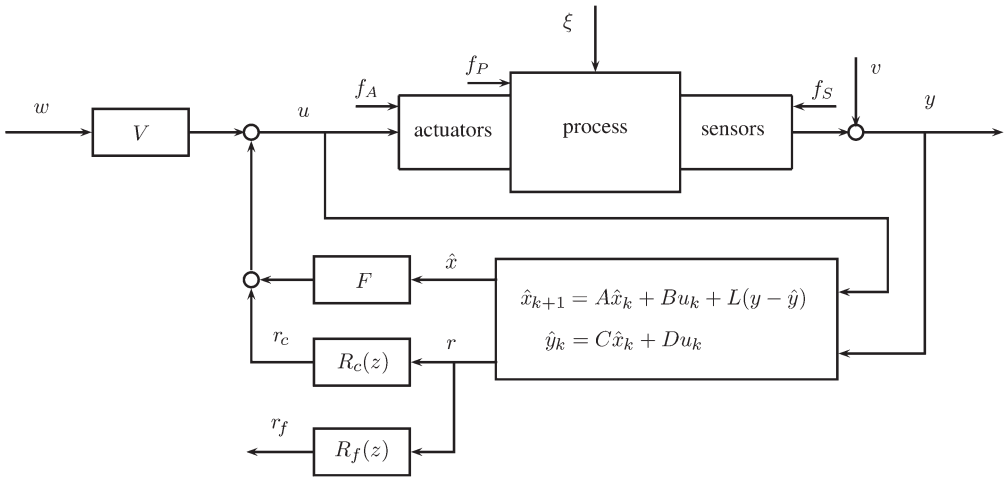


Fig. 1. Fault-tolerant architecture.

error; and 4) the parameter matrix $R_c(z)$ which concerns only system performance.

*1) Design of State Observer:* Consider the plant model (1) and (2) and assume $(A, B)$ is controllable and $(C, A)$ observable. Using the relation between a parity vector and the solution of the Luenberger equations [28], [30], the observer-based residual generator can be constructed as follows:

$$z(k+1) = A_z z(k) + B_z u(k) + L_z r(k) \in \mathcal{R}^s \quad (15)$$

$$r(k) = g_z y(k) - c_z z(k) - d_z u(k) \in \mathcal{R}. \quad (16)$$

The detailed design procedures are referred to in [21].

*2) Design of Feedback Gain and Pre-Filter:* The feedback gain $F$ should be designed in order to ensure the closed-loop stability. Since the state observer (15) and (16) delivers an estimate $z(k)$ for $Tx(k)$, namely $z(k) = T\hat{x}(k)$, the feedback gain $F$ can be designed by using conventional observer-based feedback control method. Assume the closed-loop is stabilized by $F$, recall the goal of the pre-filter is to ensure there is no static tracking error between reference signal $w$ and the system output $y$, therefore, we have following result by substituting the control input (13) into the system equations:

$$V = \left((c_z + d_z F_z)(I - A_z - B_z F_z)^{-1} B_z + d_z\right)^{\dagger} \quad (17)$$

where $(\bullet)^{\dagger}$ denotes the pseudo-inverse.

## III. Online Implementation of FTC Systems

Controllers are usually designed for the fault-free plant so that the overall system meets given performance requirements. FTC concerns the situation that the plant is subject to some faults which prevent the system to satisfy its goal. In this section, based on the aforementioned fault-tolerant architecture, two online algorithms are proposed for the purpose of FTC with low cost of online computation. One is an gradient-based optimization method which is given for the online configuration of the parameter matrix $R_c(z)$, whose aim is to optimize the system performance by means of disturbance rejection. The other is an adaptive residual generator scheme for online identification of the fault diagnosis relevant vectors. Based on the identified fault diagnosis relevant vectors, the fault-tolerant architecture could be reconstructed. The FTC could be achieved in following three steps: 1) apply adaptive scheme to identify the fault diagnosis relevant vectors online when a fault is successfully detected; 2) reconstruct the fault-tolerant architecture based on the identified fault diagnosis relevant vectors; 3) apply optimization scheme to reconfigure the parameter matrix in order to improve the system performance.

### A. Design of Parameter Matrix

Based on Youla parameterization, by choosing different parameter matrix $R_c(z)$, every stabilizing feedback controller for a given plant can be described as (3). Moreover, the nominal controller (with $R_c(z) = 0$) provides stability but may not necessarily attain satisfactory disturbance rejection. $R_c(z)$ should be introduced to enhance system performance by means of disturbance rejection. In order to obtain an implementable iteratively updated $R_c(z)$, we restrict the dynamic complexity of $R_c(z)$ in controllable canonical form, which gives us less design parameters. The state-space representation of $R_c(z) \in \mathcal{RH}_{\infty}^{l\times 1}$ is given as

$$x_r(k+1) = A_r x_r(k) + b_r r(k) \quad (18)$$

$$r_c(k) = C_r x_r(k) + d_r r(k) \quad (19)$$

where

$$A_r = \begin{bmatrix} 0 & 1 & \cdots & 0 \\ 0 & 0 & \ddots & \vdots \\ \vdots & \vdots & \ddots & 1 \\ -a_0 & -a_1 & \cdots & -a_{o_r-1} \end{bmatrix}, \; b_r = \begin{bmatrix} 0 \\ \vdots \\ 1 \end{bmatrix},$$

$$C_r = \begin{bmatrix} c_{11} & \cdots & c_{1o_r} \\ \vdots & \ddots & \vdots \\ c_{l1} & \cdots & c_{lo_r} \end{bmatrix}, \; d_r = \begin{bmatrix} d_1 \\ \vdots \\ d_l \end{bmatrix}.$$

