INTERNATIONAL JOURNAL OF ROBUST AND NONLINEAR CONTROL
Int. J. Robust. Nonlinear Control 2012; **22**:68–88
Published online 25 May 2011 in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/rnc.1759

Data-based monitoring and reconfiguration of a distributed model predictive control system

David Chilin[1], Jinfeng Liu[1], James F. Davis[1] and Panagiotis D. Christofides[1, 2, *, †]

[1]*Department of Chemical and Biomolecular Engineering, University of California, Los Angeles, CA 90095-1592, U.S.A.*
[2]*Department of Electrical Engineering, University of California, Los Angeles, CA 90095-1592, U.S.A.*

SUMMARY

In this work, we develop a data-based monitoring and reconfiguration system for a distributed model predictive control system in the presence of control actuator faults. Specifically, we first design fault detection filters and filter residuals, which are computed via exponentially weighted moving average, to effectively detect faults. Then, we propose a fault isolation approach that uses adaptive fault isolation time windows whose length depends on the rate of change of the fault residuals to quickly and accurately isolate actuator faults. Simultaneously, we estimate the magnitudes of the faults using a least-squares method and based on the estimated fault values, we design appropriate control system reconfiguration (fault-tolerant control) strategies to handle the actuator faults and maintain the closed-loop system state within a desired operating region. A nonlinear chemical process example is used to demonstrate the approach. Copyright © 2011 John Wiley & Sons, Ltd.

Received 19 November 2010; Revised 5 April 2011; Accepted 14 April 2011

KEY WORDS: data-based methods; fault detection and isolation; distributed model predictive control; fault-tolerant control; chemical process control

1. INTRODUCTION

In chemical process industry, there is a trend toward 'smart' plants that are capable of highly automated control with decision making at the plant level taking into account environmental, health, safety and economic considerations [1, 2]. Along with the move toward more automated plant operation, improved methods of fault detection, isolation and handling are necessary due to the issues raised by automation itself. Despite the many benefits of automatic process control, increased complexity and instrumentation can cause automated plants to become more susceptible to control system failures. Abnormal situations cost U.S. industries over $20 billion each year [3]. Fault-tolerant control (FTC) is a field that has received a significant amount of attention recently in the context of chemical process control and operations as a means for avoiding disaster in the case of a fault; see, for example, [4, 5]. FTC attempts to reconfigure a process control system upon detection of a fault and isolation of its cause, in order to preserve closed-loop system stability and performance. Fault detection and isolation (FDI) methods can generally be divided into two broad categories: model-based and data-based. Model-based FDI methods generally rely on mathematical models of the process developed either from first principles or from system identification. The data generated from the model are compared with measured data from the physical system to create

[*] Correspondence to: Panagiotis D. Christofides, Department of Chemical and Biomolecular Engineering, University of California, Los Angeles, CA 90095-1592, U.S.A.

[†] E-mail: pdc@seas.ucla.edu

residuals that relate to specific faults. With an accurate process model, it is possible to accomplish FDI for specific process structures (see, for example, [6, 7]). Data-based methods, on the other hand, rely on real-time process measurements and historical process measurement data in order to perform FDI. It is then possible, particularly for linear process systems, to extract information about the fault by comparing the location of the system in the state-space with the past faulty behavior (e.g. [8]). While many of these methods have been successful in achieving fault detection, data-based fault isolation remains a difficult task, particularly for nonlinear processes. The reader may refer to [9, 10] for a review of model-based and data-based FDI methods.

On the other hand, within process control, there is a trend toward distributed control architectures in which distributed optimization-based controllers compute the manipulated inputs in a coordinated fashion (e.g. [11–14]). Model predictive control (MPC) is a natural control framework to deal with the design of cooperative, distributed control systems because of its ability to handle input and state constraints, and also because it can account and compensate for the actions of other actuators. In the recent literature, several distributed MPC (DMPC) methods have been proposed that deal with the coordination of separate MPCs that communicate in order to obtain optimal input trajectories in a distributed manner; see, for example, [11, 12, 15, 16]. In our previous work [13] (see also [17]), we proposed a DMPC architecture with one-directional communication for nonlinear process systems. In this architecture, two separate MPCs designed via Lyapunov-based MPC (LMPC) were considered, in which one LMPC was used to guarantee the stability of the closed-loop system and the other LMPC was used to improve the closed-loop performance. In [14], we extended the DMPC architecture developed in [13] to include multiple distributed controllers and relaxed the requirement that one of the distributed controllers should be able to stabilize the closed-loop system. In [18], we developed an FDI and FTC system for the monitoring and reconfiguration of DMPC systems applied to general nonlinear processes in the presence of control actuator faults. The FDI and FTC system developed in [18] is based on process models and the assumption that once a faulty actuator is isolated, it can be reset to its zero state immediately.

In this paper, we take advantage of both process models and process measurements to develop a monitoring and reconfiguration system for a DMPC system in the presence of control actuator faults. Specifically, we first design fault detection filters and corresponding filter residuals, which are computed via exponentially weighted moving average (EWMA), to effectively detect actuator faults. Then, we propose a fault isolation approach that uses adaptive fault isolation time windows to quickly and accurately isolate actuator faults. Simultaneously, we estimate the magnitudes of the faults using a least-squares method and based on the estimated fault values, we design appropriate FTC strategies to handle the actuator faults and maintain the closed-loop system state within a desired operating region. A nonlinear chemical process example is used to demonstrate the approach.

2. NOTATION

The operator $|\cdot|$ is used to denote the absolute value of a scalar and the operator $\|\cdot\|$ is used to denote the Euclidean norm of a vector, whereas $\|\cdot\|_Q$ refers to the square of the weighted Euclidean norm, defined by $\|x\|_Q = x^{\mathrm{T}} Q x$ for all $x \in R^n$. The symbol $\mathrm{diag}(v)$ denotes a square diagonal matrix whose diagonal elements are the elements of the vector v.

3. PROBLEM FORMULATION AND PRELIMINARIES

3.1. Class of nonlinear systems

We consider nonlinear processes described by the following state-space model:

$$\dot{x}(t) = f(x) + \sum_{i=1}^{2} g_i(x)(u_i(t) + \tilde{u}_i(t)) \tag{1}$$

DOI: 10.1002/rnc

where $x \in R^n$ denotes the set of state variables, and $u_1 \in R^{m_1}$ and $u_2 \in R^{m_2}$ denote two sets of manipulated inputs, and $\tilde{u}_1 \in R^{m_1}$ and $\tilde{u}_2 \in R^{m_2}$ denote the unknown fault vectors for u_1 and u_2, respectively. We consider that $u_1 + \tilde{u}_1$ and $u_2 + \tilde{u}_2$ take values in non-empty convex sets $U_1 \in R^{m_1}$ and $U_2 \in R^{m_2}$, respectively. The convex sets U_1 and U_2 are defined as follows:

$$U_1 = \{u_1 + \tilde{u}_1 \in R^{m_1} : \|u_1 + \tilde{u}_1\| \leqslant u_1^{\max}\}$$

$$U_2 = \{u_2 + \tilde{u}_2 \in R^{m_2} : \|u_2 + \tilde{u}_2\| \leqslant u_2^{\max}\}$$

where $u_1^{\max}$ and $u_2^{\max}$ are the magnitudes of the input constraints. The system of Equation (1) can be re-written in a compact form as follows:

$$\dot{x}(t) = f(x) + g(x)(u(t) + \tilde{u}(t))$$

where $g(x) = [g_1(x) \;\; g_2(x)]$, $u(t) = [u_1(t)^{\mathrm{T}} \;\; u_2(t)^{\mathrm{T}}]^{\mathrm{T}}$ and $\tilde{u}(t) = [\tilde{u}_1(t)^{\mathrm{T}} \;\; \tilde{u}_2(t)^{\mathrm{T}}]^{\mathrm{T}}$. We also assume that U is a suitable composition of U_1 and U_2 such that $u + \tilde{u} \in U$ is equivalent to $u_1 + \tilde{u}_1 \in U_1$ and $u_2 + \tilde{u}_2 \in U_2$.

We use the variable $\tilde{u}_{f,j}$, $j = 1, \ldots, m_1 + m_2$, to model the possible faults associated with the jth element in the manipulated input vector u. Under fault-free operating conditions, we have $\tilde{u} = 0$, and hence, $\tilde{u}_{f,j} = 0$ for all $j = 1, \ldots, m_1 + m_2$. When fault j occurs, $\tilde{u}_{f,j}$ takes a nonzero value. We assume that f and g are locally Lipschitz vector functions and that $f(0) = 0$. This means that the origin is an equilibrium point for the fault-free system ($\tilde{u} = 0$ for all t) with $u = 0$. We also assume that the state x of the system is available synchronously and continuously at each sampling time.

Remark 1

Note that the use of two sets of manipulated inputs is adopted because of the implementation of a DMPC system to regulate the process; please see Section 3.2 for the control system design. An example of a chemical process system described by Equation (1) is given in Section 5.

3.2. Fault-free control system design

We assume that there exists a nonlinear control law $h(x)$ that determines u_1 (i.e. $u_1(t) = h(x(t))$) and renders the origin of the fault-free closed-loop system asymptotically stable with $u_2(t) = 0$. This assumption is essentially a standard stabilizability requirement made in all linear/nonlinear control methods and implies that there exists a Lyapunov function $V(x)$ of the system whose time derivative is always negative when $u_1 = h(x)$ is applied to the fault-free closed-loop system [19, 20].

We adopt the DMPC architecture introduced in [13] to design the fault-free control system. In this DMPC architecture, one LMPC is designed to determine u_1 and is responsible for the closed-loop stability; and another LMPC is designed to compute u_2 and to coordinate with u_1 to improve the closed-loop performance. We will refer to the two LMPCs computing u_1 and u_2 as LMPC 1 and LMPC 2, respectively. The two LMPCs are evaluated in a sequential fashion (i.e. LMPC 2 is first evaluated and then LMPC 1 is evaluated) at discrete time instants $\{t_{k \geqslant 0}\}$ with $t_k = t_0 + k\Delta$, $k = 0, 1, \ldots$, where t_0 is the initial time and Δ is a sampling time.

