Fault-Tolerant Control of Process Systems Using Communication Networks

Nael H. El-Farra, Adiwinata Gani, and Panagiotis D. Christofides
Dept. of Chemical Engineering, University of California, Los Angeles, CA 90095

DOI 10.1002/aic.10443
Published online April 21, 2005 in Wiley InterScience (www.interscience.wiley.com).

A methodology for the design of fault-tolerant control systems for chemical plants with distributed interconnected processing units is presented. Bringing together tools from Lyapunov-based nonlinear control and hybrid systems theory, the approach is based on a hierarchical architecture that integrates lower-level feedback control of the individual units with upper-level logic-based supervisory control over communication networks. The local control system for each unit consists of a family of control configurations for each of which a stabilizing feedback controller is designed and the stability region is explicitly characterized. The actuators and sensors of each configuration are connected, via a local communication network, to a local supervisor that orchestrates switching between the constituent configurations, on the basis of the stability regions, in the event of failures. The local supervisors communicate, through a plant-wide communication network, with a plant supervisor responsible for monitoring the different units and coordinating their responses in a way that minimizes the propagation of failure effects. The communication logic is designed to ensure efficient transmission of information between units, while also respecting the inherent limitations in network resources by minimizing unnecessary network usage and accounting explicitly for the effects of possible delays due to fault-detection, control computations, network communication and actuator activation. The proposed approach provides explicit guidelines for managing the various interplays between the coupled tasks of feedback control, fault-tolerance and communication. The efficacy of the proposed approach is demonstrated through chemical process examples. © 2005 American Institute of Chemical Engineers *AIChE J,* 51: 1665–1682, 2005

Keywords: hybrid systems and control, switching logic, stability regions, fault-tolerance, supervisory control, communication networks, process systems

Introduction

Safety and reliability are primary goals in the operation of industrial chemical plants. An important national need currently exists for enhancing the safety and reliability of chemical plants in ways that reduce their vulnerability to serious failures. Increasingly faced with the requirements of operational flexibility under tight performance specifications and other economic drivers, plant operation is relying extensively on highly automated process control systems. Automation, however, tends to increase vulnerability of the plant to faults, such as defects/malfunctions in process equipment, sensors and actuators, failures in the controllers or in the control loops, which, if not appropriately handled in the control system design, can potentially cause a host of undesired economic, environmental, and safety problems that seriously degrade the operating efficiency of the plant. These considerations provide a strong motivation for the development of systematic methods and strategies for the design of fault-tolerant control systems and have motivated many research studies in this area (see, for example, [1,2,3] and [4,5,6] for references).

Given the complex dynamics of chemical processes (due, for

Correspondence concerning this article should be addressed to P. D. Christofides at pdc@seas.ucla.edu example, to the presence of nonlinearities and constraints) and the geographically distributed, interconnected nature of plant units, as well as the large number of distributed sensors and actuators typically involved, the success of any fault-tolerant control strategy requires an integrated approach that brings together several essential elements, including: (1) the design of advanced feedback control algorithms that handle complex dynamics effectively, (2) the design of supervisory switching schemes that orchestrate the transition from the failed control configuration to available well-functioning fallback configurations to ensure fault-tolerance, and (3) the efficient exchange of information and communication between the different plant units through a high-level supervisor that coordinates the overall plant response in failure situations and minimizes the effects of failure propagation.

The realization of such an approach is increasingly aided by a confluence of recent, and ongoing, advances in several areas of process control research, including advances in nonlinear controller designs for chemical processes (for example, [7,8,9,10,11]) and advances in the analysis and control of hybrid process systems leading to the development of a systematic framework for the integration of feedback and supervisory control.[12,13] A hybrid systems framework provides a natural setting for the analysis and design of fault-tolerant control systems since the occurrence of failure and subsequent switching to fallback control configurations induce discrete transitions superimposed on the underlying continuous dynamics. Hybrid control techniques have been useful in dealing with a wide range of problems that cannot be addressed using classical control approaches, including fault-tolerant control of spatially-distributed systems (for example, [14,15]), control of processes with switched dynamics (for example, [13,16]), and the design of hybrid predictive control structures that overcome some of the limitations of classical predictive control algorithms (for example, [17]). In addition to control studies, research work on hybrid systems spans a diverse set of problems ranging from the modeling (for example, [18,19]) and simulation (for example, [19,20]) to the optimization (for example, [21,22]) and stability analysis (for example, [23,24]) of several classes of hybrid systems.

In addition to the above fundamental advances, recent innovations in actuator/sensor and communication technologies are increasingly enabling the integration of communication and control domains.[25] For example, the use of communication networks as media to interconnect the different components in an industrial control system is rapidly increasing and expected to replace the more costly point-to-point connection schemes currently employed in distributed control systems. Figure 1 shows the basic networked control architecture for (a) a single-unit plant with few actuators and sensors (centralized structure), and (b) a larger plant with several interconnected processing units and larger number of actuators and sensors (distributed hierarchical structure).

Currently, networked control systems is an active area of research within control engineering (for example, see [26,27,28,29,30] for some recent results and references in this area). In addition to the advantages of reduced system wiring (reduced installation, maintenance time and costs) in this architecture, the increased flexibility and ease of maintenance of a system using a network to transfer information is an appealing goal. In the context of fault-tolerant control in particular, systems designed in this manner allow for easy modification of the control strategy by rerouting signals, having redundant systems that can be activated automatically when component failure occurs, and in general they allow having a high-level supervisory control over the entire plant. The appealing features of communication networks motivate investigating ways for integrating them in the design of fault-tolerant control systems to ensure a timely and coordinated response of the plant in ways that minimize the effects of failure propagation between plant units. This entails devising strategies to deal with some of the fundamental issues introduced by the network, including issues of bandwidth limitations, quantization effects, network scheduling, and communication delays, which continue to be topics of active research.