$x_r(k) \in \mathcal{R}^{o_r}$ is the state vector and $A_r \in R^{o_r \times o_r}$ is a stable matrix chosen by the designer. Typically, one would select the eigenvalues of the matrix $A_r$ to be comparable in magnitude with the eigenvalues of the matrix $A_z - L_z c_z$, which

determines the observer dynamics. $C_r \in \mathcal{R}^{l\times o_r}$, $d_r \in \mathcal{R}^l$ are the design parameters of $R_c(z)$, thus we note

$$\theta_c = col(C_r) \in \mathcal{R}^{l\cdot o_r}, \quad \theta_d = col(d_r) \in \mathcal{R}^l \tag{20}$$

$$\theta = [\,\theta_c^T \quad \theta_d^T\,]^T. \tag{21}$$

Based on the fault-tolerant architecture, as shown in Fig. 1, the closed-loop dynamics could be obtained as follows:

$$\begin{bmatrix} x(k+1) \\ x_r(k+1) \\ e(k+1) \end{bmatrix} = \begin{bmatrix} A+BF & BC_r & Bd_rc_z - BF_z \\ 0 & A_r & b_rc_z \\ 0 & 0 & A_z - L_zc_z \end{bmatrix} \begin{bmatrix} x(k) \\ x_r(k) \\ e(k) \end{bmatrix} + \begin{bmatrix} BV & 1 & Bd_r \\ 0 & 0 & b_r \\ 0 & T & -L_z \end{bmatrix} \begin{bmatrix} w(k) \\ \xi(k) \\ v(k) \end{bmatrix} \tag{22}$$

$$\begin{bmatrix} y(k) \\ u(k) \end{bmatrix} = \begin{bmatrix} c+dF & dC_r & d(d_rc_z - F_z) \\ F & C_r & d_rc_z - F_z \end{bmatrix} \begin{bmatrix} x(k) \\ x_r(k) \\ e(k) \end{bmatrix} + \begin{bmatrix} dV & 0 & 1+dd_r \\ V & 0 & d_r \end{bmatrix} \begin{bmatrix} w(k) \\ \xi(k) \\ v(k) \end{bmatrix} \tag{23}$$

where $e(k) = Tx(k) - z(k) \in \mathcal{R}^n$ denotes the state estimation error for the system. The control objective is chosen as

$$J(\theta(k), k) = \sum_{k=k_0}^{N+k_0-1} e_y^T(\theta(k), k)\, W e_y(\theta(k), k) \tag{24}$$

where $k_0$ and $N$ are integers represent the starting point and the time window, respectively, $W = W^T > 0$ is a weighting factor and

$$e_y(\theta(k), k) = \begin{bmatrix} y(k) \\ u(k) \end{bmatrix}.$$

The control objective (24) is to achieve fast regulation with disturbance rejection on system output while containing the control effort. The design parameter $\theta$ can be updated by using following gradient-based iterative tuning method:

$$\theta(k+1) = \theta(k) - \gamma(k)\frac{\partial J(\theta(k), k)}{\partial \theta} \tag{25}$$

where $\gamma(k)$ is a small positive value denotes the step size at time point $k$. The step size $\gamma(k)$ scales the convergence speed, therefore it could be determined by searching for the minimum of the cost function along the search direction. In this paper, the step size is only chosen as a small positive constant, more details on the step size determination are referred to in [32]. In order to obtain the gradients, the following theorems play a key role in the iterative configuration of the $R_c(z)$.

*Theorem 1:* Given the fault-tolerant architecture shown in Fig. 1 with plant model (1) and (2) and state observer (15) and (16), then the following gradient estimator delivers the gradients of $y(k)$, $u(k)$, w.r.t. the design parameters of $R_c(z)$:

$$\frac{\partial z(k+1)}{\partial \theta_{c_i}} = (A_z + B_zF_z)\frac{\partial z(k)}{\partial \theta_{c_i}} + B_zI_{c_i}x_r(k) \tag{26}$$

$$\begin{bmatrix} \frac{\partial y(k)}{\partial \theta_{c_i}} \\ \frac{\partial u(k)}{\partial \theta_{c_i}} \end{bmatrix} = \begin{bmatrix} c_z + d_zF_z \\ F_z \end{bmatrix} \frac{\partial z(k)}{\partial \theta_{c_i}} + \begin{bmatrix} d_zI_{c_i} \\ I_{c_i} \end{bmatrix} x_r(k) \tag{27}$$

where $\theta_c$ is $i$th element of $\theta_c$ and $I_{c_i} = (\partial C_r/\partial \theta_{c_i})$

$$\frac{\partial z(k+1)}{\partial \theta_{d_i}} = (A_z + B_zF_z)\frac{\partial z(k)}{\partial \theta_{d_i}} + B_zI_{d_i}r(k) \tag{28}$$

$$\begin{bmatrix} \frac{\partial y(k)}{\partial \theta_{d_i}} \\ \frac{\partial u(k)}{\partial \theta_{d_i}} \end{bmatrix} = \begin{bmatrix} c_z + d_zF_z \\ F_z \end{bmatrix} \frac{\partial z(k)}{\partial \theta_{d_i}} + \begin{bmatrix} d_zI_{d_i} \\ I_{d_i} \end{bmatrix} r(k) \tag{29}$$

where $\theta_d$ is $i$th element of $\theta_d$ and $I_{d_i} = (\partial d_r/\partial \theta_{d_i})$.

*Proof:* The proof is straightforward and omitted here.

*Remark:* Theorem 1 indicates that the gradient estimator (26)–(29) could deliver the gradients w.r.t. $\theta$ directly online, and more important is the gradient estimator can be built by using the system matrices of the pre-designed observer $A_z$, $B_z$, $c_z$, $d_z$ and the predesigned feedback control gain $F_z$.

*Theorem 2:* Given the fault-tolerant architecture shown in Fig. 1 with plant model (1) and (2), state observer (15) and (16), the gradient estimator (26)–(29) and the gradient-based iterative tuning method (25). Then there exists a positive $\gamma^*$ such that for all $\gamma \in (0, \gamma^*)$ the following results are hold.

1) Convergence is exponential:

$$\|\theta(k) - \theta^*\| \le C_1(1 - L_g\gamma)^{k-k_0}\|\theta(0) - \theta^*\| \quad \text{for all } k = 0, 1, \ldots \tag{30}$$

where $C_1$, $L_g$ are some constants that $C_1 > 0$, $L_g > 0$ which independent of $\gamma$.

2) Convergence accuracy:

$$\|\theta(k) - \theta^*\| \le C_2\gamma \tag{31}$$

in which $\theta^*$ is the global minimizer of the averaged equation of criterion (24). The constant $C_2 > 0$ is independent of $\gamma$.