Specifically, the optimization problem of LMPC 2 at time t_k depends on the state measurement $x(t_k)$ and is formulated as follows:

$$\min_{u_2 \in S(\Delta)} \quad \int_{t_k}^{t_{k+N}} L(\tilde{x}(\tau), u_1(\tau), u_2(\tau))\,\mathrm{d}\tau \tag{2a}$$

$$\text{s.t.} \quad \dot{\tilde{x}}(t) = f(\tilde{x}(t)) + \sum_{i=1}^{2} g_i(\tilde{x}(t)) u_i(t) \tag{2b}$$

$$\tilde{x}(t_k) = x(t_k) \tag{2c}$$

$$u_1(t) = h(\tilde{x}(t_{k+j})) \quad \forall\, t \in [t_{k+j}, t_{k+j+1}) \tag{2d}$$

DOI: 10.1002/rnc

$$u_2(t) \in U_2 \tag{2e}$$

$$\frac{\partial V(x)}{\partial x} g_2(x(t_k)) u_2(t_k) \leqslant 0 \tag{2f}$$

with $L(\tilde{x}, u_1, u_2) = \|\tilde{x}(\tau)\|_{Q_c} + \|u_1(\tau)\|_{R_{c1}} + \|u_2(\tau)\|_{R_{c2}}$, where $S(\Delta)$ is the family of piece-wise constant functions with sampling period Δ, N is the prediction horizon, Q_c, R_{c1} and R_{c2} are positive-definite weighting matrices, $j = 0, \ldots, N-1$, $\tilde{x}$ is the predicted trajectory of the fault-free system with u_2 being the input trajectory computed by LMPC 2 of Equation (2) and u_1 being the nonlinear controller $h(x)$ applied in a sample-and-hold fashion. The optimal solution to this optimization problem is denoted as $u_2^*(t|t_k)$. This information is sent to LMPC 1. The optimization problem of LMPC 1 depends on $x(t_k)$ and the decision made by LMPC 2 (i.e. $u_2^*(t|t_k)$). Specifically, LMPC 1 is based on the following optimization problem:

$$\min_{u_1 \in S(\Delta)} \int_{t_k}^{t_{k+N}} L(\tilde{x}(\tau), u_1(\tau), u_2(\tau))\, \mathrm{d}\tau \tag{3a}$$

$$\text{s.t.} \quad \dot{\tilde{x}}(t) = f(\tilde{x}(t)) + \sum_{i=1}^{2} g_i(\tilde{x}(t)) u_i(t) \tag{3b}$$

$$\tilde{x}(t) = x(t_k) \tag{3c}$$

$$u_1(t) \in U_1 \tag{3d}$$

$$u_2(t) = u_2^*(t|t_k) \tag{3e}$$

$$\frac{\partial V(x)}{\partial x} g_1(x(t_k)) u_1(t_k) \leqslant \frac{\partial V(x)}{\partial x} g_1(x(t_k)) h(x(t_k)) \tag{3f}$$

The optimal solution to this optimization problem is denoted by $u_1^*(t|t_k)$.

Once both optimization problems are solved, the manipulated inputs of the DMPC system based on LMPC 1 and LMPC 2 are defined as follows:

$$u_1^L(t) = u_1^*(t|t_k) \quad \forall\, t \in [t_k, t_{k+1})$$

$$u_2^L(t) = u_2^*(t|t_k) \quad \forall\, t \in [t_k, t_{k+1}).$$

The fault-free closed-loop system of Equation (1) under this DMPC scheme with inputs defined by $u_1 = u_1^L$ and $u_2 = u_2^L$ maintains practical stability because of the two Lyapunov-based constraints of Equations (2f) and (3f) [13].

3.3. FTC considerations

The presence of the control action u_2 brings extra control flexibility to the closed-loop system which can be used to carry out FTC. Specifically, we further assume that the control input u_1 can be decomposed into two subsets (i.e. $u_1 = [u_{11}^{\mathrm{T}}\ \ u_{12}^{\mathrm{T}}]^{\mathrm{T}}$) and that there exists a nonlinear control law $h_2(x) = [h_{21}(x)^{\mathrm{T}}\ \ h_{22}(x)^{\mathrm{T}}]^{\mathrm{T}}$ which determines u_{11} and u_2 (i.e. $u_{11} = h_{21}(x)$ and $u_2 = h_{22}(x)$) and is able to asymptotically stabilize the fault-free closed-loop system with $u_{12} = 0$. This assumption implies that there exists a Lyapunov function $V_2(x)$ of the system whose time derivative is always negative when $u_{11} = h_{21}(x)$, $u_{12} = 0$ and $u_2 = h_{22}(x)$ are applied.

Based on $h_2(x)$, we can design a backup DMPC system (i.e. DMPC with LMPC 1 of Equation (5) and LMPC 2 of Equation (4) below) to manipulate u_{11} and u_2 to stabilize the closed-loop system following the results developed in [14]. We still design two LMPC controllers in the backup DMPC system. One LMPC is used to manipulate u_{11} and the other is used to manipulate u_2. In this backup DMPC system, the two LMPCs coordinate their actions to maintain the closed-loop stability. We refer to the LMPC manipulating u_{11} as the backup LMPC 1 and the LMPC manipulating u_2 as the backup LMPC 2. The two backup LMPCs are also evaluated in sequence.

DOI: 10.1002/rnc

The backup LMPC 2 optimizes u_2 and is designed as follows:

$$\min_{u_2 \in S(\Delta)} \quad \int_{t_k}^{t_{k+N}} L(\tilde{x}(\tau), u_1(\tau), u_2(\tau))\,\mathrm{d}\tau \tag{4a}$$

$$\text{s.t.} \quad \dot{\tilde{x}}(t) = f(\tilde{x}(t)) + g_1(\tilde{x}(t))[u_{11}(t)^{\mathrm{T}} u_{12}(t)^{\mathrm{T}}]^{\mathrm{T}} + g_2(\tilde{x}(t))u_2(t) \tag{4b}$$

$$\tilde{x}(t_k) = x(t_k) \tag{4c}$$

$$u_{11}(t) = h_{21}(\tilde{x}(t_{k+j})) \quad \forall\, t \in [t_{k+j}, t_{k+j+1}) \tag{4d}$$

$$u_{12}(t) = 0 \tag{4e}$$

$$u_2(t) \in U_2 \tag{4f}$$

$$\frac{\partial V_2(x)}{\partial x} g_2(x(t_k))u_2(t_k) \leqslant \frac{\partial V_2(x)}{\partial x} g_2(x(t_k))h_{22}(x(t_k)) \tag{4g}$$

The solution to the optimization problem of Equation (4) is denoted by $u_2^{b,*}(t|t_k)$. The backup LMPC 1 optimizes u_{11} and is designed as follows:

$$\min_{u_{11} \in S(\Delta)} \quad \int_{t_k}^{t_{k+N}} L(\tilde{x}(\tau), u_1(\tau), u_2(\tau))\,\mathrm{d}\tau \tag{5a}$$

$$\text{s.t.} \quad \dot{\tilde{x}}(t) = f(\tilde{x}(t)) + g_1(\tilde{x}(t))[u_{11}(t)^{\mathrm{T}} u12(t)^{\mathrm{T}}]^{\mathrm{T}} + g_2(\tilde{x}(t))u_2(t) \tag{5b}$$

$$\tilde{x}(t_k) = x(t_k) \tag{5c}$$

$$u_{11}(t) \in U_1 \tag{5d}$$

$$u_{12}(t) = 0 \tag{5e}$$

$$u_2 = u_2^{b,*}(t|t_k) \tag{5f}$$

$$\frac{\partial V_2(x)}{\partial x} g_1(x(t_k))[u_{11}(t)^{\mathrm{T}}\ \ 0^{\mathrm{T}}]^{\mathrm{T}} \leqslant \frac{\partial V_2(x)}{\partial x} g_1(x(t_k))[h_{21}(x(t_k))^{\mathrm{T}}\, 0^{\mathrm{T}}]^{\mathrm{T}} \tag{5g}$$

The solution to the optimization problem of Equation (5) is denoted by $u_{11}^{b,*}(t|t_k)$. The control inputs of the backup DMPC are defined as follows:

$$u_{11}^b(t) = u_{11}^{b,*}(t|t_k) \quad \forall\, t \in [t_k, t_{k+1})$$

$$u_{12}^b(t) = 0 \quad \forall\, t$$

$$u_2^b(t) = u_2^{b,*}(t|t_k) \quad \forall\, t \in [t_k, t_{k+1})$$

The fault-free closed-loop system of Equation (1) under the backup DMPC control with inputs defined by $u_{11} = u_{11}^b$, $u_{12} = 0$ and $u_2 = u_2^b$ maintains practical stability of the closed-loop system because of the Lyapunov-based constraints of Equations (4g) and (5g) [14].

To present the proposed method, in this work, we consider control actuator faults that can be detected by appropriate nonlinear dynamic fault filters via observing the evolution of the closed-loop system state. In order to isolate the occurrence of a fault, it is further required to assume that the control actuator in question is the only one influencing the observed 'faulty' states (i.e. each fault has a unique fault signature). For more discussions on systems having verifiable isolable structures, please see [7, 21].

4. FDI AND FTC SYSTEM DESIGN

In this section, we develop a combined model-based and data-based FDI and FTC method for the closed-loop system of Equation (1) under the DMPC of Equations (2) and (3).

4.1. Design of fault detection filters and residuals

The FDI filter is designed on the basis of the process model of Equation (1) under the DMPC system of Equations (2) and (3) used to control the process and it is used to predict the expected process dynamic state response in the absence of faults. These expected state values are compared with the corresponding real-time measured process states, forming the residuals (i.e. $r_p(t)$ defined below). Then, the residuals are compared with threshold values computed from closed-loop data under normal operation and a fault is declared (detected) when the residual values exceed the thresholds. Fault isolation subsequently is carried out by comparing the fault signature (i.e. what residuals exceed their thresholds) with the signature of process/fault interaction of the various explicitly modeled faults computed from the process model. The DMPC system of Equations (2) and (3) is the control system for the fault-free closed-loop system. We first design an FDI scheme to detect faults in this control system. In this FDI scheme, a filter is designed for each state and the design of the filter for the pth, $p=1,\ldots,n$, state in the system state vector x is as follows [7]:

$$\dot{\hat{x}}_p(t) = f_p(X_p) + g_{1p}(X_p)u_1^L(t) + g_{2p}(X_p)u_2^L(t) \tag{6}$$

where $\hat{x}_p$ is the filter output for the pth state, f_p, g_{1p} and g_{2p} are the pth components of the vector functions f, g_1 and g_2, respectively. The state X_p is obtained from both the actual state measurements, x, and the filter output, $\hat{x}_p$, as follows:

$$X_p(t) = [x_1(t),\ldots,x_{p-1}(t),\hat{x}_p(t),x_{p+1}(t),\ldots,x_n(t)]^{\mathrm{T}}$$

Note that in the filter of Equation (6), the control inputs $u_1^L(t)$ and $u_2^L(t)$ are determined by LMPC 1 of Equation (3) and LMPC 2 of Equation (2) as applied to the actual process based on the state X_p, and are updated every control sampling time Δ (i.e. the sampling time instants $\{t_{k\geqslant 0}\}$).

The FDI filters are only initialized at $t=0$ such that $\hat{x}_p(0)=x_p(0)$. The information generated by the filters provides a fault-free estimate of the actual system state at any time t and allows easy detection of the actual system state deviations due to faults. For each state associated with a filter, an FDI residual is defined as follows:

$$r_p(t) = \|\hat{x}_p(t) - x_p(t)\|$$

with $p=1,\ldots,n$. The residual r_p is computed continuously because $\hat{x}_p(t)$ is known for all t and the state measurement, x, is also available for all t. If no fault occurs, the filter states track the system states. In this case, the dynamics of the system states and the FDI filter states are identical, so $r_p(t)=0$ for all times.