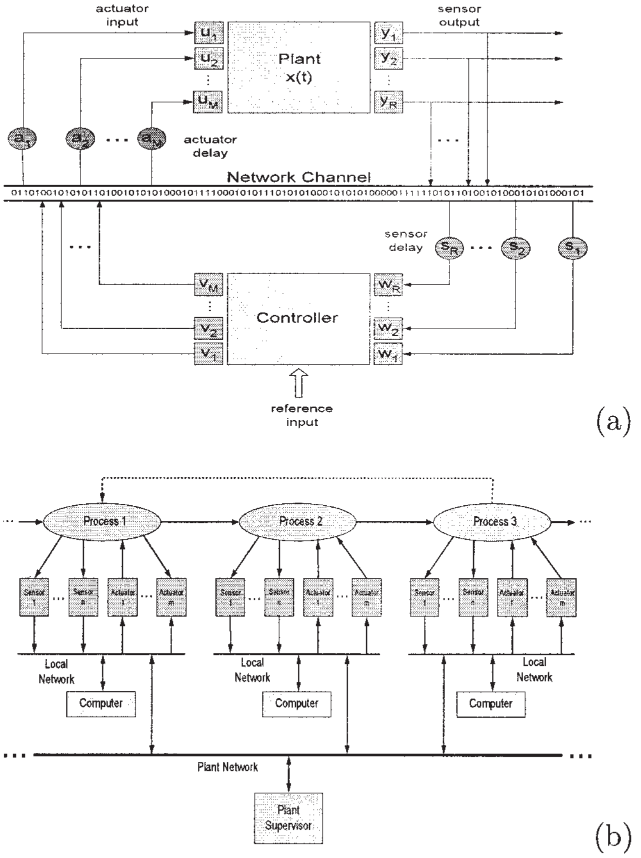

Figure 1. (a) A centralized networked control system for a single-unit plant, and (b) a hierarchical distributed networked control architecture for a multi-unit plant.

Motivated by the earlier considerations, we develop in this work a fault-tolerant control system design methodology, for plants with multiple (distributed) interconnected processing units, that accounts explicitly for the inherent complexities in supervisory control and communication tasks resulting from the distributed interconnected nature of plant units. The approach brings together tools from Lyapunov-based control and hybrid systems theory, and is based on a hierarchical distributed architecture that integrates lower-level feedback control of the individual units with upper-level logic-based supervisory control over communication networks. The local control systems consist each of a family of feedback control configurations together with a local supervisor that communicates with actuators and sensors, via a local communication network, to orchestrate the transition between control configurations, on the basis of their fault-recovery regions, in the event of failures. The local supervisors communicate, through a plant-wide communication network, with a plant supervisor responsible for monitoring the different units and coordinating their responses in a way that minimizes the propagation of failure effects. The communication logic is designed to ensure efficient transmission of information between units while also respecting the inherent limitations in network resources by minimizing unnecessary network usage and accounting explicitly for the effects of possible delays due to fault-detection, control computations, network communication, and actuator activation. The proposed approach provides explicit guidelines for managing the interplays between the coupled tasks of feedback control, fault-tolerance and communication. The efficacy of the proposed approach is demonstrated through chemical process examples.

Preliminaries

System description

We consider a plant composed of l connected processing units, each of which is modeled by a continuous-time multivariable nonlinear system with constraints on the manipulated inputs, and represented by the following state space description

$$\begin{aligned}
\dot{x}_1 &= f_1^{k_1}(x_1) + G_1^{k_1}(x_1)u_1^{k_1} \\
\dot{x}_2 &= f_2^{k_2}(x_2) + G_2^{k_2}(x_2)u_2^{k_2} + W_{2,1}^{k_2}(x_2)x_1 \\
&\vdots \\
\dot{x}_l &= f_l^{k_l}(x_l) + G_l^{k_l}(x_l)u_l^{k_l} + \sum_{p=1}^{l-1} W_{l,p}^{k_l}(x_l)x_p \\
\|u_i^{k_i}\| &\le u_{i,max}^{k_i} \\
k_i(t) &\in \mathcal{K}_i := \{1, \ldots, N_i\}, \quad N_i < \infty, \quad i = 1, \ldots, l
\end{aligned} \tag{1}$$

where $x_i := [x_i^{(1)} \quad x_i^{(2)} \quad \cdots \quad x_i^{(n_i)}]^T \in \mathbb{R}^{n_i}$ denotes the vector of process state variables associated with the i-th processing unit, $u_i^{k_i} := [u_{i,1}^{k_i} \quad u_{i,2}^{k_i} \quad \cdots \quad u_{i,m_i}^{k_i}]^T \in \mathbb{R}^{m_i}$ denotes the vector of constrained manipulated inputs associated with the k_i-th control configuration in the i-th processing unit, $u_{i,max}^{k_i}$ is a positive real number that captures the maximum size of the vector of manipulated inputs dictated by the constraints, $\|\cdot\|$ denotes the Euclidean norm of a vector, and N_i is the number of different control configurations that can be used to control the i-th processing unit. The index, $k_i(t)$, which takes values in the finite set $\mathcal{K}_i$, represents a discrete state that indexes the righthand side of the set of differential equations in Eq. 1. For each value that k_i assumes in $\mathcal{K}_i$, the i-th processing unit is controlled via a different set of manipulated inputs which define a given control configuration. For each unit, switching between the available N_i control configurations is controlled by a local supervisor that monitors the operation of the unit and orchestrates, accordingly, the transition between the different control configurations in the event of control system failures. This in turn determines the temporal evolution of the discrete state, $k_i(t)$, which takes the form of a piecewise constant function of time. The local supervisor ensures that only one control configuration is active at any given time, and allows only a finite number of switches over any finite interval of time.

Without loss of generality, it is assumed that $x_i = 0$ is an equilibrium point of the uncontrolled i-th processing unit (that is, with $u_i^{k_i} = 0$), and that the vector functions, $f_i^{k_i}(\cdot)$, and the matrix functions, $G_i^{k_i}(\cdot)$ and $W_{j,p}^{k_j}(\cdot)$, are sufficiently smooth on their domains of definition, for all $k_i \in \mathcal{K}_i$, $i = 1, \ldots, l$, $j = 2, \ldots, l$, $p = 1, \ldots, l - 1$. For the j-th processing unit, the term, $W_{j,p}^{k_j}(x_j)x_p$, represents the connection that this unit has with the p-th unit upstream. Note from the summation notation in Eq. 1 that each processing unit can in general be connected to all the units upstream from it. Our nominal control objective (that is, in the absence of control system failures) is to design, for each processing unit, a stabilizing feedback controller that enforces asymptotic stability of the origin of the closed-loop system in the presence of control actuator constraints. Moreover, the assumption that the state x_i enters the x_{i+1}-subsystem in a linear fashion is made for notational simplicity and can be relaxed. To simplify the presentation of our results, we will focus only on the state feedback control problem where measurements of all process states are available for all times.