*Proof:* To illustrate the proof of Theorem 2, we consider the following adaptive systems of the general form:

$$x(k+1) = A(\theta(k))\,x(k) + B(\theta(k), k), \quad x_0 \tag{32}$$

$$\theta(k+1) = \theta(k) + \gamma g(\theta(k), x(k), k), \quad \theta_0 \tag{33}$$

under following assumptions:

1) $A$ is stable.
2) $A$ and $B$ are continuously differentiable w.r.t. $\theta$.
3) $B$ is bounded in $k$.
4) $D_\theta B(\theta, k)$ (the Jacobian of B w.r.t. $\theta$) is bounded in $k$.
5) $g$ is locally bounded and locally Lipschitz continuous in $(\theta, x)$ uniformly in $k$ and processes an average $g^a(\theta^a(k))$:

$$g^a(\theta^a(k)) = \lim_{N\to\infty}\frac{1}{N}\sum_{k=k_0}^{N+k_0-1} g(\theta(k), x(k), k). \tag{34}$$

Consider the averaged equation

$$\theta^a(k+1) = \theta^a(k) + \gamma g^a(\theta^a(k), x^a(k), k). \tag{35}$$

There exists a positive constant $\gamma^*$ such that for all $\gamma \in (0, \gamma^*)$ the solutions $\theta(k, \theta_0, k_0, \gamma)$ of the difference equation (33) for

any $k_0 \geq 0$ and for any $\theta_0$ can be uniformly approximated by $\theta^a(k, \theta_0)$ solution of the averaged difference equation (35) on $k \geq k_0$

$$\lim_{\gamma \to 0} \|\theta(k, \theta_0, k_0, \gamma) - \theta^a(k, \theta_0, \gamma)\| = 0 \qquad \forall\, k \geq k_0. \quad (36)$$

The proof of (36) is referred to in [33] and omitted here.

Consider the averaged equation of the control objective

$$J^a\left(\theta^a(k)\right) = \lim_{N \to \infty} \frac{1}{N} \sum_{k=k_0}^{N+k_0-1} e_y^T\left(\theta^a(k), k\right) W e_y\left(\theta^a(k), k\right) \quad (37)$$

and denote the global minimizer of (37) as $\theta^*$

$$\theta^* = \arg\min_{\theta^a} J^a\left(\theta^a(k)\right). \quad (38)$$

Recall the gradient estimator (26)–(29), the gradient estimator is stable due to $A_z + B_z F_z$ is stable, and the gradient vector is indeed independent of $\theta$. It can be easily shown that the Hessian matrix of the optimization criterion (37) w.r.t. $\theta$ is positive semi-definite. Therefore, the optimization problem (37) w.r.t. $\theta$ is convex and its global optimum is guaranteed [32], which implies that provided $\gamma$ is sufficiently small then $\theta^a(k)$ is bounded and converges to $\theta^*$. Recall the adaptive system (37), we have then

$$g^a\left(\theta^*, x^a(k), k\right) = 0. \quad (39)$$

Furthermore, based on the averaging analysis [33], $\|\theta(k+1) - \theta(k)\|$ is of order $\gamma$ and thus we have

$$\begin{aligned}\theta(k+1) - \theta^* &= \theta(k) - \theta^* + \gamma g^a\left(\theta^*, x^a(k), k\right) \\ &\quad + \gamma\left[g\left(\theta(k), x(k), k\right) - g^a\left(\theta^*, x^a(k), k\right)\right] \\ &= \theta(k) - \theta^* + O(\gamma^2).\end{aligned}$$

Exploiting the continuity of $g$ and $x^a$, we have the following estimate:

$$\|\theta(k+1) - \theta^*\| \leq (1 + \gamma L_g)\|\theta(k) - \theta^*\| + O(\gamma) \quad (40)$$

for some $L_g > 0$. From the comparison principle we get finally

$$\|\theta(k) - \theta^*\| \leq C_1(1 + \gamma L_g)^{k-k_0}\|\theta(0) - \theta^*\| \quad (41)$$

where the constant $C_1 > 0$ is independent of $\gamma$. For $k \leq L/\gamma$ the function $(1 + \gamma L_g)^k \leq (1 + \gamma L_g)^{L/\gamma} \leq e^{LL_g}$ is of order of 1, which establishes

$$\|\theta(k) - \theta^*\| \leq C_2\gamma \quad (42)$$

in which the constant $C_2 > 0$ is independent of $\gamma$.

*Remark:* Theorem 2 indicates that, generally, the adaptive algorithm achieves near optimal performance in an exponential manner. The convergence is according to a large time constant, as $\gamma$ is sufficiently small.

In summary, the following algorithm is given for the online configuration of the parameter matrix $R_c(z)$.

Algorithm: Parameter matrix optimization

*Step 1:* Choose $A_r$ to be stable and choose initial values for $c_r$, $d_r$.

*Step 2:* Construct the gradient estimators according to (26)–(29) and set initial state equal to zero.

*Step 3:* Choose step size $\gamma$ for each time step $k$, and update $\theta$ iteratively according to (25).

*Step 4:* End the iterative calculation when stop conditions are fulfilled.

Next, an adaptive scheme is briefly introduced based on the above fault-tolerant architecture for the purpose of FTC.

### B. Adaptive Scheme

Since the faults could happen in every component of the system and could be unknown, it is very difficult or even impossible to design a controller to deliver a good performance in fault-free case to suit every possible fault. Therefore, it is necessary to redesign the controller in order to guarantee the closed-loop stability on the one hand, and on the other hand to achieve a good system performance. For this reason, in this section, an adaptive residual generator is briefly introduced, based on which the reconfiguration of the controller could be accomplished. It should be noticed that differing from the contribution in [31], whose aim is to identify the fault diagnosis relevant vectors online in order to reconstruct the residual generator to suit a new operating point of a nonlinear system, the adaptive residual generator is proposed here for the FTC purpose. In the sequel, we assume that there exists no structural change, i.e., change e.g., in the observability, and the changes in parameters are slow and can be considered nearly constant in a (large) time interval. For our purpose, the residual generator (15) and (16) is extended to

$$z(k+1) = A_z z(k) + B_z u(k) + L_z r(k) + L_0 r(k) \quad (43)$$
$$r(k) = g_z y(k) - c_z z(k) - d_z u(k) \quad (44)$$

where $L_0$ provides additional degree of design freedom and should ensure the stability of $\bar{A}_z = A_z - L_z c_z - L_0 c_z$. Let

$$\begin{aligned}\delta &= \begin{bmatrix}\delta_u \\ \delta_y\end{bmatrix} \in \mathcal{R}^{(s+1)(m+l)}, \\ \delta_u &= col\begin{bmatrix}B_z \\ d_z\end{bmatrix},\ \delta_y = \begin{bmatrix}L_z \\ g_z\end{bmatrix} \\ Q\left(u(k), y(k)\right) &= \begin{bmatrix}\mathcal{U}(k) & -L_0 u^T(k) & \mathcal{Y}(k) & L_0 y^T(k)\end{bmatrix} \\ \mathcal{U}(k) &= \begin{bmatrix}u_1(k) \times I_{s\times s} & \cdots & u_l(k) \times I_{s\times s}\end{bmatrix} \\ \mathcal{Y}(k) &= \begin{bmatrix}y_1(k) \times I_{s\times s} & \cdots & y_m(k) \times I_{s\times s}\end{bmatrix}.\end{aligned}$$