In the practical case where sensor measurement noise and process noise are present, the residual will be nonzero even without an actuator fault. In order to reduce the influence of process noise on fault detection, we define a weighted residual $r_{E,p}$, $p=1,\ldots,n$, for each residual r_p, calculated at discrete time instants $\{t_{i\geqslant 0}\}$ with $t_i = t_0 + i\Delta_r$, $i=0,1,2,\ldots$ The weighted residual is calculated using an EWMA method as follows [22]:

$$r_{E,p}(t_i) = \lambda r_p(t_i) + (1-\lambda) r_{E,p}(t_{i-1}) \tag{7}$$

with $r_{E,p}(t_0) = r_p(t_0)$ and the weighting factor $\lambda \in (0,1]$. The parameter λ determines the rate at which previous data enter into the calculations of the weighted residual. When $\lambda=1, r_{E,p}$ is equivalent to r_p. The benefit of using EWMA residuals is their ability to better capture smaller drifts in the system and to provide protection against occasional spikes. The value of λ is typically set between 0.2 and 0.5 depending on the sensitivity and responsiveness desired [22]. All further mention of residuals will be with reference to the EWMA residuals.

DOI: 10.1002/rnc

Also, due to sensor measurement and process noise, fault detection thresholds are necessary so that a fault is declared only when a residual exceeds its specific threshold value. The thresholds are based on historical process variance data under no fault (normal) operating conditions and chosen to the desired degree of confidence to quickly detect possible faults. In some cases, the residual may deviate temporarily due to normal process variance and should not be interpreted as a fault. In these cases, it is important to properly confirm that the residual is deviating because of a fault by waiting a specified amount of time. By waiting, we improve our confidence that a fault has occurred and reduce the probability of false alarms. In the detection of a fault, three threshold values for each EWMA residual are used. The threshold values for the EWMA residual, $r_{E,p}$, are calculated as discussed in [22] using the following formula:

$$\sigma_{p,k} = \bar{r}_p + k s_p \sqrt{\frac{\lambda}{2-\lambda}} \tag{8}$$

where $k = 3, 4, 5$ are the weighting factors used, $\bar{r}_p$ and s_p are the mean value and standard deviation of the pth residual (r_p) based on historical fault-free operation data of the closed-loop system, respectively. Specifically, the historical process operation data in the application discussed in the following section were obtained by running the closed-loop system under fault-free conditions for a simulated period of 5 h and collecting the system state and residual values in order to compute the average value and standard deviation for each state and each residual. These values where then used in conjunction with Equation (8) to compute the threshold values ($\sigma_{p,k}$) for each EWMA residual.

4.2. Fault detection and isolation using adaptive windows

In this subsection, we augment our previous FDI system [7] to include an adjustable time window based on the rate of change in the residual with the goals of reducing the probability of false alarms and false isolation, and achieving a quicker fault recovery response.

On the occurrence of a fault, certain residuals directly associated with the fault will immediately become nonzero at different rates (or in the case where process noise and measurement noise are present their thresholds will be exceeded at different times depending on the fault's magnitude). An improvement over previous work is the use of EWMA residuals in combination with adjustable fault isolation time windows.

When there is a residual that exceeds its second threshold and stays above it for a time period Δt_d, then a fault is declared. For example, if $r_{E,p}$ exceeds $\sigma_{p,4}$ at time $t_{\sigma_{p,4}}$ and stays above $\sigma_{p,4}$ from $t_{\sigma_{p,4}}$ to $t_{\sigma_{p,4}} + \Delta t_d$, then a fault is declared. The waiting time Δt_d is used to reduce the incidence of false alarms and, in particular, intermittent spikes.

Fault isolation is carried out simultaneously with fault detection. We define a fault signature as a set $I_p = [i_1, i_2, \ldots, i_n]$, where $i_p = 1$ if the residual $r_{E,p} \geqslant \sigma_{p,3}$, otherwise $i_p = 0$ for a particular fault $\tilde{u}_{f,p}$. We consider that the system has an isolable structure which implies that each of the possible faults has a unique fault signature; that is, $I_p \neq I_q$ with $p \neq q$. Based on the rate of change in the first residual that exceeds its second threshold, a time window over which a fault may be isolated is calculated. If there is no residual that exceeds its third threshold within the time window, the fault is isolated at the end of the time window. The isolated fault has a signature composed of all the residuals that exceed their second thresholds. If there is at least one residual that exceeds its third threshold within the fault isolation time window, a new fault isolation time window is calculated and the fault is isolated at the end of this new time window. For example, if $r_{E,p}$ is the first residual that exceeds its threshold $\sigma_{p,4}$, a time window, Δt_p, is calculated as follows:

$$\Delta t_p = w(t_{\sigma_{p,4}} - t_{\sigma_{p,3}}) \tag{9}$$

where w is a constant or a complex function of the model and its current state, and $t_{\sigma_{p,4}}$ and $t_{\sigma_{p,3}}$ are the time instants the residual $r_{E,p}$ exceeds $\sigma_{p,4}$ and $\sigma_{p,3}$, respectively. If from $t_{\sigma_{p,4}}$ to $t_{\sigma_{p,4}} + \Delta t_p$, there is no residual that exceeds its third threshold, the fault is isolated at time $t_{\sigma_{p,4}} + \Delta t_p$ with a signature composed of all the residuals whose values exceed their second thresholds. If from $t_{\sigma_{p,4}}$ to $t_{\sigma_{p,4}} + \Delta t_p$, there is at least one residual that exceed its third threshold, for example, $r_{E,q}$

DOI: 10.1002/rnc

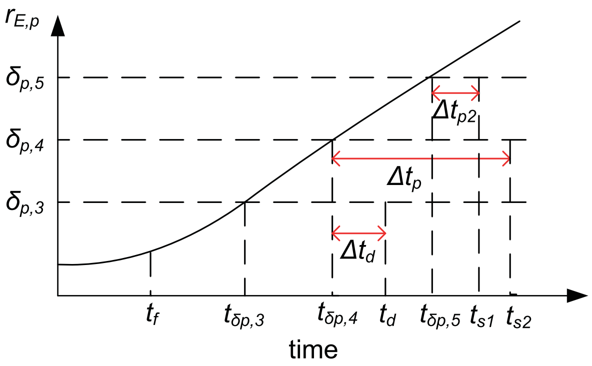

Figure 1. An example of residual evolution after the occurrence of a fault at t_f that affects residual $r_{E,p}$.

exceeds $\sigma_{q,5}$ at time $t_{\sigma_{q,5}}$, then a new fault isolation time window, Δt_q, is calculated following Equation (9) based on the change rate of $r_{E,q}$ from $t_{\sigma_{q,4}}$ to $t_{\sigma_{q,5}}$, and the fault is isolated at $\min\{t_{\sigma_{p,4}}+\Delta t_p, t_{\sigma_{q,5}}+\Delta t_q\}$.

An example with one residual is shown in Figure 1. In Figure 1, a fault occurs at time t_f, which drives a residual $r_{E,p}$ to go up. The residual exceeds its first and second thresholds at time $t_{\sigma_{p,3}}$ and $t_{\sigma_{p,4}}$, respectively. Once the residual exceeds its second threshold $\sigma_{p,4}$ and stays above it for a short waiting time Δt_d, a fault is declared at time t_d ($t_d = t_{\sigma_{p,4}} + \Delta t_d$). At the time $t_{\sigma_{p,4}}$, a fault isolation time window Δp is also calculated. Because the residual continues to increase and exceeds its third threshold, $\sigma_{p,5}$, at time $t_{\sigma_{p,5}}$, a new isolation window Δt_{p2} is calculated. The fault is isolated at time $t_{s1} = \min\{t_{s1}, t_{s2}\}$. The isolated fault has a signature composed of all the residuals that exceed their second threshold.

4.3. Fault parameter estimation

After a fault has been isolated, the FTC system must know the magnitude of the fault in order to target the corresponding new operating point and properly stabilize the system in the presence of the fault. To simplify the description of the proposed method, we consider faults of constant magnitudes in this work; however, faults with slowly time-varying values can be handled using the proposed method in a straightforward manner.

When a residual ($r_{E,p}$) exceeds its first threshold ($\sigma_{p,3}$), we begin to collect the sampled system states as well as the actual control inputs applied to the system. When the fault is confirmed and isolated, a least-square optimization problem is solved to get an estimate of the magnitude of the fault based on the sampled system states and the actual control inputs. Specifically, we collect the sampled system states, $x(t)$, and record the actual control inputs (i.e. $u_1(t) = u_1^L(t)$ and $u_2(t) = u_2^L(t)$) applied to the system for $t_{\sigma_{p,3}}$ to the fault isolation time t_{isolate} with a sampling time Δ_e. The magnitude of the fault $\tilde{u}_{f,j}$ is estimated by solving the following optimization problem:

$$\min_{\tilde{u}_{f,j}} \sum_{i=0}^{M} (x(t_f + i\Delta_e) - \tilde{x}(t_f + i\Delta_e))^2 \tag{10a}$$

$$\text{s.t.} \quad \dot{\tilde{x}}(t) = f(\tilde{x}(t)) + g(\tilde{x}(t))(u^L(t) + d) \tag{10b}$$

$$\tilde{x}(t_f) = x(t_f) \tag{10c}$$

where $u^L(t) = [u_1^L(t)^{\mathrm{T}} \;\; u_2^L(t)^{\mathrm{T}}]^{\mathrm{T}}$ is the actual control inputs that have been applied to the closed-loop system from $t_{\sigma_{p,3}}$ to t_{isolate}, M is the maximum integer satisfying $M\Delta_e \leqslant t_{\text{isolate}} - t_{\sigma_{p,3}}$, $d = [0 \;\ldots\; \tilde{u}_{f,j} \;\ldots\; 0]^{\mathrm{T}}$ is the fault vector, and $x(t_f)$ is the system state at the fault detection time. The solution to the optimization problem of Equation (10) is denoted by $\tilde{u}^*_{f,j}$, which is an optimal estimate of the actual fault $\tilde{u}_{f,j}$ from a least-square point of view.

DOI: 10.1002/rnc

4.4. FTC strategies

When a fault is detected, isolated and the magnitude of the fault is estimated, suitable FTC strategies can be carried out to keep the closed-loop system state within a desired operating region. Because of the fault, the origin (the operating point of the fault-free system) may not be achievable because of the input constraints and the system structure. In this case, we may operate the system at a new operating point within the desired operating region. To determine the new operating point x_s, we propose to solve an optimization problem. Specifically, when the fault is $\tilde{u}^*_{f,j}$, the new operating point, x_s, is obtained by solving the following optimization problem:

$$\min_{x_s, u_s} \quad \|x_s\|_S \tag{11a}$$

$$\text{s.t.} \quad f(x_s)+g(x_s)(u_s+d)=0 \tag{11b}$$

$$u_s+d \in U \tag{11c}$$

$$x_s \in X \tag{11d}$$

where S is a positive weighting matrix, $d=[0 \ \ldots \ \tilde{u}^*_{f,j} \ \ldots \ 0]^{\mathrm{T}}$ and X denotes the desired operating state space region. The objective of the above optimization problem is to find an operating point within the desired operating state space region such that the distance (measured by weighted Euclidean norm) between the new operating point and the origin is minimized. We assume that the optimization problem of Equation (11) is always feasible which implies that we can always find the new operating point x_s and the corresponding new steady-state control input values $u_s=[u^{\mathrm{T}}_{1s} \ u^{\mathrm{T}}_{2s}]^{\mathrm{T}}$.