Problem statement and solution overview

Consider the plant of Eq. 1 where, for each processing unit, a stabilizing feedback control system has been designed and implemented. Given some catastrophic fault—that has been detected and isolated—in the actuators of one of the control systems, our objective is to develop a plant-wide fault-tolerant control strategy that: (1) preserves closed-loop stability of the failing unit, if possible, and (2) minimizes the negative impact of this failure on the closed-loop stability of the remaining processing units downstream. To accomplish both of these objectives, we construct a hierarchical control structure that integrates lower-level feedback control of the individual units with upper-level logic-based supervisory control over communication networks. The local control system for each unit consists of a family of control configurations for each of which a stabilizing feedback controller is designed, and the stability region is explicitly characterized. The actuators and sensors of each configuration are connected, via a local communication network, to a local supervisor that orchestrates switching between the constituent configurations, on the basis of the stability regions, in the event of failures. The local supervisors communicate, through a plant-wide communication network, with a plant supervisor responsible for monitoring the different units and coordinating their responses in a way that minimizes the propagation of failure effects. The basic problem under investigation is how to coordinate the tasks of feedback, control system reconfiguration and communication, both at the local (processing unit) and plant-wide levels in a way that ensures timely recovery in the event of failure and preserves closed-loop stability.

Remark 1: In the design of any fault-tolerant control system, an important task that precedes the control system reconfiguration is the task of fault-detection and isolation (FDI). There is an extensive body of literature on this topic including,

Table 1. Process Parameters and Steady-State Values for the Chemical Reactor of Eq. 13

$F = 4.998$	m^3/hr
$V = 1.0$	m^3
$R = 8.314$	$KJ/kmol \cdot K$
$T_{A0} = 300.0$	K
$C_{A0} = 4.0$	$kmol/m^3$
$C_{B0} = 0.0$	$kmol/m^3$
$\Delta H_1 = -5.0 \times 10^4$	KJ/kmol
$\Delta H_2 = -5.2 \times 10^4$	KJ/kmol
$\Delta H_3 = -5.4 \times 10^4$	KJ/kmol
$k_{10} = 3.0 \times 10^6$	hr^{-1}
$k_{20} = 3.0 \times 10^5$	hr^{-1}
$k_{30} = 3.0 \times 10^5$	hr^{-1}
$E_1 = 5.0 \times 10^4$	KJ/kmol
$E_2 = 7.53 \times 10^4$	KJ/kmol
$E_3 = 7.53 \times 10^4$	KJ/kmol
$\rho = 1000.0$	kg/m^3
$c_p = 0.231$	$KJ/kg \cdot K$
$T^s = 388.57$	K
$C_A^s = 3.59$	$kmol/m^3$
$C_B^s = 0.41$	$kmol/m^3$

for example, the design of fault-detection and isolation schemes, based on fundamental process models (for example, [31,32]) and statistical/pattern recognition and fault diagnosis techniques (for example, [33,34,35,36,37,38]). In this work, we focus mainly on the interplay between the communication network and the control system reconfiguration task. To this end, we assume that the FDI tasks take place at a time scale that is very fast compared to the time constant of the overall process dynamics and the time needed for the control system reconfiguration, and, thus, can be treated separately from the control system reconfiguration (we note that the time needed for FDI is accounted for in the control system reconfiguration through a time-delay; see the next section and the simulation studies for details). In the context of process control applications, this sequential and decoupled treatment of FDI and control system reconfiguration is further justified by the overall slow dynamics of chemical plants.

Motivating example

In this section, we introduce a simple benchmark example that will be revisited later to illustrate the design and implementation aspects of the fault-tolerant control design methodology to be proposed in the next section. While the discussion will center around this example, we note that the proposed framework can be applied to more complex plants involving more complex arrangements of processing units as shown in Eq. 1. To this end, consider two well-mixed, nonisothermal continuous stirred-tank reactors (CSTRs) in series, where three parallel irreversible elementary exothermic reactions of the form $A \xrightarrow{k_1} B$, $A \xrightarrow{k_2} U$ and $A \xrightarrow{k_3} R$ take place, where A is the reactant species, B is the desired product and U, R are undesired byproducts. The feed to CSTR 1 consists of pure A at flow rate F_0, molar concentration C_{A0}, and temperature T_0, and the feed to CSTR 2 consists of the output of CSTR 1, and an additional fresh stream feeding pure A at flow rate F_3, molar concentration C_{A03}, and temperature T_{03}. Due to the nonisothermal nature of the reactions, a jacket is used to remove/provide heat to both reactors. Under standard modeling assumptions, a mathematical model of the plant can be derived from material and energy balances, and takes the following form

$$\frac{dT_1}{dt} = \frac{F_0}{V_1}(T_0 - T_1) + \sum_{i=1}^{3} \frac{(-\Delta H_i)}{\rho c_p} R_i(C_{A1}, T_1) + \frac{Q_1}{\rho c_p V_1}$$

$$\frac{dC_{A1}}{dt} = \frac{F_0}{V_1}(C_{A0} - C_{A1}) - \sum_{i=1}^{3} R_i(C_{A1}, T_1)$$

$$\frac{dT_2}{dt} = \frac{F_1}{V_2}(T_1 - T_2) + \frac{F_3}{V_2}(T_{03} - T_2) + \sum_{i=1}^{3} \frac{(-\Delta H_i)}{\rho c_p} R_i(C_{A2}, T_2) + \frac{Q_2}{\rho c_p V_2}$$

$$\frac{dC_{A2}}{dt} = \frac{F_1}{V_2}(C_{A1} - C_{A2}) + \frac{F_3}{V_2}(C_{A03} - C_{A2}) - \sum_{i=1}^{3} R_i(C_{A2}, T_2) \tag{2}$$