Equations (43) and (44) can be rewritten as

$$z(k+1) = \bar{A}_z z(k) + Q\left(u(k), y(k)\right)\delta \quad (45)$$
$$r(k) = \begin{bmatrix}-u^T(k) & y^T(k)\end{bmatrix}\begin{bmatrix}d_z^T \\ g_z^T\end{bmatrix} - c_z z(k). \quad (46)$$

The task consists of designing a residual generator which is adaptive to $\delta$ and delivers a residual signal $r(k)$ satisfying $\lim_{k\to\infty} r(k) = 0$ and, if possible, with an exponential converging

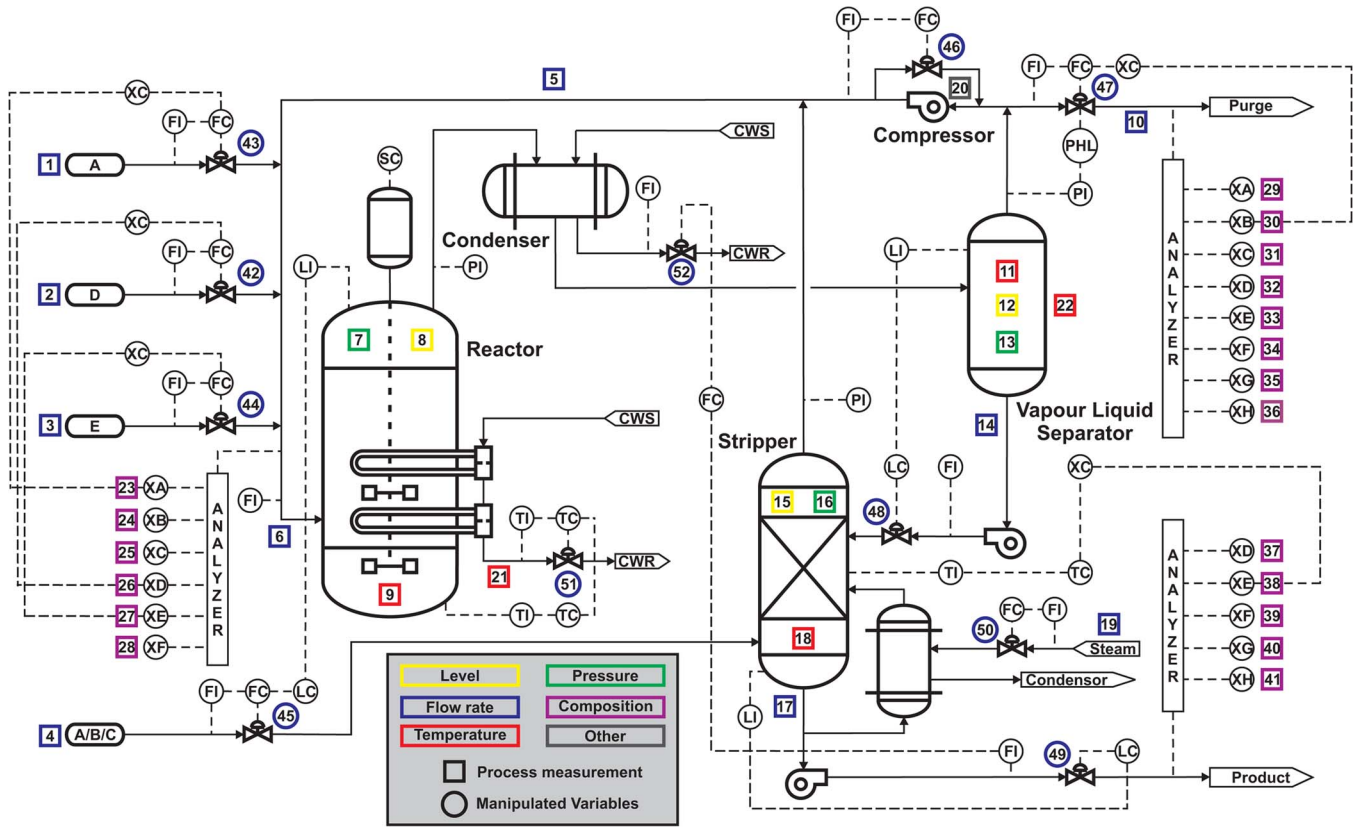


Fig. 2. Process flow diagram of Tennessee Eastman process.

speed independent of a constant change in $\delta$. The adaptive residual generator scheme consists of three subsystems:

1) *Residual generator:*

$$\hat{z}(k+1) = \bar{A}_z \hat{z}(k) + Q\left(u(k), y(k)\right)\hat{\delta}(k) + V(k+1)\left(\hat{\delta}(k+1) - \hat{\delta}(k)\right) \tag{47}$$

$$r(k) = \left[-u^T(k) \quad y^T(k)\right]\begin{bmatrix} \hat{d}_z^T(k) \\ \hat{g}_z^T(k) \end{bmatrix} - c_z \hat{z}(k). \tag{48}$$

2) *Auxiliary filter:*

$$V(k+1) = \bar{A}_z V(k) + Q\left(u(k), y(k)\right) \tag{49}$$

$$\begin{aligned} \varphi(k) &= c_z V(k) - \left[0 \quad \cdots \quad -u^T(k) \quad 0 \quad \cdots \quad y^T(k)\right] \\ &\left[0 \quad \cdots \quad -u^T(k) \quad 0 \quad \cdots \quad y^T(k)\right]\delta \\ &= \left[-u^T(k) \quad y^T(k)\right]\begin{bmatrix} d_z^T(k) \\ g_z^T(k) \end{bmatrix}. \end{aligned} \tag{50}$$

3) *Parameter estimator:*

$$\hat{\delta}(k+1) = \gamma(k)\varphi^T(k) r(k) + \hat{\delta}(k) \tag{51}$$

$$\gamma(k) = \frac{\mu}{\sigma + \varphi(k)\varphi^T(k)}, \quad \sigma \geq 0,\ 0 < \mu < 2. \tag{52}$$

The detailed proof of stability and the convergence property can be found in [31]. To this end, the design of the adaptive observer is summarized in following algorithm.