Note that the proposed method is only one of many possible approaches to determine the new operating point in the case of a fault. The basic idea of the proposed method is to find a new operating point that stays as close as possible to the original operating point (i.e. the origin $x=0$).

Once we find the new operating point x_s, we proceed to design the FTC strategies for the fault-free DMPC system (see Equations (2) and (3)) in the presence of actuator faults. In general, when there is a fault in the control system, it is impossible to carry out FTC unless there is another backup control loop. However, in the fault-free DMPC system, because of the extra control flexibility brought into the whole system by u_2 (LMPC 2), it is possible in some cases to carry out FTC without activating new control actuators.

When there is a persistent fault in the loop of u_2 which is denoted by d_2, and the fault can be detected and isolated in a reasonable time frame, it is possible to switch off the controller LMPC 2 and only use u_1 in the control system. When LMPC 2 is switched off from the closed-loop system, u_2 is set by the fault value (i.e. $u_2=d_2$); and in the DMPC scheme of Equations (2) and (3), only LMPC 1 is evaluated at each sampling time. In order to maintain the stability of the closed-loop system, the design of LMPC 1 will need to be updated with the new operating point and its corresponding new steady-state control input values (i.e. the cost function $L(x, u_1, u_2)$ needs to be updated with x_s and u_s in a way such that $L(x_s, u_{1s}, u_{2s})=0$), and updated with the fault magnitude information (i.e. $u_2=d_2$); the design of $h(x)$ also need to be updated with the new steady-state information. The control inputs determined by the updated LMPC 1 will be referred to as $u'_1(x)$. This FTC strategy will maintain the closed-loop stability if implemented quickly such that the state of the closed-loop system is still within the stability region of the backup controllers and parameter estimation is sufficiently accurate, however, the performance of the closed-loop system may degrade to some extent.

When there is a fault in the subset u_{12}, which is denoted by d_1, the FTC strategy would shut down the control action of u_{12} and reconfigure the DMPC algorithms to the backup DMPC of Equations (4) and (5) to manipulate u_{11} and u_2 to control the process. In order to maintain the stability of the closed-loop system, the designs of the two backup LMPCs and the design of $h_2(x)$ needs to be updated with the new operating point and the corresponding new steady-state control input values; as well as being updated with the fault magnitude information. The control inputs determined by the updated designs will be referred to as $u''_1(x)$ and $u''_2(x)$.

However, when there is a fault in the subset u_{11}, it is impossible to successfully carry out FTC without activating backup actuators within the DMPC systems for the class of nonlinear systems considered in this work.

The FTC switching rules for the system of Equation (1) within the DMPC system of Equations (2) and (3) are described as follows:

(1) When a fault in the actuator associated with u_2 is isolated at t_f, the FTC switching rule is:

$$u_1(t) = \begin{cases} u_1^L(x), & t \leqslant t_f \\ u_1', & t > t_f \end{cases} \tag{12a}$$

$$u_2(t) = \begin{cases} u_2^L(x), & t \leqslant t_f \\ d_2, & t > t_f \end{cases} \tag{12b}$$

(2) When a fault in the actuator associated with u_{12} is detected at t_f, the FTC switching rule is:

$$u_1(t) = \begin{cases} u_1^L(x), & t \leqslant t_f \\ \begin{bmatrix} u_{11}''(x) \\ d_1 \end{bmatrix}, & t > t_f \end{cases} \tag{13a}$$

$$u_2(t) = \begin{cases} u_2^L(x), & t \leqslant t_f \\ u_2''(x), & t > t_f \end{cases} \tag{13b}$$

Remark 2
While in this work, the full process state is assumed to be measured in real-time, it would be possible to implement the proposed approach using partial measurements of the full process state vector, provided that the available measurements in such a case allow to detect, isolate and estimate the magnitude of the fault.

Remark 3
In the present work, a DMPC system involving two distributed LMPCs which are solved in sequence is adopted to control the process of Equation (1). The benefit of solving the two LMPCs sequentially is that a single communication (i.e. LMPC 2 is solved first and sends its optimal trajectory to LMPC 1 which then calculates its own trajectory) leads to control actions that guarantee closed-loop stability due to the Lyapunov constraints while simultaneously solving LMPC 1 and LMPC 2 may require multiple communications between the two controllers to achieve a similar performance level.

Remark 4
Because of the structure of the system considered, it is possible in general that the origin is outside the accessible region of the system at the time of DMPC reconfiguration after FDI has occurred (i.e. the reconfigured DMPC cannot stabilize the closed-loop system at the origin). What can be done and is done in this case in the present work (see the following section) is to reconfigure the backup control system to try to maintain the closed-loop system within a region as close to the origin as possible.

5. APPLICATION TO A REACTOR–SEPARATOR PROCESS

5.1. Process description and modeling

The process considered in this study is a three vessel, reactor–separator system consisting of two CSTRs and a flash tank separator as shown in Figure 2. Its detailed description and modeling can

DOI: 10.1002/rnc

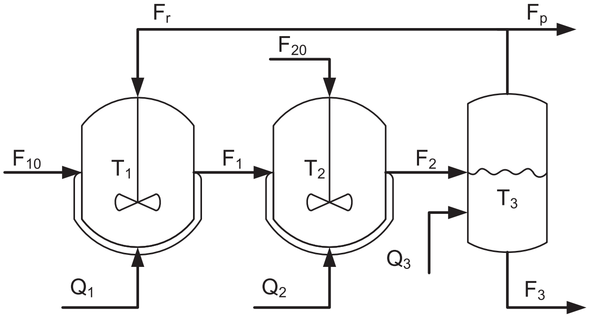

Figure 2. Two CSTRs and a flash tank with recycle stream.

Table I. The desired operating steady-state x_s.

T_1	C_{A1}	C_{B1}	C_{C1}
370 (K)	3.32 (kmol/m^3)	0.17 (kmol/m^3)	0.04 (kmol/m^3)
T_2	C_{A2}	C_{B2}	C_{C2}
435 (K)	2.75 (kmol/m^3)	0.45 (kmol/m^3)	0.11 (kmol/m^3)
T_3	C_{A3}	C_{B3}	C_{C3}
435 (K)	2.88 (kmol/m^3)	0.50 (kmol/m^3)	0.12 (kmol/m^3)

Table II. The steady-state input values.

Q_{1s}	Q_{2s}	Q_{3s}	F_{20s}
0 (kJ/h)	0 (kJ/h)	0 (kJ/h)	5 (m^3/h)

be found in [18]. Sensor and process noise were added to the simulations. The desired operating steady-state is the unstable steady state, x_s, whose values are shown in Table I.

For this process, we have two sets of manipulated inputs. The first set of manipulated inputs is the heat injected to or removed from the three vessels, that is $u_1 = [Q_1 - Q_{1s} \;\; Q_2 - Q_{2s} \;\; Q_3 - Q_{3s}]^{\mathrm{T}}$; the second set includes the inlet flow rate to vessel 2, that is $u_2 = F_{20} - F_{20s}$. The variables Q_{1s}, Q_{2s}, Q_{3s} and F_{20s} denote the steady-state input values of the inputs whose values are shown in Table II. The control inputs are subject to the constraints $|Q_i - Q_{is}| \leqslant u_1^{\max} = 10^6\,\mathrm{kJ/h}$ $(i = 1, 2, 3)$ and $|F_{20} - F_{20s}| \leqslant u_2^{\max} = 5\,\mathrm{m}^3/\mathrm{h}$.

In the design of the fault-free DMPC system for the process, we consider a quadratic Lyapunov function $V(x) = x^{\mathrm{T}} P x$ with $P = \mathrm{diag}([20 \;\; 10^3 \;\; 10^3 \;\; 10^3 \;\; 10 \;\; 10^3 \;\; 10^3 \;\; 10^3 \;\; 10 \;\;\; 10^3 \;\; 10^3 \;\; 10^3])$ and design the controller $h(x)$ as three PI controllers with proportional gains $K_{p1} = K_{p2} = K_{p3} = 8000$ and integral time constants $\tau_{I1} = \tau_{I2} = \tau_{I3} = 10$ based on the measurements of T_1, T_2 and T_3, respectively. Note that, in the absence of process noise and measurement noise, this design of $h(x)$ manipulating u_1 can stabilize the closed-loop system asymptotically without the use of u_2. Based on $h(x)$ and $V(x)$, we design LMPC 1 following Equation (3) to determine u_1 and LMPC 2 following Equation (2) to determine u_2. In the design of the LMPCs, the weighting matrices are chosen to be $Q_c = P$, $R_1 = \mathrm{diag}([(5\;5\;5) \times 10^{-12}])$ and $R_2 = 100$. The horizon for the optimization problem is $N = 4$ with a time step of $\Delta = 0.05\,\mathrm{h}$.

In addition, the set of control inputs u_1 can be divided into two subsets, $u_{11} = [Q_1 - Q_{1s} \;\; Q_3 - Q_{3s}]^{\mathrm{T}}$ and $u_{12} = Q_2 - Q_{2s}$. The input combination u_{11} and u_2 is able to stabilize the closed-loop system which can be used as the input configuration of the backup DMPC system of Equations (4) and (5). In order to design the backup DMPC, we need to design a second Lyapunov-based controller $h_2(x)$ which manipulates u_{11} and u_2. We also design h_2 through PI control law with proportional gains $K_{p1}^b = K_{p2}^b = 8000$, $K_{p3}^b = -0.3$ and integral time constants $\tau_{I1}^b = \tau_{I2}^b = \tau_{I3}^b = 10$ based on the measurements of T_1, T_3 and T_2, respectively. The control design h_2 can stabilize the

Table III. EWMA residual means and standard deviations.

$\bar{r}_{T_2}$	$\bar{r}_{C_{A2}}$	$\bar{r}_{C_{B2}}$	$\bar{r}_{C_{C2}}$
0.664900	0.013944	0.003421	0.003980
s_{T_2}	$s_{C_{A2}}$	$s_{C_{B2}}$	$s_{C_{C2}}$
0.464139	0.010351	0.002810	0.002960

closed-loop system asymptotically with $Q_2=0$ in the absence of process noise and measurement noise. In the design of the backup DMPC system, we choose $V_2(x)=V(x)$.