where $R_i(C_{Aj}, T_j) = k_{i0}\exp(-E_i/RT_j)C_{Aj}$, for $j = 1, 2$. T, C_A, Q, and V denote the temperature of the reactor, the concentration of species A, the rate of heat input/removal from the reactor, and the volume of reactor, respectively, with subscript 1 denoting CSTR 1, and subscript 2 denoting CSTR 2. ΔH_i, k_i, E_i, $i = 1, 2, 3$, denote the enthalpies, pre-exponential constants and activation energies of the three reactions, respectively, c_p and ρ denote the heat capacity and density of fluid in the reactor. Using typical values for the process parameters (see Table 2), CSTR 1, with $Q_1 = 0$, has three steady-states: two locally asymptotically stable and one unstable at $(T_1^s, C_{A1}^s) = (388.57$ K, 3.59 kmol/m$^3)$. The unstable steady-state of CSTR 1 corresponds to three steady-states for CSTR 2 (with $Q_2 = 0$), one of which is unstable at $(T_2^s, C_{A2}^s) = (429.24$ K, 2.55 kmol/m$^3)$.

The control objective is to stabilize both reactors at the (open-loop) unstable steady-states. Operation at these points is typically sought to avoid high temperatures, while simulta-

Table 2. Process Parameters and Steady-State Values for the Chemical Reactors of Eq. 2

$F_0 = 4.998$	m^3/hr
$F_1 = 4.998$	m^3/hr
$F_3 = 30.0$	m^3/hr
$V_1 = 1.0$	m^3
$V_2 = 3.0$	m^3
$R = 8.314$	$KJ/kmol \cdot K$
$T_0 = 300.0$	K
$T_{03} = 300.0$	K
$C_{A0} = 4.0$	$kmol/m^3$
$C_{A03}^s = 2.0$	$kmol/m^3$
$\Delta H_1 = -5.0 \times 10^4$	KJ/kmol
$\Delta H_2 = -5.2 \times 10^4$	KJ/kmol
$\Delta H_3 = -5.4 \times 10^4$	KJ/kmol
$k_{10} = 3.0 \times 10^6$	hr^{-1}
$k_{20} = 3.0 \times 10^5$	hr^{-1}
$k_{30} = 3.0 \times 10^5$	hr^{-1}
$E_1 = 5.0 \times 10^4$	KJ/kmol
$E_2 = 7.53 \times 10^4$	KJ/kmol
$E_3 = 7.53 \times 10^4$	KJ/kmol
$\rho = 1000.0$	kg/m^3
$c_p = 0.231$	$KJ/kg \cdot K$
$T_1^s = 388.57$	K
$C_{A1}^s = 3.59$	$kmol/m^3$
$T_2^s = 429.24$	K
$C_{A2}^s = 2.55$	$kmol/m^3$

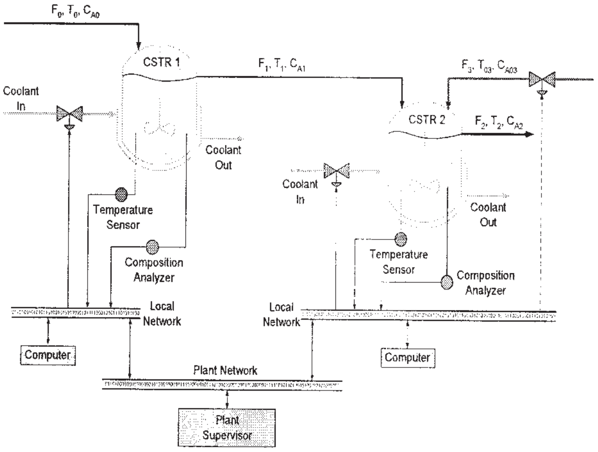

Figure 2. CSTR units in series.

neously achieving reasonable conversion. To accomplish the control objective under normal conditions (with no failures), we choose the rates of heat input, $u_1^1 = Q_1$ and $u_1^2 = Q_2$, as manipulated inputs, subject to the constraints $|Q_1| \leq u_{max}^{Q_1} = 2.7 \times 10^6$ KJ/hr and $|Q_2| \leq u_{max}^{Q_2} = 2.8 \times 10^6$ KJ/hr.

As shown in Figure 2, each unit has a local control system with its sensors and actuators connected through a communication network. The local control systems in turn communicate with the plant supervisor (and with each other) through a plant-wide communication network. Note that in designing each control system, only measurements of the local process variables are used (for example, the controller for the second unit uses only measurements of T_2 and C_{A2}). This decentralized architecture is intended to minimize unnecessary communication costs incurred by continuously sending measurement data from the first to the second unit over the network. We note that while this issue may not be a pressing one for the small plant considered here (where a centralized structure can in fact be easily designed), real plants nonetheless involve a far more complex arrangement of units with thousands of actuators and sensors, which makes the complexity of a centralized structure, as well as the cost of using the network to share measurements between units quite significant. For this reason, we choose the distributed structure in Figure 2 in order to highlight some of the manifestations of the inherent interplays between the control and communication tasks.

The fault-tolerant control problem under consideration involves a total failure in both control systems (Q_1 and Q_2) after some time of startup, with the failure in the first unit being permanent. Our objective will be to preserve closed-loop stability of CSTR 2 by switching to an alternative control configuration involving, as manipulated variables, the rate of heat input $u_2^1 = Q_2$, subject to the same constraint, and the inlet reactant concentration $u_2^2 = C_{A03} - C_{A03}^s$, subject to the constraint $|C_{A03} - C_{A03}^s| \leq u_{max}^{C_{A03}} = 0.4$ kmol/m^3 where $C_{A03}^s = 3.0$ kmol/m^3. The main question, which we address in the next section, is how to devise the switching and network communication logics in a way that ensures fault-tolerance in the second unit and, simultaneously, accounts for the inherent limitations in network resources and possible delays in fault-detection, communication and actuator activation.