Algorithm: Adaptive residual generator

*Step 1:* Set the initial values $k = 0$, $\hat{z}(0)$, $\hat{\delta}(0)$, $V(0) = 0$, $\varphi(0) = 0$, $r(0) = \hat{g}_z(0)y(0) - c_z\hat{z}(0) - \hat{d}_z(0)u(0)$.

*Step 2:* Compute $V(k+1)$, $\hat{\delta}(k+1)$ and $\hat{z}(k+1)$ according to (49), (51) and (47).

*Step 3:* Increase $k$ by one, receive $y(k)$, $u(k)$.

*Step 4:* Compute $r(k)$, $\varphi(k)$ according to (48) and (50).

Since the changes in the original system could be identified using above adaptive algorithm and the residual generator (43) and (44) delivers the state estimate of the faulty plant, the feedback gain $F$ which delivers the closed-loop stability, the pre-filter $V$ which eliminates the static tracking error and as well as the gradient estimator (26)–(29) for parameter matrix $R_c(z)$ could be automatically updated based on the given adaptive observer scheme.

## IV. Benchmark Example

In this section, the performance and effectiveness of the proposed FTC schemes are illustrated through the TE (Tennessee Eastman) benchmark process. The simulation results are demonstrated and discussed in detail.

### A. Benchmark Description

TE process model is a realistic simulation program of a chemical plant which is widely accepted as a benchmark for control and monitoring studies. The process is described in [34] and the FORTRAN code of the process is available over Internet. Fig. 2 shows the flow diagram of the process with five major units, i.e., reactor, condenser, compressor, separator and stripper. The process allows total 52 measurements out of which 41 are process variables and 11 are manipulated variables listed in Tables I and II.

TABLE I
PROCESS VARIABLES

| Block name | Variable name | Number | Marker |
|---|---|---|---|
| Input feed | A feed (stream 1) | XMEAS(1) | 1 |
| | D feed (stream 2) | XMEAS(2) | 2 |
| | E feed (stream 3) | XMEAS(3) | 3 |
| | A and C feed | XMEAS(4) | 4 |
| Reactor | Reactor feed rate | XMEAS(6) | 6 |
| | Reactor pressure | XMEAS(7) | 7 |
| | Reactor level | XMEAS(8) | 8 |
| | Reactor temperature | XMEAS(9) | 9 |
| Separator | Separator temperature | XMEAS(11) | 11 |
| | Separator level | XMEAS(12) | 12 |
| | Separator pressure | XMEAS(13) | 13 |
| | Separator underflow | XMEAS(14) | 14 |
| Stripper | Stripper level | XMEAS(15) | 15 |
| | Stripper pressure | XMEAS(16) | 16 |
| | Stripper underflow | XMEAS(17) | 17 |
| | Stripper temperature | XMEAS(18) | 18 |
| | Stripper steam flow | XMEAS(19) | 19 |
| Miscellaneous | Recycle flow | XMEAS(5) | 5 |
| | Purge rate | XMEAS(10) | 10 |
| | Compressor work | XMEAS(20) | 20 |
| | Reactor water temperature | XMEAS(21) | 21 |
| | Separator water temperature | XMEAS(22) | 22 |
| Reactor feed analysis | Component A | XMEAS(23) | 23 |
| | Component B | XMEAS(24) | 24 |
| | Component C | XMEAS(25) | 25 |
| | Component D | XMEAS(26) | 26 |
| | Component E | XMEAS(27) | 27 |
| | Component F | XMEAS(28) | 28 |
| Purge gas analysis | Component A | XMEAS(29) | 29 |
| | Component B | XMEAS(30) | 30 |
| | Component C | XMEAS(31) | 31 |
| | Component D | XMEAS(32) | 32 |
| | Component E | XMEAS(33) | 33 |
| | Component F | XMEAS(34) | 34 |
| | Component G | XMEAS(35) | 35 |
| | Component H | XMEAS(36) | 36 |
| Product analysis | Component D | XMEAS(37) | 37 |
| | Component E | XMEAS(38) | 38 |
| | Component F | XMEAS(39) | 39 |
| | Component G | XMEAS(40) | 40 |
| | Component H | XMEAS(41) | 41 |

TABLE II
PROCESS MANIPULATED VARIABLES

| Variable name | Base value(%) | Units | Number | Marker |
|---|---|---|---|---|
| D feed flow | 63.053 | $\mathrm{kg}h^{-1}$ | XMV(1) | ⑫ |
| A feed flow | 24.644 | kscmh | XMV(3) | ⑬ |
| E feed flow | 53.980 | $\mathrm{kg}h^{-1}$ | XMV(2) | ⑭ |
| A and C feed flow | 61.302 | kscmh | XMV(4) | ⑮ |
| Compressor recycle valve | 22.210 | % | XMV(5) | ⑯ |
| Purge valve | 40.064 | % | XMV(6) | ⑰ |
| Separator pot liquid flow | 38.100 | $\mathrm{m}^3\mathrm{h}^{-1}$ | XMV(7) | ⑱ |
| Stripper liquid product flow | 46.534 | $\mathrm{m}^3\mathrm{h}^{-1}$ | XMV(8) | ⑲ |
| Stripper steam valve | 47.446 | % | XMV(9) | ⑳ |
| Reactor cooling water flow | 41.106 | $\mathrm{m}^3\mathrm{h}^{-1}$ | XMV(10) | ㉑ |
| Condenser cooling water flow | 18.114 | $\mathrm{m}^3\mathrm{h}^{-1}$ | XMV(11) | ㉒ |

In the TE process model, 20 process faults were initially defined in [34] and an additional valve fault was further introduced in [35], which can be seen in Table III. Since no prior knowledge about the mathematical model of TE process is available, the process monitoring fault diagnosis (PM-FD) system shall be designed only based on the process data. The data sets given in [35] are widely accepted for PM-FD study, in which 22 training sets (including normal operating condition) were collected to record the process measurements for 24 operation hours. Correspondingly, 22 (on-line) test data sets including 48 hours plant operation time were generated. These data sets can be downloaded from http://brahms.scs.uiuc.edu. According to the original TE code, a Simulink code provided by [36] is available to simulate the plant's closed-loop behavior. Based on the simulator, the operation modes, measurement noise, sampling time and magnitudes of the faults can be easily modified and thus its generated data sets are more helpful for PM-FD comparison study. Note that the control structure utilized in [35] is different from the one in [36], which may lead to some differences in later simulation study. In the analysis, the base operating mode of TE process is considered to be identical with the case in [35] to simulate the process behavior under stationary operating conditions. The simulator can be downloaded from http://depts.washington.edu/control/LARRY/TE/download.html.