In order to perform FDI for the reactor–separator system, we construct the FDI filters for the states affected directly by the four manipulated inputs as in Equation (6). The states affected directly by the manipulated inputs are T_1, C_{A2}, C_{B2}, C_{C2}, T_2 and T_3. The FDI residuals take the following form:

$$\begin{aligned} r_{T_i}(t) &= |\hat{T}_i(t)-T_i(t)|, \quad i=1,2,3 \\ r_{C_{i2}}(t) &= |\hat{C}_{i2}(t)-C_{i2}(t)|, \quad i=A,B,C \end{aligned} \tag{14}$$

Based on these residuals, we design the EWMA residuals with $\lambda=0.5$ and the sampling time $\Delta_r=0.005$. The mean values and standard deviations of the EWMA residuals are shown in Table III.

We consider two different faults in the following simulations. First, we consider a fault in the heat input/removal actuator to vessel 2, that is a fault in Q_2. Because Q_2 only affects the state T_2 directly and all the measurements are continuously available, when there is an actuator fault in Q_2, only the residual corresponding to T_2 exceeds its threshold. The second fault we consider is a fault in the inlet flow actuator to vessel 2, that is a fault in F_{20}. Because the control action F_{20} affects directly the states T_2, C_{A2}, C_{B2} and C_{C2}, when there is an actuator fault in F_{20}, more than one of the residuals will exceed their thresholds. In the simulations, $\Delta t_d=36\,\text{s}=0.01\,\text{h}$, $w=[4\ 3\ 3;\ 2\ 2\ 2]$ and $\Delta_e=0.005\,\text{h}$.

5.2. Simulation results

Four different simulation sets are presented to demonstrate the merits of isolating by using the adaptive windows based on EWMA residuals. For each simulation, the plant is initialized at the desired steady-state x_s (see Table I) and simulated to 5.0 h with a fault being triggered at 1.050 h. Process and measurement noise is applied to the plant.

The first case considered triggers a small magnitude Q_2 fault that will demonstrate longer isolation windows to minimize false alarms along with a quicker response since fault detection/isolation begins tracking the potential fault sooner. The second and third cases will demonstrate the quick detection and isolation of a large magnitude Q_2 fault using adaptive windows in comparison to an FDI scheme using fixed isolation times. The fourth case will demonstrate an F_{20} small magnitude fault.

In the first set of simulations, a Q_2 fault with a magnitude of 15% of $u_1^{\max}$ is triggered at 1.050 h (we will refer to it as 'small' Q_2 fault). From the design of the system, the Q_2 fault directly affects the temperature in vessel 2 where we expect only the residual for T_2 to deviate. When the residual for T_2, r_{E,T_2}, (see top left plot of Figure 3) exceeds a chosen confidence level (i.e. its first threshold $\sigma_{T_2,3}$) at 1.065 h, the FDI system begins monitoring the rate of change of the T_2 residual. The residual r_{E,T_2} exceeds its second threshold $\sigma_{T_2,4}$ at 1.075 h. This fault is confirmed and declared at time 1.085 h after the waiting time $\Delta t_d=0.01\,\text{h}$. At the same time 1.075 h, a fault isolation window of $3.6\,\text{min}=0.1\,\text{h}$ is calculated based on the rate of change of r_{E,T_2} to ensure proper isolation of the fault. However, because r_{E,T_2} deviates quickly and exceeds $\sigma_{T_2,5}$ at 1.080 h, the isolation window is updated to $36\,\text{s}=0.01\,\text{h}$ and the Q_2 fault is isolated at 1.090 h. The fault is estimated at 18.5 kJ/h (actual fault value is 20 kJ/h). The FTC system reconfigures the control system, which is able to stabilize the closed-loop system near the target steady state by 1.500 h as shown in the concentration profiles in Figure 4 and the temperature profiles in Figure 5. The corresponding control actions are shown in Figure 6.

DOI: 10.1002/rnc

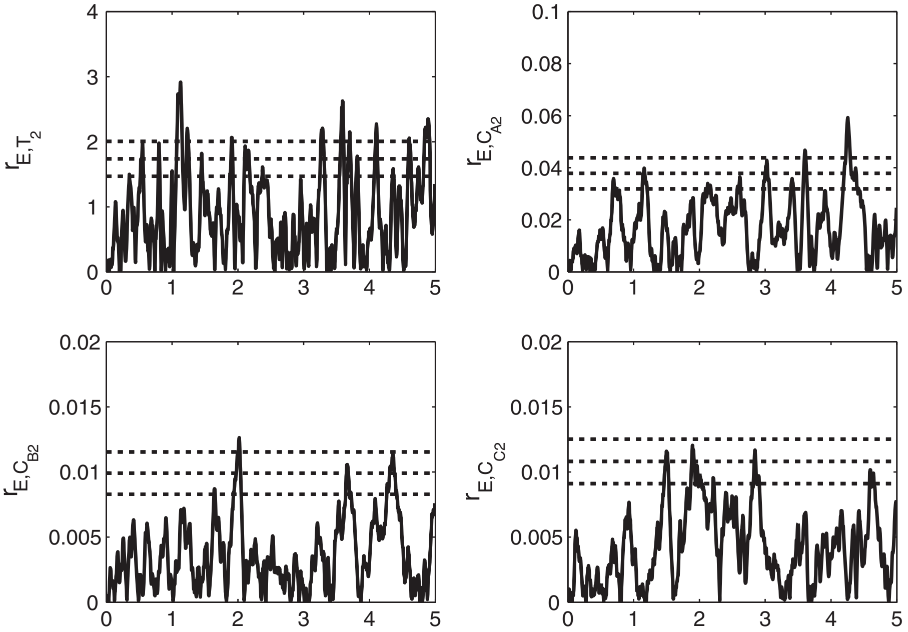

Figure 3. Case 1: Q_2 ‘small’ fault is isolated using longer waiting time calculated from the residual change of T_2. r_{E,T_2} (top left plot) exceeds $\sigma_{T_{2,3}}$ at 1.065 h and then exceeds $\sigma_{T_{2,4}}$ where the fault isolation time is set to 3.6 min. When r_{E,T_2} further exceeds $\sigma_{T_{2,5}}$ at 1.080 h, the waiting time is updated to 36 s. The fault is isolated and estimated as 18.5 kJ/h (actual 20.0 kJ/h) at 1.090 h.

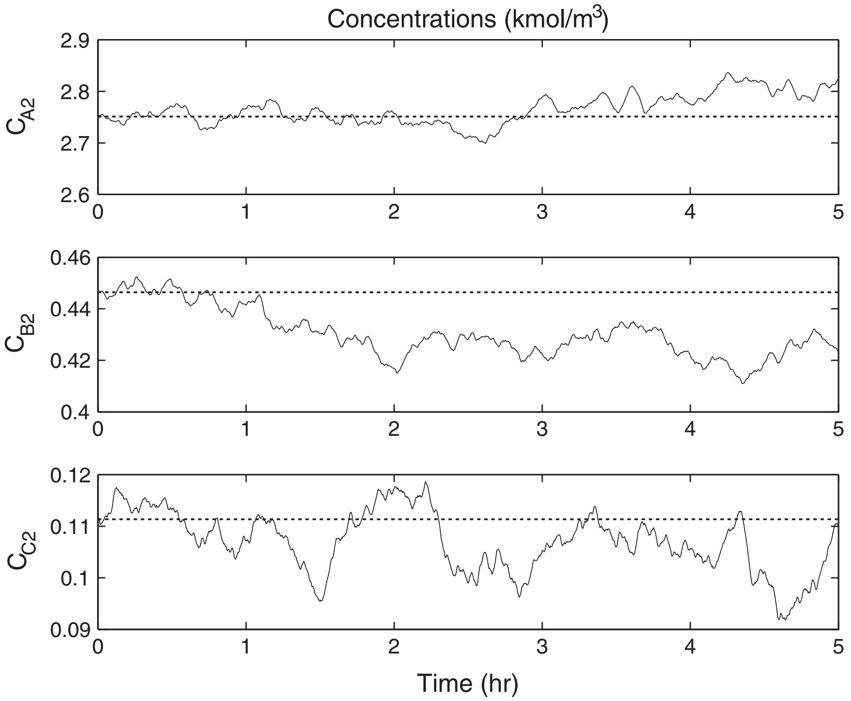

Figure 4. Case 1: Q_2 ‘small’ fault is isolated and control system is reconfigured to stabilize the closed-loop system—concentrations. Note the new steady-state values and scale.

In the second case, a Q_2 fault is set to a magnitude of 80% of $u_1^{\max}$ and is triggered at 1.050 h (we will refer to it as the ‘large’ Q_2 fault). The larger Q_2 fault will demonstrate the FDI system’s quicker response and improved robustness when used in conjunction with FTC (Figure 7).

DOI: 10.1002/rnc

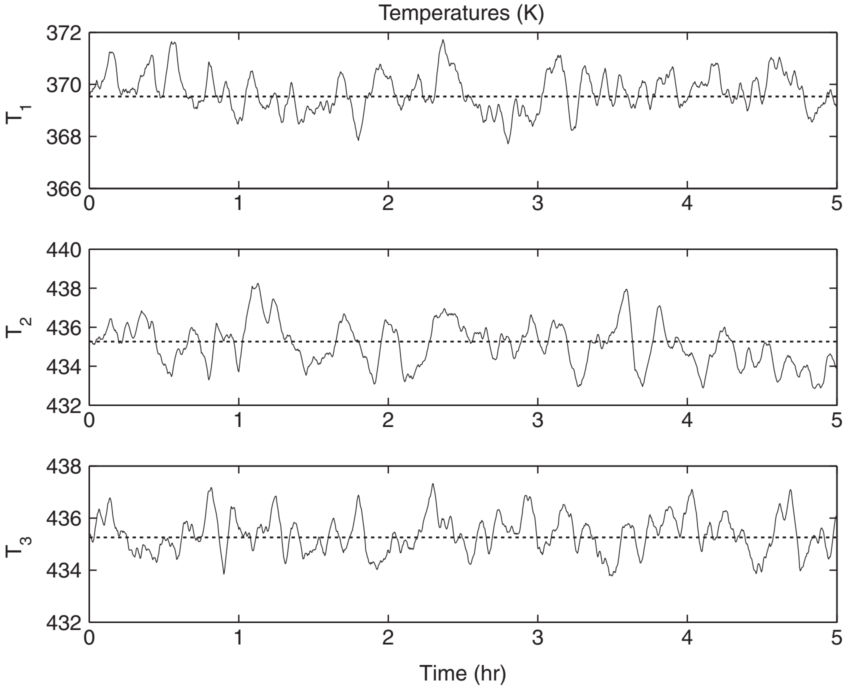

Figure 5. Case 1: Q_2 'small' fault is isolated and control system is reconfigured to stabilize the closed-loop system—temperatures.