Fault-Tolerant Control System Design Methodology

In this section, we outline the main steps involved in the fault-tolerant control system design procedure. These include: (1) the synthesis of a stabilizing feedback controller for each of the available fallback control configurations, (2) the explicit characterization of the stability region for each configuration which characterizes the operating conditions for which fault-recovery can be guaranteed, (3) the design of a switching law that orchestrates the reconfiguration of the failing control system in a way that safeguards closed-loop stability in the event of failures, and (4) the design of the network communication logic in a way that minimizes the propagation of failure effects between plant units while also accounting for bandwidth constraints and delays. A major feature of the design methodology is the inherent coupling between the aforementioned tasks, whereby each task affects how the rest are carried out. Later is a more detailed description of each step, and a discussion on how the tradeoffs between the different steps are managed.

Constrained feedback controller synthesis

Referring to the system of Eq. 1, consider first the case when no failures take place anywhere in the plant. Under such conditions, our objective is to design, for each processing unit, a "nominal" feedback controller that enforces asymptotic closed-loop stability, and provides an explicit characterization of the stability region under actuator constraints. One way to do this is to use Lyapunov-based control techniques. Specifically, consider the nonlinear system describing the i-th processing unit under the k_i-th control configuration, for which a control Lyapunov function $V_i^{k_i}$, is available. Using this function, one can construct the following bounded nonlinear control law (see [39,9])

$$u_i^{k_i} = -r(x_i, u_{i,max}^{k_i})\beta^T(x_i) \tag{3}$$

where

$$r(x_i, u_{i,max}^{k_i}) = \frac{\alpha^*(x_i) + \sqrt{(\alpha^*(x_i))^2 + (u_{i,max}^{k_i}\|\beta^T(x_i)\|)^4}}{\|\beta^T(x_i)\|^2[1 + \sqrt{1 + (u_{i,max}^{k_i}\|\beta^T(x_i)\|)^2}]} \tag{4}$$

$\alpha^*(x_i) = \alpha(x_i) + \rho_i^{k_i}\|x_i\|^2$, $\rho_i^{k_i} > 0$ is a real number, $\alpha(x_i) = L_{f_i^{k_i}}V_i^{k_i}(x_i)$, $\beta^T(x_i) = (L_{G_i^{k_i}}V_i^{k_i})^T(x_i)$, the notation $L_{f_i^{k_i}}V_i^{k_i}$ is used to denote the Lie derivative of the scalar function $V_i^{k_i}$, with respect to the vector field, $f_i^{k_i}$, and $L_{G_i^{k_i}}V_i^{k_i}$ is a row vector whose constituent components are the Lie derivatives of $V_i^{k_i}$ along the column vectors of the matrix $G_i^{k_i}$. Note that the control law of Eqs. 3 and 4 requires measurements of the local process state variables, x_i, only, and not measurements from other plant units upstream. This fully decentralized design is motivated by the desire to minimize unnecessary communication costs which would be incurred when sharing measurement data between the different units over the communication network. By disregarding the interconnections between the units in the controller design, however, closed-loop stability for a given unit rests on the stability properties of the upstream units. In particular, using a combination of Lyapunov and small-gain theorem type arguments, one can show that, starting from any invariant subset (for example, a level-set of $V_i^{k_i}$) of the region described by

$$\Phi_i(u_{i,max}^{k_i}) := \{x_i \in \mathbb{R}^{n_i} : \alpha(x_i) + \rho_i^{k_i}\|x_i\|^2 \le u_{i,max}^{k_i}\|\beta^T(x_i)\|\} \quad (5)$$

the control law of Eqs. 3 and 4 asymptotically stabilizes the i-th unit, under the k_i-th control configuration, at the origin provided that the closed-loop states of the upstream units x_1, $x_2, \ldots, x_{i-1}$, converge asymptotically to the origin. In this case, and because of the way the various units are connected (see Eq. 1), the closed-loop states of the upstream units can be viewed as bounded vanishing perturbations that affect the i-th unit and, therefore, a control law that asymptotically stabilizes the unperturbed i-th unit (that is, disregarding the upstream states) also stabilizes the closed-loop system when the perturbations (connections) are added.

Having designed the nominal feedback control systems, we now proceed to consider the effect of control actuator failure on the feedback controller design for each unit. To this end, let us consider a total failure in the actuators of the k_i-th control configuration in the i-th control system. This failure, if not addressed properly, can lead to closed-loop instabilities both within the i-th processing unit itself (where the failure has occurred), and within all the remaining units downstream. Minimizing the effects of failure propagation throughout the plant can be achieved in one of two ways. The first involves reconfiguring the local control system of the i-th unit—once the failure is detected and isolated—by appropriately switching from the malfunctioning control configuration to some well-functioning fallback configuration (recall that each processing unit has a family of control configurations). If this is feasible and can be done sufficiently fast, then the inherent fault-tolerance of the local control system is sufficient to preserve closed-loop stability not only for the i-th unit with the failing control system, but also for the other units downstream without having to reconfigure their control systems. However, if local fault-recovery is not possible (this can happen, for example, in cases when the failure occurs at times that the state lies outside the stability regions of all the available fallback control configurations; see the next subsection for details), then it becomes necessary to communicate the failure information to the control systems downstream and reconfigure them in order to preserve their closed-loop stability.

The main issue here is how to design the feedback control law for a given fallback configuration in the units downstream in a way that respects the actuators' constraints, and guarantees closed-loop stability despite the failure in the control system of some upstream unit. The choice of the feedback law depends on our choice of the communication policy. To explain this interdependence, we first note that a total failure in the control system of the i-th unit will cause its state x_i, to move away from the origin (possibly settling at some other steady-state). Therefore, unless the nominal feedback controllers for the downstream units $i + 1$, $i + 2, \ldots, l$, are redesigned to account for this incoming "disturbance," the evolution of their states x_{i+1}, $x_{i+2}, \ldots, x_l$, will be adversely affected driving them away from the desired steady-state. To account for the disturbance caused by the upstream control system failure, one option is to send available measurements of x_i, through the communication network, to the affected units and redesign their controllers accordingly. From a communications cost point of view, however, this option may be costly since it requires continued usage of the network resources after the failure, which can adversely affect the performance of other units sharing the same communication medium due to bandwidth limitations and overall delays.