TABLE III
DESCRIPTIONS OF PROCESS FAULTS IN TE PROCESS

| Fault number | Process variable | Type |
|---|---|---|
| IDV(1) | A/C feed ratio, B composition constant | Step |
| IDV(2) | B composition, A/C ration constant | Step |
| IDV(3) | D feed temperature | Step |
| IDV(4) | Reactor cooling water inlet temperature | Step |
| IDV(5) | Condenser cooling water inlet temperature | Step |
| IDV(6) | A feed loss | Step |
| IDV(7) | C header pressure loss-reduced availability | Step |
| IDV(8) | A,B,C feed composition | Random variation |
| IDV(9) | D feed temperature | Random variation |
| IDV(10) | C feed temperature | Random variation |
| IDV(11) | Reactor cooling water inlet temperature | Random variation |
| IDV(12) | Condenser cooling water inlet temperature | Random variation |
| IDV(13) | Reaction kinetics | Slow Drift |
| IDV(14) | Reactor cooling water valve | Sticking |
| IDV(15) | Condenser cooling water valve | Sticking |
| IDV(16) | Unknown | Unknown |
| IDV(17) | Unknown | Unknown |
| IDV(18) | Unknown | Unknown |
| IDV(19) | Unknown | Unknown |
| IDV(20) | Unknown | Unknown |
| IDV(21) | The valve fixed at steady state position | Constant position |

### B. Simulation Results and Discussions

Recall our FTC objective in the TE benchmark is to detect faults as early as possible and maintain the process performance in the admissible range in both fault-free and faulty cases. Since among the faults listed in Table III, the fault IDV(1) leads a product degradation of the TE process, it is taken in this section as an example for dedicated study. With respect to its physical meaning, it can be thought of as an abnormal change of the process parameters that affects the A/C feed ratio and B composition. The fault will show itself as a change in the stripper underflow with a process delay of about 1–2 hours, which is marked in Fig. 2 with marker 17 (XMEAS(17)). We first implement performance prediction and diagnosis scheme (15) and (16) based on the original PID-type controllers. The upper plot of Fig. 3 shows the measurement of the process product, the estimation is plotted in the middle plot and the lower plot of Fig. 3 shows the difference between the measurement and its estimation.

Assume the fault IDV(1) starts at 10 simulation hour, define the testing statistic $J = (1/2\sigma_r^2 M)(\sum_{i=1}^{M} r(i))^2$ and set

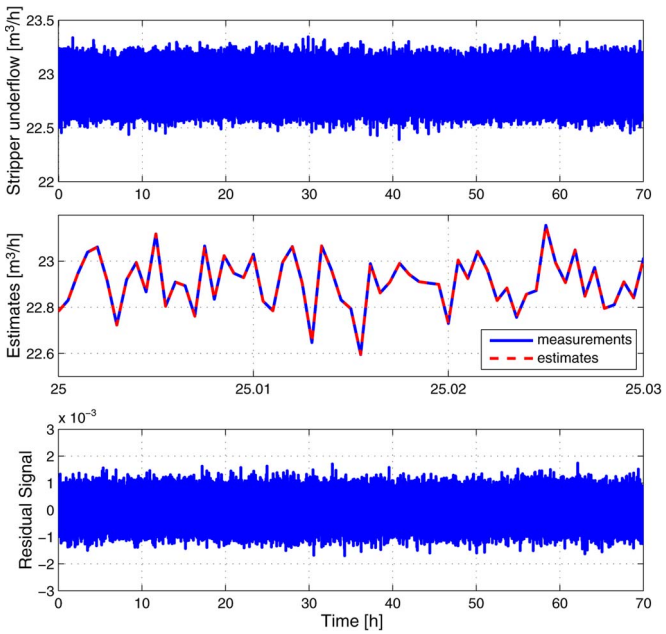


Fig. 3. Process production and its estimates (fault-free case).

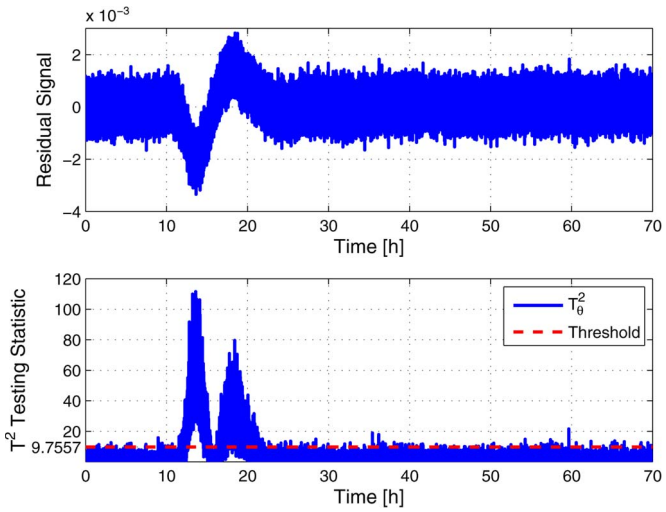


Fig. 4. Process monitoring results (faulty case).

threshold $J_{th} = \chi_\alpha/2$ [28]. The upper plot of Fig. 4 shows the residual signal $r$, and the evaluated residual signal $(M = 5)$ and the corresponding threshold are shown in the lower plot. It can be easily seen from the lower plot of Fig. 4 that, the fault IDV(1) is successfully detected. A more comprehensive comparative study on different data-driven FDI methods is given in [37] where it is shown that the approach described above is one of the most effective. In what follows, it is assumed that the fault detection is carried out by an appropriate method.

In the simulation studies, the FTC architecture shown in Fig. 1 is used, the TE process is controlled by the PID-type controllers. The order of the parameter matrix $R_c(z)$ is set as $o_r = 5$, the time window for performance optimization is set as $N = 1000$ samples and the step size $\gamma$ is chosen as a small positive constant, i.e., $\gamma = 8 \times 10^{-4}$. A comparison of the strip underflow without and with the application of the

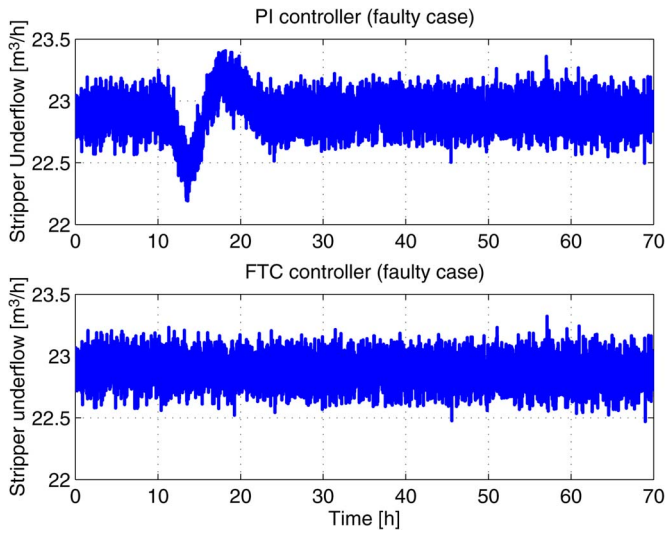


Fig. 5. Process production (faulty case using different controllers).