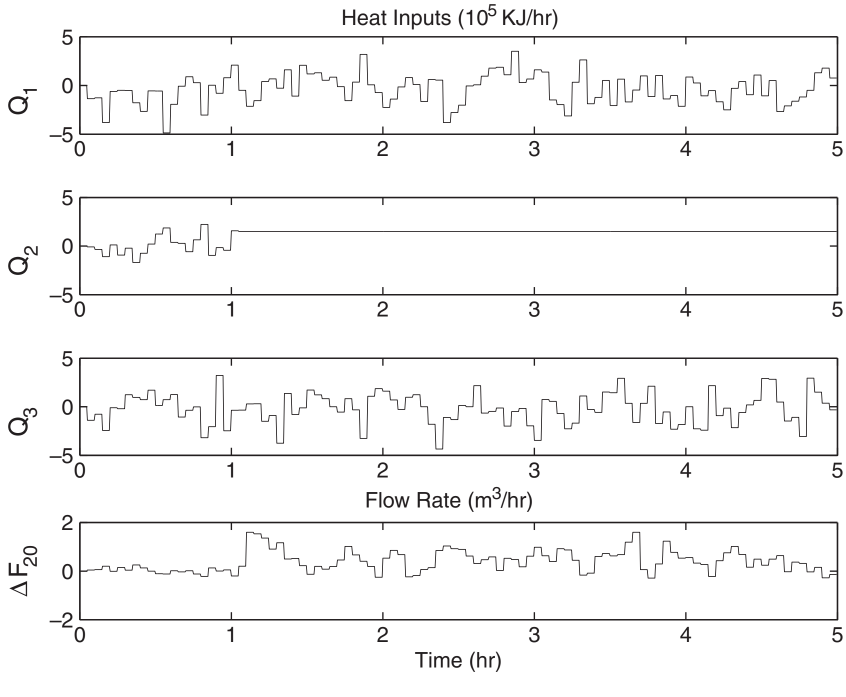

Figure 6. Case 1: Q_2 'small' fault is isolated and control system is reconfigured to stabilize the closed-loop system—control actions.

In Figure 7, the 'large' fault compared with a 'small' fault of case 1 (Figure 3) causes the residual to deviate much quicker with the FDI system beginning to monitor at 1.060 h when r_{T_2} immediately exceeds $\sigma_{T_2,5}$ and an isolation window of 36 s is calculated. The fault is declared and isolated at

DOI: 10.1002/rnc

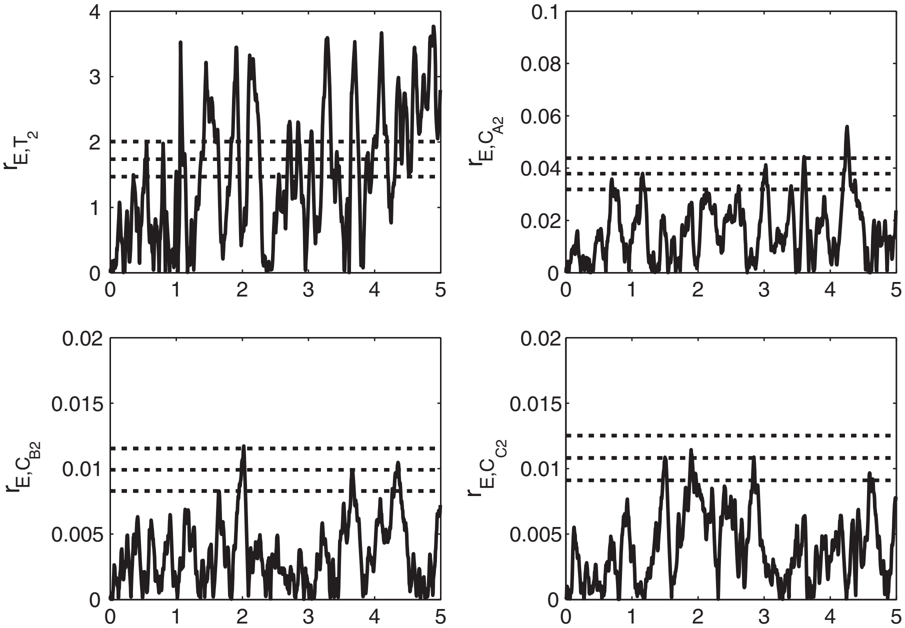

Figure 7. Case 2: Q_2 'large' fault is isolated using a shorter waiting time based on residual change of T_2. r_{E,T_2} (top left plot) immediately exceeds $\sigma_{T_{2,5}}$ in a single measurement update at 1.060 h. The calculated waiting time is 36 s. The fault is estimated as 88 kJ/h (actual 80 kJ/h) and FTC is implemented at 1.070 h.

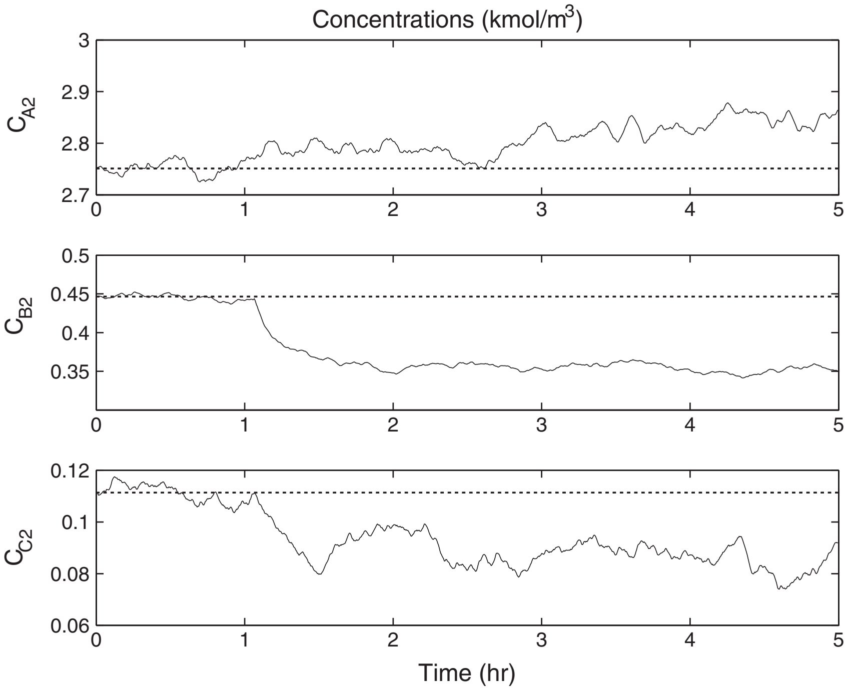

Figure 8. Case 2: Q_2 'large' fault is isolated and control system is reconfigured to stabilize the closed-loop system—concentrations. Note the new steady-state values and scale.

1.070 h soon after the r_{E,T_2} exceeded $\sigma_{T_2,5}$, and the fault is estimated as 88 kJ/h (actual 80 kJ/h). Figures 8 and 9 show that the FTC system is able to stabilize the system at a new steady-state after reconfiguration at 1.070 h. The corresponding control actions are shown in Figure 10.

DOI: 10.1002/rnc

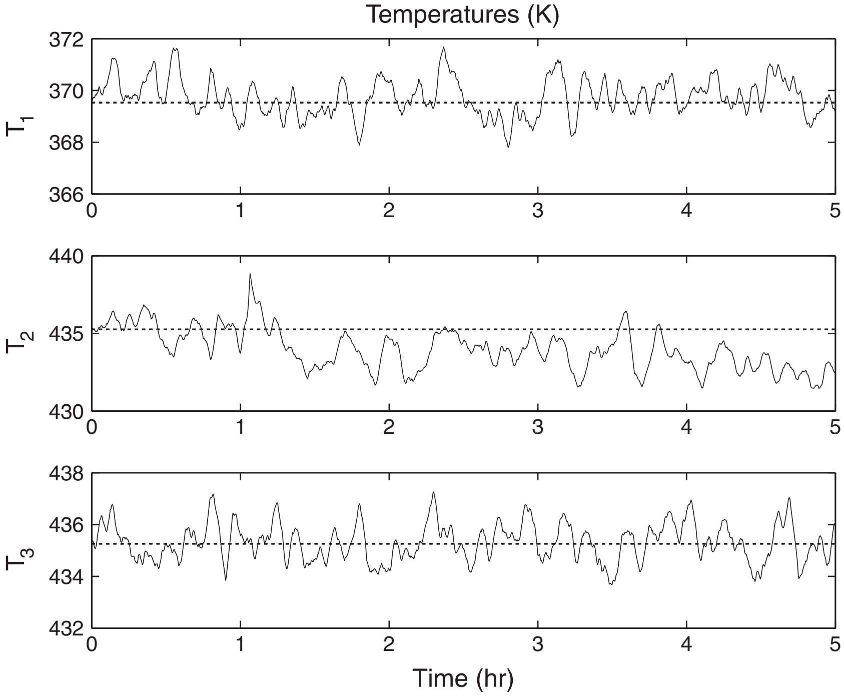

Figure 9. Q_2 'large' fault is isolated and control system is reconfigured to stabilize the closed-loop system—temperatures.

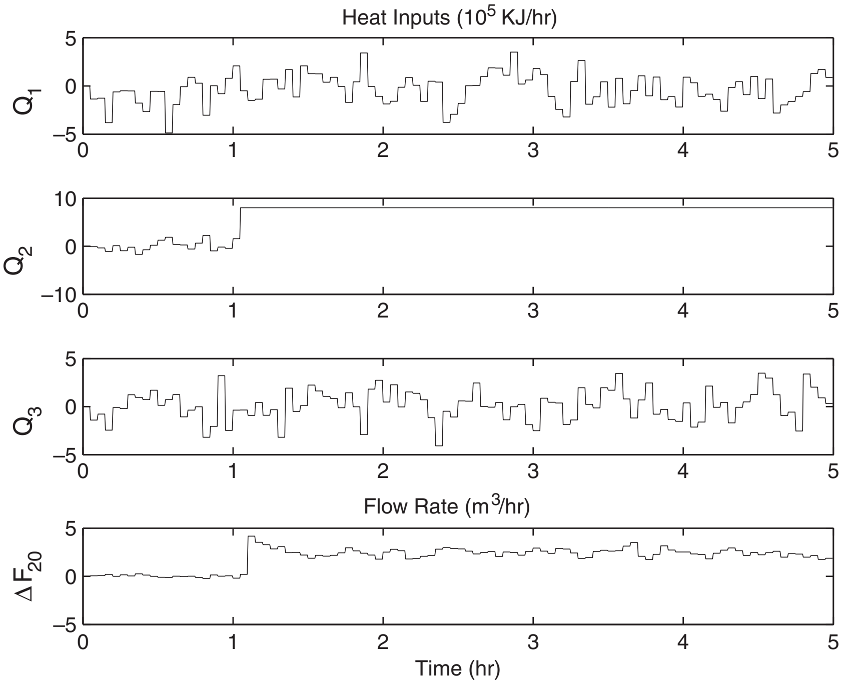

Figure 10. Q_2 'large' fault is isolated and control system is reconfigured to stabilize the closed-loop system—control actions.