To reduce unnecessary network usage, we propose an alternative approach where the failure in the i-th processing unit is viewed as a bounded nonvanishing disturbance affecting units $i + 1$, $i + 2, \ldots, l$, and use the available process models of these units to capture, or estimate, the size of this disturbance (by comparing, for example, the evolution of the process variables for the i-th unit under the failed and well-functioning control configurations through simulations). In this formulation, state measurements from the i-th unit need not be shared with the other units; instead, only bounds on the disturbance size are transmitted to the downstream units. This approach involves using the network only once at the failure time and not continuously thereafter. The disturbance information can then be used to design an appropriate robust controller for each downstream unit to attenuate the effect of the incoming disturbance and enforce robust closed-loop stability. To illustrate how this can be done, let us assume that the failure in the control system of unit i occurs at $t = T_f$, and that the failure is detected immediately (the effect of possible delays in fault-detection and how to account for them are discussed below in the subsection on communication logic design). Consider some unit j, downstream from the i-th unit, that is described by the following model

$$\dot{x}_j = f_j^{k_j}(x_j) + G_j^{k_j}(x_j)u_j^{k_j} + \delta_i \sum_{p=1}^{i-1} W_{j,p}^{k_j}(x_j)x_p + \sum_{p=i}^{j-1} W_{j,p}^{k_j}(x_j)\theta_p \quad (6)$$

for $i = 1, \ldots, l - 1$, $j = i + 1, \ldots, l$, where $\delta_i = 0$ for $i = 1$, and $\delta_i = 1$ for $i = 2, \ldots, l - 1$. The third term on the righthand side of Eq. 6 describes the input from all the units upstream of unit i. The θ_p's are time-varying, but bounded functions of time that describe the evolution of the states of the i-th unit and all the units downstream from unit i, but upstream from unit j (that is, $\theta_p(t) = x_p(t)$, $p = i, \ldots, j - 1$). The choice of using the notation θ_p, instead of x_p, for units $i, \ldots, j - 1$ is intended to distinguish the effect of these units (where the failure originates and propagates downstream) as nonvanishing disturbances to the j-th unit, compared with the units upstream from unit i which are unaffected by the failure. Note that for unit $j = i + 1$, which immediately follows the failing unit, the only source of disturbances that should be accounted for in its controller design is that coming from the i-th unit with the failing control system. However, for units that lie further downstream, that is, for $j = i + 2, \ldots, l$, the controller design needs to account for the additional disturbances resulting from the effect of the failure on the intermediate units separating units i and j.

For a system of the form of Eq. 6, one possible choice of a stabilizing controller is the following bounded robust Lyapunov-based control law proposed in [10] which has the general form

$$u_j^{k_j} = -r_j(x_j, u_{j,max}^{k_j}, \theta_b)\beta^T(x_j) \tag{7}$$

where

$$r_j(x_j, u_{j,max}^{k_j}, \theta_b) = \frac{\alpha_1(x_j) + \sqrt{(\alpha_2(x_j))^2 + (u_{j,max}^{k_j}\|\beta^T(x_j)\|)^4}}{(\|\beta^T(x_j)\|)^2[1 + \sqrt{1 + (u_{j,max}^{k_j}\|\beta^T(x_j)\|)^2}]} \tag{8}$$

$$\alpha_1(x_j) = \alpha(x_j) + \left(\rho_j^{k_j}\|x_j\| + \sum_{p=i}^{j-1} \chi_j^{k_j}\theta_b^p(T_f)\|\omega_p^T(x_j)\|\right) \times \left(\frac{\|x_j\|}{\|x_j\| + \phi_j^{k_j}}\right) \tag{9}$$

$$\alpha_2(x_j) = \alpha(x_j) + \rho_j^{k_j}\|x_j\| + \sum_{p=i}^{j-1} \chi_j^{k_j}\theta_b^p(T_f)\|\omega_p^T(x_j)\| \tag{10}$$

$\theta_b^p(T_f) := \max_{t \geq T_f}\|x_p(t)\|$, $p = i, \ldots, j-1$ are positive real numbers that capture the size of the disturbances, originating from the failure in the control system of the i-th unit, and propagating downstream, $\omega_p(x_j) = (L_{W_{j,p}^{k_j}}V_j^{k_j})(x_j)$ is a row vector whose constituent components are the Lie derivatives of $V_j^{k_j}$ along the column vectors of the matrix $W_{j,p}^{k_j}$, $V_j^{k_j}$, is a robust control Lyapunov function for the j-th system under the k_j-th control configuration, and $\rho_j^{k_j} > 0$, $\chi_j^{k_j} > 1$, $\phi_j^{k_j} > 0$ are tuning parameters. Estimates of the disturbance bounds θ_b^p, can be obtained by comparing, through simulations, for example, the responses of the p-th unit under the pre- and post-failure configurations (see the simulation studies section for an example). It should be noted that since all the incoming disturbances to unit j take effect only after T_f, the controller of Eqs. 7–10 is implemented only for $t \geq T_f$. For $t < T_f$, the nominal controllers of Eqs. 3 and 4 are used.

Remark 2: When compared with the nominal controller of Eqs. 3 and 4, we observe that the nonlinear gain function for the fallback controller $r_j(\cdot)$ in Eqs. 7–10, depends not only on the size of actuator constraints $u_{j,max}^{k_j}$, and the particular fallback control configuration being used k_j, but also on the size of the disturbances caused by the occurrence of failure θ_b^p. This gain reshaping procedure is carried out in order to guarantee constraint satisfaction, and enforce robust closed-loop stability, with an arbitrary degree of attenuation of the effect of the failure on the j-th unit downstream. Note that, owing to the assumption of a persistent failure in the i-th unit (that is, a nonvanishing disturbance), asymptotic closed-loop stability cannot be achieved for any of the units downstream. Instead, practical stability can be enforced, whereby the states of each unit are driven, in finite-time, to a neighborhood of the origin whose size can be made arbitrarily small by selecting the controller tuning parameters ($\rho_j^{k_j}$, $\chi_j^{k_j}$, $\phi_j^{k_j}$) appropriately (see [14] for a detailed proof). These closed-loop properties are enforced within a well-defined state-space region that is explicitly characterized in the next subsection.