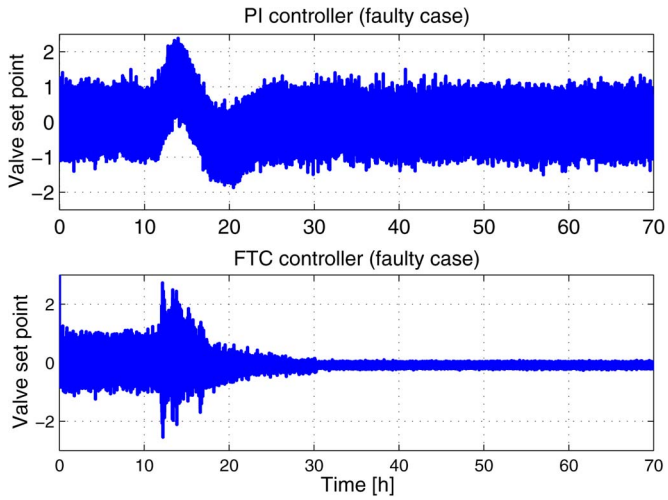


Fig. 6. Control inputs (faulty case using different controllers).

proposed FTC scheme are demonstrated in the upper part and the lower part of Fig. 5, respectively. It is obvious that, after the application of the proposed FTC scheme, the performance of the process production in the faulty case is significantly improved and is very close to the faulty-free case.

To demonstrate the effectiveness of the optimization scheme, the control inputs of the plant without and with the proposed FTC scheme are shown in Fig. 6, respectively. Recall our optimization criterion (24), which aims to achieve fast regulation with disturbance rejection while constraining the control effort. It can be clearly seen from Fig. 6 that the variance of the control input is significantly decreased by the use of the proposed scheme: the variance of the control input with the FTC scheme for the time window of 30–70 simulation hours is calculated as 0.0020 while the variance of the control input delivered without it for the same time window is calculated as 0.1387. It should also be noted that the control input in the studied case is the control signal of the valves. Technically speaking, decreasing the frequency in the change of the position of a valve (decreasing the chattering) would increase its life-time.

## V. Conclusion

In this paper, a novel fault tolerant control architecture is proposed suitable for online implementation. It consists of two parts, the adaptive residual generator enables the online identification of the fault diagnosis relevant vectors, and a gradient-based optimization approach for the online configuration of the parameter matrix. In this way, the whole fault-tolerant architecture can be reconfigured online and the online calculation cost is reduced to minimal. The performance and the effectiveness of the proposed scheme are demonstrated through the TE benchmark model. Through the case study, it is evident the controller is adapted to the faulty situation so that the overall system continues to satisfy its goal, and the process performances are automatically optimized.

## References


[1] S.-L. Jaemsae-Jounela, "Future trends in process automation," *Annu. Rev. Control*, vol. 31, no. 2, pp. 211–220, 2007.

[2] J. J. Gertler, *Fault Detection and Diagnosis in Engineering Systems*. New York, NY, USA: Marcel Dekker, 1998.

[3] R. J. Patton, P. M. Frank, and R. N. Clark, Eds., *Issues of Fault Diagnosis for Dynamic Systems*. New York, NY, USA: Springer-Verlag, 2000.

[4] M. Blanke, M. Kinnaert, J. Lunze, and M. Staroswiecki, *Diagnosis and Fault-Tolerant Control*. New York, NY, USA: Springer-Verlag, 2003.

[5] R. Isermann, *Fault Diagnosis Systems*. New York, NY, USA: Springer-Verlag, 2006.

[6] J. Stoustrup, M. Grimble, and H. Niemann, "Design of integrated systems for the control and detection of actuator/sensor faults," *Sensor Rev.*, vol. 17, no. 2, pp. 138–149, 1997.

[7] G. G. Yen and L. Ho, "Online multiple-model-based fault diagnosis and accommodation," *IEEE Trans. Ind. Electron.*, vol. 50, no. 2, pp. 296–312, Apr. 2003.

[8] S. X. Ding, "Integrated design of feedback controllers and fault detectors," *Ann. Rev. Control*, vol. 33, no. 2, pp. 124–135, Dec. 2009.

[9] K. Rothenhagen and F. W. Fuchs, "Doubly fed induction generator model-based sensor fault detection and control loop reconfigurat," *IEEE Trans. Ind. Electron.*, vol. 56, no. 10, pp. 4229–4238, Oct. 2009.

[10] R. Muradore and P. Fiorini, "A PLS-based statistical approach for fault detection and isolation of robotic manipulators," *IEEE Trans. Ind. Electron.*, vol. 59, no. 8, pp. 3167–3175, Aug. 2012.

[11] H. Dong, Z. Dong, and H. Gao, "Fault detection for Markovian jump systems with sensor saturations and randomly varying nonlinearities," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 59, no. 10, pp. 2354–2362, Oct. 2012.

[12] S. A. Arogeti, D. Wang, C. B. Low, and M. Yu, "Fault detection isolation and estimation in a vehicle steering system," *IEEE Trans. Ind. Electron.*, vol. 59, no. 12, pp. 4810–4820, Dec. 2012.

[13] Y. Fujimoto and T. Sekiguchi, "Fault-tolerant configuration of distributed discrete controllers," *IEEE Trans. Ind. Electron.*, vol. 50, no. 1, pp. 86–93, Feb. 2003.

[14] A. M. S. Mendes and A. J. M. Cardoso, "Fault-tolerant operating strategies applied to three-phase induction-motor drives," *IEEE Trans. Ind. Electron.*, vol. 53, no. 6, pp. 1807–1817, Dec. 2006.

[15] A. Akrad, M. Hilairet, and D. Diallo, "Design of a fault-tolerant controller based on observers for a PMSM drive," *IEEE Trans. Ind. Electron.*, vol. 58, no. 4, pp. 1416–1427, Apr. 2011.