The purpose of the third case is to better illustrate the need for variable windows and minimum waiting times for proper isolation. In the third case we trigger an identical Q_2 fault as in case 2 with the exception that the FDI system uses fixed isolation windows. Similarly, when r_{E,T_2} first

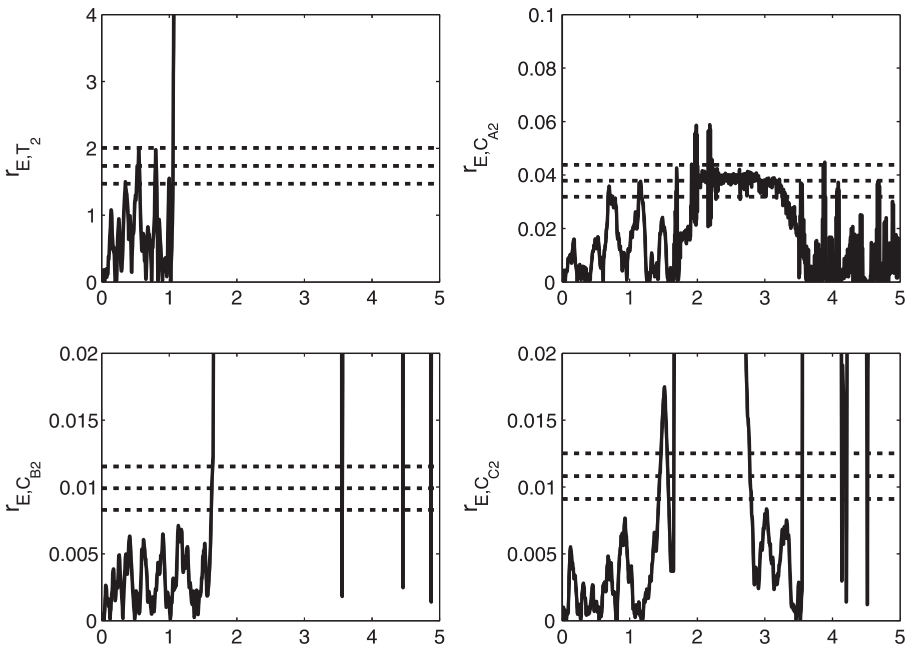

Figure 11. Case 3: Same fault conditions as in case 2, i.e. Q_2 'large' fault, using fixed isolation time. r_{E,T_2} immediately exceeding $\sigma_{T_{2,5}}$ at 1.060 h, but the isolation windows is set to worst case condition of 4.8 min. The fault is isolated at 1.135 h and estimated as 89 kJ/h (actual 80 kJ/h) but the FTC is unable to stabilize the closed-loop system.

exceeds $\sigma_{T_2,5}$ at 1.060 h, the isolation system monitors the remaining residuals for a matching fault signature over a fixed window of 4.8 min (Figure 11). At the end of the fixed window at 1.135 h, the FTC system (identical to that of case 2) reconfigures the DMPC system with a Q_2 fault estimate of 89 kJ/h (actual 80 kJ/h) and is unable to stabilize the system due to the plant state having left the stability region of the reconfigured control system (Figures 12 and 13). Note that after fault isolation the residuals are no longer used.

The fourth case involves an F_{20} fault whose fault signature includes r_{E,T_2} and at least one other concentration residual ($r_{E,CA_2}, r_{E,CB_2}, r_{E,CC_2}$). In this case, an F_{20} fault occurs with a magnitude of 17% of $u_2^{\max}$. In Figure 14, the T_2 residual exceeds $\sigma_{T_2,3}$ at 1.100 h while the residual for concentration of component B in the second tank exceeds $\sigma_{C_{B2},4}$ at 1.105 h. A fault is declared at 1.115 h when r_{E,CB_2} stays above $\sigma_{C_{B2},4}$ for 0.01 h. A fault isolation window of 4.8 min is calculated at 1.105 h. However, within the isolation window, r_{E,CB_2} exceeds $\sigma_{C_{B2},5}$ at 1.120 h and a new isolation window of 36 s is calculated. At the end of the new isolation window (i.e. $t = 1.130$h), no matching fault signature is found and the FDI system continues monitoring the residuals until 1.150 h when a matching fault signature is found when r_{E,T_2} exceeds $\sigma_{T_2,5}$ at 1.140 h and stays above it for 0.01 h. The FTC is implemented once the fault is isolated with a fault estimate 1.08 m^3/h (actual 0.85 m^3/h) (Figures 15–17).

6. CONCLUSIONS

In this work, we developed a monitoring and reconfiguration system for a DMPC system in the presence of control actuator faults taking advantage of both process models and process measurements. Specifically, we first designed fault detection filters and corresponding filter residuals, which are computed via the EWMA method, to effectively detect actuator faults. Then, we proposed a fault isolation approach which uses adaptive fault isolation time windows to quickly and accurately isolate actuator faults and reduce the probability of false alarms. Subsequently, we designed appropriate FTC strategies to handle the actuator faults by reconfiguring the DMPC system and maintain

DOI: 10.1002/rnc

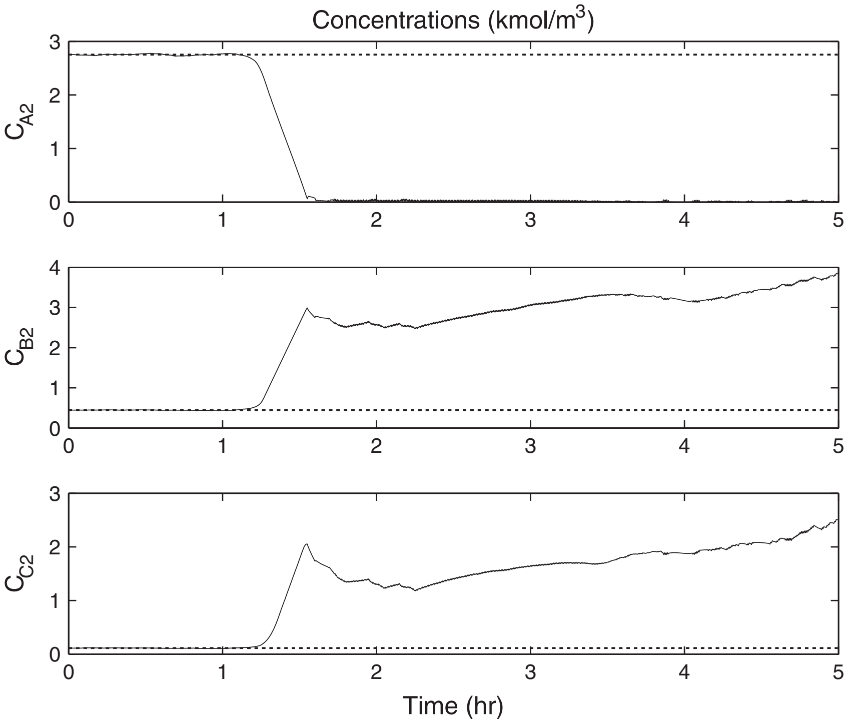

Figure 12. Case 3: Same fault conditions as in case 2, i.e. Q_2 'large' fault, using fixed isolation time. The control system is reconfigured at 1.135 h and is unable to stabilize the closed-loop system—concentrations.

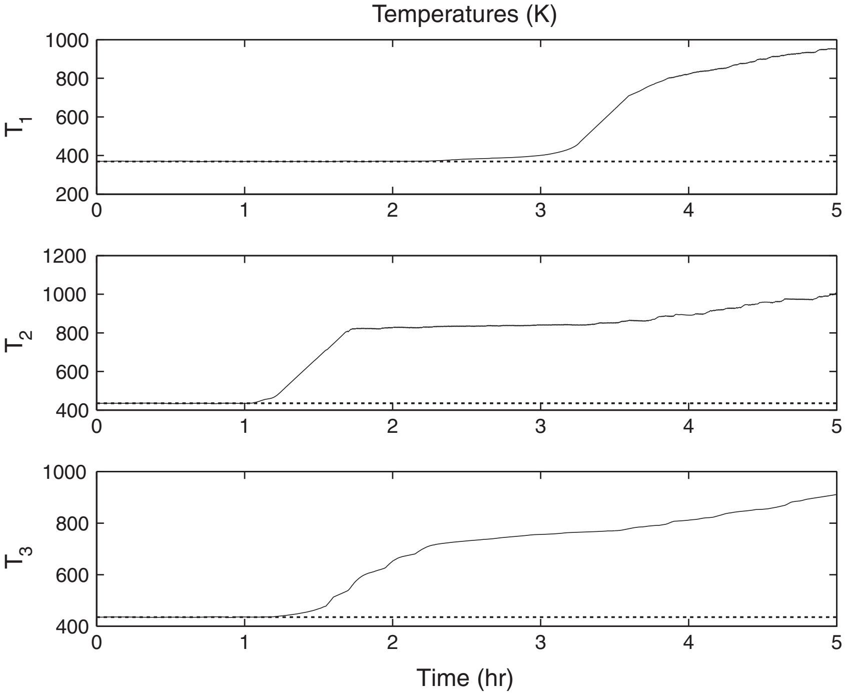

Figure 13. Case 3: Same fault conditions as in case 2, i.e. Q_2 'large' fault, using a fixed isolation time. The control system is reconfigured at 1.135 h and is unable to stabilize the closed-loop system—temperatures.

DOI: 10.1002/rnc

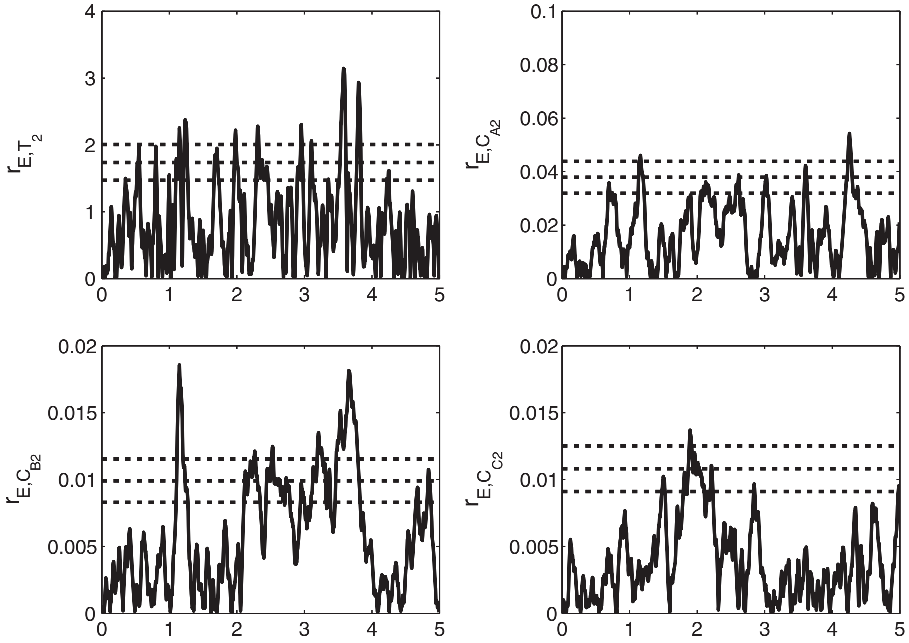

Figure 14. Case 4: F_{20} fault demonstrates FDI system with multiple residuals exceeding the thresholds under small magnitude fault. The residual r_{T_2} first exceeds $\sigma_{T_2,3}$ at 1.100 h and $r_{C_{B2}}$ exceeds $\sigma_{C_{B2},4}$ at 1.105 h. A fault is declared at 1.115 h and an isolation time window of 4.8 min is calculated. A new fault isolation window of 36 s is calculated when $r_{C_{B2}}$ exceeds $\sigma_{C_{B2},5}$ at 1.120 h and at the end of the new isolation window, no matching fault signature is found. The FDI system continues monitoring the residuals until 1.150 h when a matching fault signature is found. The FTC is implemented once the fault is isolated with a fault estimate 1.08 m^3/h (actual 0.85 m^3/h).