Remark 3: Note that since the processing units upstream of unit i are not affected by its failing control system, the nominal controllers designed for these units (see Eqs. 3 and 4) will asymptotically stabilize their states x_p, $p = 1, \ldots, i-1$, at the origin regardless of the failure; hence, these state can be viewed as bounded vanishing inputs to the j-th unit and, thus, need not be accounted for in the controller design. The terms describing the intermediate units $p = i+1, \ldots, j-1$ cannot, however, be treated as vanishing inputs. The reason is that even if the control systems of these units are immediately and appropriately reconfigured to suppress the effect of the failure, their controllers, as discussed earlier, will at best be able to drive the states of these units, in finite time, only near the origin without achieving asymptotic convergence. Finally, we note that our framework can handle incipient failures in upstream units by treating them as slowly-varying disturbances in the downstream units through the robust controller design. This is possible because the robust nonlinear controller design requires only a bound on the magnitude of the disturbance, and does not impose any limitations on the rate of change of the disturbance.

Remark 4: It should be noted that the fault-tolerant control system design methodology proposed in this section is not restricted to the use of the bounded controller designs given in Eqs. 3 and 4 (for the nominal case) and in Eqs. 7–10 (for the case with failure). Any other stabilizing controller design that accounts for the constraints, enforces the desired robustness properties under failure, and provides an explicit characterization of the stability region can be used, including the recently-developed hybrid predictive control algorithms,[40,41,17,42] which embed the implementation of predictive controllers within the explicitly-characterized stability region of Lyapunov-based nonlinear bounded controllers.

Remark 5: Control Lyapunov function (CLF)-based stabilization of nonlinear systems has been studied extensively in the nonlinear control literature (for example, see [39,43,44]). The construction of constrained CLFs (that is, CLFs that take the constraints into account) remains a difficult problem (especially for nonlinear systems) that is the subject of ongoing research. For several classes of nonlinear systems that arise commonly in the modeling of practical systems, systematic and computationally feasible methods are available for constructing unconstrained CLFs (CLFs for the unconstrained system), by exploiting the system structure. Examples include the use of quadratic functions to construct CLFs. In this work, the bounded controllers in Eqs. 3 and 4 and Eqs. 7–10 are designed using unconstrained CLFs, which are also used to explicitly characterize the associated stability regions. While the resulting estimates do not necessarily capture the entire domain of attraction, we will use them throughout the article only for a concrete illustration of the basic ideas of the results. It is possible to obtain estimates using other methods, such as Zubov's method[45] and a combination of several CLFs which can yield substantially less conservative estimates.

Remark 6: The treatment of the failure in the control system of unit i as a bounded disturbance is rooted in the assumption that x_i, while moving away from the origin after failure, will eventually settle at some other (undesirable) steady-state (recall that this is how the disturbance bound is computed). In the case when the i-th processing unit has only a single steady-state in the post-failure configuration, however, the failure cannot be treated as a bounded disturbance since x_i will simply grow unbounded after the failure and not settle anywhere. In such a case, unless the control system of unit i is fixed in time, a shutdown of the plant will be unavoidable.

Characterization of fault-recovery regions

Consider once again the j-th processing unit described by the model of Eq. 6. In the previous section, we outlined how to design, for a given fallback control configuration $k_j \in \mathcal{K}_j$, a robust feedback controller that, when implemented, can preserve closed-loop stability for this unit in the event of control system failure in some upstream unit, i. Given that actuator constraints place fundamental limitations on the ability of the controller to steer the closed-loop dynamics at will, it is important for the control system designer to explicitly characterize these limitations by identifying, or estimating, the set of admissible states starting from where the controller of Eqs. 7–10 is guaranteed to robustly stabilize the closed-loop system for unit j (region of robust closed-loop stability). Since suppression of the upstream failure effects on unit j is formulated as a robust stabilization problem, we shall refer to the robust stability region associated with any of the fallback configurations, also as the fault-recovery region. As discussed in the next subsection, the characterization of this region plays a central role in devising the appropriate switching policy that reconfigures the control system and ensures fault-recovery.

For the class of robust control laws given in Eqs. 7–10, using a Lyapunov argument, one can show that the set

$$\Pi_j^{k_j}(u_{j,max}^{k_j},\ \theta_b(T_f)) := \Bigg\{ x_j \in \mathbb{R}^{n_j} : \alpha(x_j) + \rho_j^{k_j}\|x_j\| + \sum_{p=i}^{j-1} \chi_j^{k_j}\theta_b^p(T_f)\|\omega_p^T(x_j)\| \leq u_{j,max}^{k_j}\|\beta^T(x_j)\| \Bigg\} \tag{11}$$

describes a region in the state space where the control action satisfies the constraints, and the Lyapunov function decays monotonically along the trajectories of the closed-loop system outside of a small neighborhood around the origin (see [10] for the detailed mathematical analysis). Note that the size of this set depends both on the magnitude of the constraints and the size of the disturbance (which in turn depends on the failure time, T_f). In particular, as the constraints get tighter and/or the disturbances greater, the set becomes smaller. Since $\Pi_j^{k_j}$, however, is in general, not an invariant set, there is no guarantee that a trajectory starting within $\Pi_j^{k_j}$ will remain within it for all the times that the k_j-th control configuration is active, that is, $\Pi_j^{k_j}$ by itself is not necessarily a stability region. One way to estimate the fault-recovery region associated with a given control configuration using Eq. 11 is to construct an invariant subset—preferably the largest—within $\Pi_j^{k_j}$, which we denote by $\Omega_j^{k_j}(u_{j,max}^{k_j},\ \theta_b(T_f))$ (for example, $\Omega_j^{k_j}$ can be chosen as a level-set of $V_j^{k_j}$). For a given fallback configuration k_j, implementation of the controller of Eqs. 7–10 at any time that the state is within $\Omega_j^{k_j}$ ensures that the closed-loop trajectory stays within the region defined by $\Pi_j^{k_j}$—and, hence, $V_j^{k_j}$ continues to decay monotonically outside of a small neighborhood around the origin—for all the times that the k_j-th configuration is active. The estimate provided by $\Omega_j^{k_j}$ can be conservative but can also be improved using computer simulations. This approach was followed in the simulation examples in order to obtain appropriate estimates of the fault-recovery regions.