[16] K. Nguyen-Duy, T. Liu, D. Chen, and J. Y. Hung, "Improvement of matrix converter drive reliability by online fault detection and a fault-tolerant switching strategy," *IEEE Trans. Ind. Electron.*, vol. 59, no. 1, pp. 244–256, Jan. 2012.

[17] J. W. Bennett, G. J. Atkinson, B. C. Mecrow, and D. J. Atkinson, "Fault-tolerant design considerations and control strategies for aerospace drives," *IEEE Trans. Ind. Electron.*, vol. 59, no. 5, pp. 2049–2058, May 2012.

[18] S. Huang, K. K. Tan, and T. H. Lee, "Fault diagnosis and fault-tolerant control in linear drives using the Kalman filter," *IEEE Trans. Ind. Electron.*, vol. 59, no. 11, pp. 4285–4292, Nov. 2012.

[19] K. Zhou and Z. Ren, "A new controller architecture for high performance, robust, and fault-tolerant control," *IEEE Trans. Autom. Control*, vol. 46, no. 10, pp. 1613–1618, Oct. 2001.

[20] S. X. Ding, G. Yang, P. Zhang, E. L. Ding, T. Jeinsch, N. Weinhold, and M. Schultalbers, "Feedback control structures, embedded residual signals, and feedback control schemes with an integrated residual access," *IEEE Trans. Control Syst. Technol.*, vol. 18, no. 2, pp. 352–367, Mar. 2010.

[21] S. X. Ding, Y. Wang, P. Zhang, Y. Yang, and E. L. Ding, "Data-driven design of fault-tolerant control systems," in *Proc. IFAC Symp. Safeprocess*, 2012, pp. 1323–1328.

[22] K. Zhou, J. C. Doyle, and K. Glover, *Robust and Optimal Control*. Englewood Cliffs, NJ, USA: Prentice-Hall, 1996.

[23] W. Favoreel, B. De Moor, and P. Van Overschee, "Subspace state space system identification for industrial processes," *J. Process Control*, vol. 10, no. 2/3, pp. 149–155, Apr. 2000.

[24] S. J. Qin, "An overview of subspace identification," *Comput. Chemical Eng.*, vol. 30, no. 10–12, pp. 1502–1513, Sep. 2006.

[25] P. Van Overschee and B. De Moor, *Subspace Identification for Linear Systems*. Dordrecht, The Netherlands: Kluwer, 1996.

[26] P. Zhang and S. X. Ding, "On fault detection in linear discrete-time, periodic, and sampled-data systems (survey)," *J. Control Sci. Eng.*, vol. 2008, pp. 1–19, 2008.

[27] V. Venkatasubramanian, R. Rengaswamy, K. Yin, and S. N. Kavuri, "A review of process fault detection and diagnosis. Part I: Quantitative model-based methods," *Comput. Chemical Eng.*, vol. 27, no. 3, pp. 293–311, Mar. 2003.

[28] S. X. Ding, P. Zhang, A. Naik, E. L. Ding, and B. Huang, "Subspace method aided data-driven design of fault detection and isolation systems," *J. Process Control*, vol. 19, no. 9, pp. 1496–1510, Oct. 2009.

[29] B. A. Francis, *A Course in H-Infinity Control Theory*. Berlin, Germany: Springer-Verlag, 1987.

[30] S. X. Ding, *Model-based Fault Diagnosis Techniques: Design Schemes, Algorithms, and Tools*. Berlin, Germany: Springer-Verlag, 2008.

[31] S. X. Ding, S. Yin, P. Zhang, E. L. Ding, and A. Naik, "An approach to data-driven adaptive residual generator design and implementation," in *Proc. IFAC Symp. Safeprocess*, 2009, pp. 941–946.

[32] J. S. Arora, *Introduction to Optimum Design*, 2nd ed. Amsterdam, The Netherlands: Elsevier, 2004.

[33] I. M. Y. Mareels and J. W. Polderman, *Adaptive Control Systems: An Introduction*. Basel, Germany: Birkhäuser Verlag, 1996.

[34] J. Downs and E. Fogel, "A plant-wide industrial process control problem," *Comput. Chemical Eng.*, vol. 17, no. 3, pp. 245–255, Mar. 1993.

[35] L. H. Chiang, E. L. Russell, and R. D. Braatz, "Fault diagnosis in chemical processes using fisher discriminant analysis, discriminant partial least squares, and principal component analysis," *Chemometrics Intell. Lab. Syst.*, vol. 50, no. 2, pp. 243–252, Mar. 2000.

[36] N. L. Ricker, "Decentralized control of the Tennessee Eastman challenge process," *J. Process Control*, vol. 6, no. 4, pp. 205–221, Aug. 1996.

[37] S. Yin, S. X. Ding, A. Haghani, H. Hao, and P. Zhang, "A comparison study of basic data-driven fault diagnosis and process monitoring methods on the benchmark Tennessee Eastman process," *J. Process Control*, vol. 22, no. 9, pp. 1567–1581, Oct. 2012.


**Shen Yin** (M'12) received the B.E. degree in automation from Harbin Institute of Technology, Harbin, China, in 2004, the M.Sc. degree in control and information system and the Ph.D. degree in electrical engineering and information technology from University of Duisburg-Essen, Duisburg, Germany.

His research interests are model based and data-driven fault diagnosis, fault tolerant control and big data focused on industrial electronics applications.

**Hao Luo** received the B.E. degree in electrical engineering from Xi'an Jiaotong University, Xi'an, China, in 2007, the M.Sc. degree in electrical engineering and information technology from University of Duisburg-Essen, Duisburg, Germany, in 2012. He is currently working toward the Ph.D. degree at the Institute for Automatic Control and Complex Systems (AKS) at the University of Duisburg-Essen.

His research interests include model based and data-driven fault diagnosis, fault-tolerant systems and their applications on industrial systems.

**Steven X. Ding** received the Ph.D. degree in electrical engineering from the Gerhard-Mercator University of Duisburg, Duisburg, Germany, in 1992.

From 1992 to 1994, he was an R&D Engineer at Rheinmetall GmbH. From 1995 to 2001, he was a Professor of control engineering at the University of Applied Science Lausitz in Senftenberg, Germany, and served as Vice President of this university during 1998–2000. Since 2001, he has been a Professor of control engineering and the head of the Institute for Automatic Control and Complex Systems (AKS) at the University of Duisburg-Essen, Germany. His research interests are model-based and data-driven fault diagnosis, fault-tolerant systems and their application in industry with a focus on automotive systems and chemical processes.