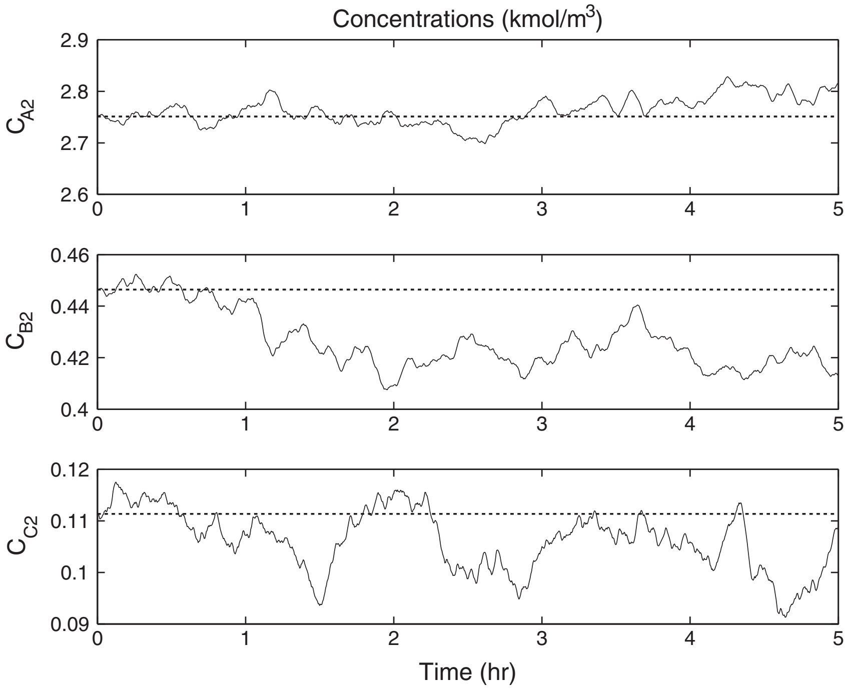

Figure 15. Case 4: F_{20} fault is isolated and control system is reconfigured to stabilize the closed-loop system—concentrations.

DOI: 10.1002/rnc

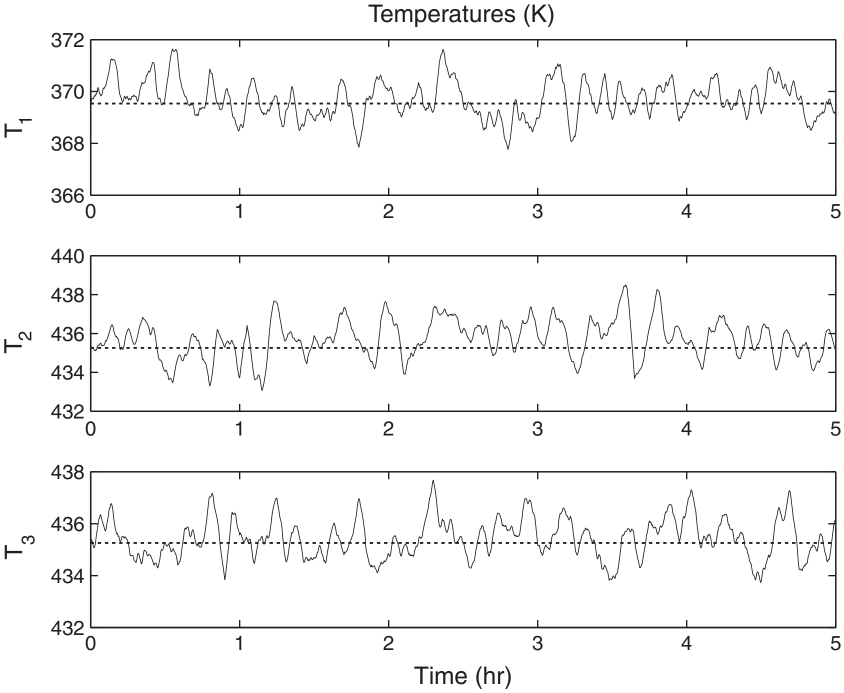

Figure 16. Case 4: F_{20} fault is isolated and control system is reconfigured to stabilize the closed-loop system—temperatures.

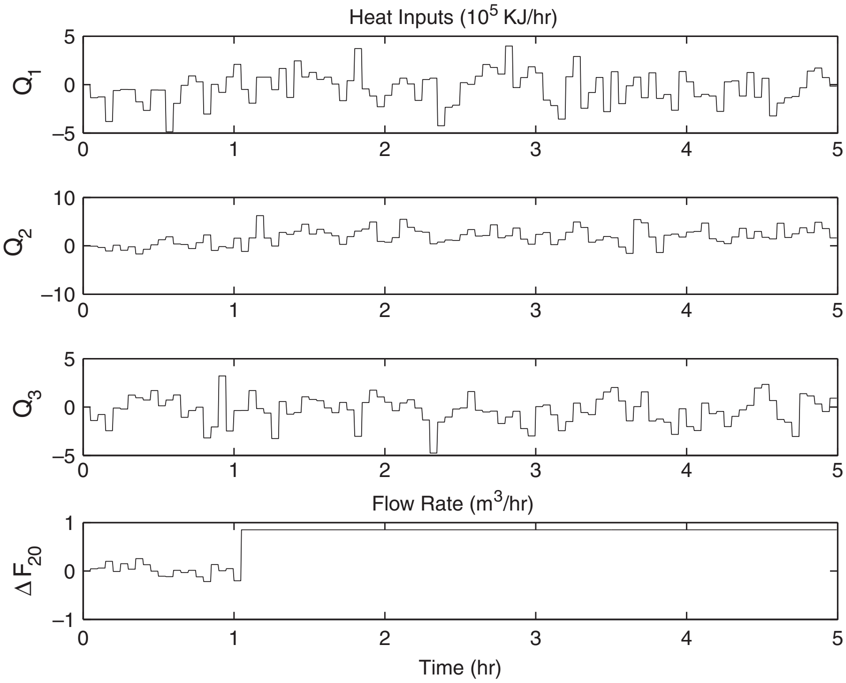

Figure 17. Case 4: F_{20} fault is isolated and control system is reconfigured to stabilize the closed-loop system—control actions.

the closed-loop system state within a desired operating region. The applicability and effectiveness of the proposed approach were illustrated via extensive simulations based on a nonlinear reactor–separator process example.

DOI: 10.1002/rnc

REFERENCES

1. Ydstie EB. New vistas for process control: integrating physics and communication networks. *AIChE Journal* 2002; **48**:422–426.
2. Christofides PD, Davis JF, El-Farra NH, Clark D, Harris KRD, Gipson JN. Smart plant operations: vision, progress and challenges. *AIChE Journal* 2007; **53**:2734–2741.
3. Nimmo I. Adequately address abnormal operations. *Chemical Engineering Progress* 1995; **91**:36–45.
4. Mhaskar P, Gani A, El-Farra NH, Christofides PD, Davis JF. Integrated fault-detection and fault-tolerant control of process systems. *AIChE Journal* 2006; **52**:2129–2148.
5. El-Farra NH, Ghantasala S. Actuator fault isolation and reconfiguration in transport-reaction processes. *AIChE Journal* 2007; **53**:1518–1537.
6. Fukunaga K. *Introduction to Statistical Pattern Recognition*. Academic Press: New York, 1990.
7. Mhaskar P, McFall C, Gani A, Christofides PD, Davis JF. Isolation and handling of actuator faults in nonlinear systems. *Automatica* 2008; **44**:53–62.
8. Romagnoli J, Palazoglu A. *Introduction to Process Control*. CRC Press: Boca Raton, 2006.
9. Venkatasubramanian V, Rengaswamy R, Yin K, Kavuri S. A review of process fault detection and diagnosis part I: quantitative model-based methods. *Computers and Chemical Engineering* 2003; **27**:293–311.
10. Venkatasubramanian V, Rengaswamy R, Kavuri S, Yin K. A review of process fault detection and diagnosis part III: process history based methods. *Computers and Chemical Engineering* 2003; **27**:327–346.
11. Rawlings JB, Stewart BT. Coordinating multiple optimization-based controllers: new opportunities and challenges. *Journal of Process Control* 2008; **18**:839–845.
12. Scattolini R. Architectures for distributed and hierarchical model predictive control—a review. *Journal of Process Control* 2009; **19**:723–731.
13. Liu J, Muñoz de la Peña D, Christofides PD. Distributed model predictive control of nonlinear process systems. *AIChE Journal* 2009; **55**:1171–1184.
14. Liu J, Chen X, Muñoz de la Peña D, Christofides PD. Sequential and iterative architectures for distributed model predictive control of nonlinear process systems. *AIChE Journal* 2010; **56**:2137–2149.
15. Camponogara E, Jia D, Krogh BH, Talukdar S. Distributed model predictive control. *IEEE Control Systems Magazine* 2002; **22**:44–52.
16. Maestre JM, Muñoz de la Peña D, Camacho EF. A distributed MPC scheme with low communication requirements. *Proceedings of the American Control Conference*, Saint Louis, MO, U.S.A., 2009; 2797–2802.
17. Liu J, Muñoz de la Peña D, Christofides PD. Distributed model predictive control of nonlinear systems subject to asynchronous and delayed measurements. *Automatica* 2010; **46**:52–61.
18. Chilin D, Liu J, Muñoz de la Peña D, Christofides PD, Davis JF. Detection, isolation and handling of actuator faults in distributed model predictive control systems. *Journal of Process Control* 2010; **20**:1059–1075.
19. Lin Y, Sontag ED, Wang Y. A smooth converse Lyapunov theorem for robust stability. *SIAM Journal on Control and Optimization* 1996; **34**:124–160.
20. Christofides PD, El-Farra NH. *Control of Nonlinear and Hybrid Process Systems*: *Designs for Uncertainty*, *Constraints and Time-delays*. Springer: Berlin, Germany, 2005.
21. Ohran B, Muñoz de la Peña D, Christofides PD, Davis JF. Enhancing data-based fault isolation through nonlinear control. *AIChE Journal* 2008; **53**:223–241.
22. Lucas JM, Saccucci MS. Exponentially weighted moving average control schemes: properties and enhancements. *Technometrics* 1990; **32**:1–12.

DOI: 10.1002/rnc