Remark 7: Note that, unlike the nominal stability regions associated with the nominal controllers of Eqs. 3 and 4 and obtained from Eq. 5, the fault-recovery region of any downstream unit j, cannot be computed *a priori* (that is, before plant startup) since this region, as can be seen from Eq. 11, depends on the failure time which is unknown prior to startup. However, once the failure occurs, estimates of the disturbance bounds can be computed by the local supervisors of the upstream units $i, \ldots, j-1$ (through on-line simulations of each unit's response under the pre- and post-failure configurations) and then transmitted, through the communication network, to unit j which in turn uses these bounds to construct, on-line, both the controller and the fault-recovery region (see the subsection on communication logic for a discussion on how the resulting computational delays can be handled).

Supervisory switching logic design

Having designed the robust feedback control law and characterized the fault-recovery region associated with each fallback configuration, the third step in our design methodology is to derive the switching policy that the local supervisor of the downstream unit j, needs to follow in reconfiguring the local control system (that is, activating/deactivating the appropriate fallback configurations) in the event of the upstream failure. In the general case when more than one fallback control configuration is available for the unit under consideration, the question is how to decide which of these configurations can and should be activated at the time of failure in order to preserve closed-loop stability. The key idea here is that because of the limitations imposed by constraints on the fault-recovery region of each configuration, the local supervisor can only activate the configuration whose fault-recovery region contains the closed-loop state at the time of the failure. Without loss of generality, let the active control configuration in the j-th unit, prior to the occurrence of failure in unit i, be $k_j(T_f^-) = \mu$ for some $\mu \in \mathcal{K}_j$, where $k_j(T_f^-) = \lim\limits_{t \to T_f^-} k_j(t)$ and T_f is the time that the control system of unit i fails, then the switching rule given by

$$k_j(t) = \nu,\ \forall\, t \geq T_f^+, \text{ if } x_j(T_f) \in \Omega_j^{\nu}(u_{j,max}^{\nu},\ \theta_b(T_f)) \tag{12}$$

for some $\nu \in \mathcal{K}_j$, $\nu \neq \mu$, guarantees that the closed-loop system of the j-th unit is stable. The implementation of the above switching law requires monitoring, by the local supervisor, of the evolution of the closed-loop state trajectory with respect to the fault-recovery regions associated with the various control actuator configurations. Another way to look at the earlier switching logic is that it implicitly determines, for a fixed fallback configuration, the times that the control system of the j-th unit can tolerate upstream failures by switching to this configuration. If failure occurs at times when x_j lies outside the fault-recovery region of all available configurations, this analysis suggests that either the constraints should be relaxed—to enlarge the fault-recovery region of the given configurations—or additional fallback control loops must be introduced. The second option, however, is ultimately limited by the maximum allowable number of control loops that can be designed for the given processing unit. If neither option is feasible, a shutdown could be unavoidable. The proposition of constructing the switching logic on the basis of the stability regions was first proposed in [12] for the control of switched nonlinear systems.

Design of the communication logic

Given the distributed interconnected nature of the plant units—and, thus, the potential for failure effects propagating from one unit to another—an essential element in the design of the fault-tolerant control system is the use of a communication medium that ensures fast and efficient transmission of information during failure events. As discussed in the introduction, communication networks offer such a medium that is both fast (relative to the typically slow dynamics of chemical processes) and inexpensive (relative to current point-to-point connection schemes which require extensive cabling and higher maintenance time and costs). The ability of the network to fulfill this role, however, requires that we devise the communication policy in a way that respects the inherent limitations in network resources, such as bandwidth constraints and overall delays, by minimizing unnecessary usage of the network.

In the section on feedback controller synthesis, we have already discussed how the bandwidth constraint issue can be handled by formulating the problem as a robust control problem, where the failure in the control system of the i-th processing unit and the subsequent effects on units $i + 1, \ldots, j - 1$ are treated as a bounded nonvanishing disturbances that affect unit j downstream. The communication policy requires that the local supervisors of units $i, \ldots, j - 1$ perform the following tasks: (1) compute the disturbance bounds using the process model of each unit, and (2) send this information, together with other relevant information, such as the failure type, the failure time and operating conditions, to the plant supervisor. The plant supervisor in turn forwards the information to the local supervisor of unit j utilizing the plant-wide communication network (see Figure 1b). This policy avoids unnecessary overloading of the network (which could result when measurements from the upstream units are sent continuously to unit j), while also guaranteeing fault-tolerance in the downstream units. The idea of using knowledge of the plant dynamics to balance the tradeoff between bandwidth limitations (which favor reduced communication of measurements) and optimum control performance (which favors increased communication of measurements) is conceptually aligned with the notion of minimum attention control (for example, see [46,27]). In our work, however, this idea is utilized in the context of fault-tolerant control.

The second consideration in devising the communication logic is the issue of time delays which typically result from the time sharing of the communication medium, as well as the computing time required for the physical signal coding and communication processing. The characteristics of these time delays depend on the network protocols adopted as well as the hardware chosen. For our purposes here, we consider an overall fixed time delay (which we denote by τ^j_{max}) that combines the contribution of several delays, including: (1) delays in fault-detection, (2) the time that the local supervisors of units $i, \ldots, j - 1$ take to compute the effective disturbance bounds (through simulations comparing the pre- and post-failure state evolutions in each unit), (3) the time that the local supervisors of units $i, \ldots, j - 1$ take to send the information to the plant supervisor, (4) the time that it takes the plant supervisor to forward the information to the local supervisor of unit j, (5) the time that it takes the local supervisor for unit j to compute the fault-recovery region for the given fallback configurations using the information arriving from the upstream units, and the time that it takes for the supervisor's decision to reach and activate the appropriate fallback configuration, and (6) the inherent actuator/sensor dead-times.

Failure to take such delays into account can result in activating the wrong control configuration and subsequent instability. For example, even though the upstream failure may take place at $t = T_f$, the fallback configuration in the control system of unit j will not be switched in before $t = T_f + \tau^j_{max}$. If the delay is significant, then the switching rule in the previous section should be modified such that the local supervisor for unit j activates configuration $k_j = \nu$, for which $x_j(T_f + \tau^j_{max}) \in \Omega^\nu_j(u^\nu_{j,max}, \theta_b)$. This modification is yet another manifestation of the inherent coupling between the switching and communication logics. The implementation of the modified switching rule that accounts for delays requires that the local supervisor of unit j be able to predict where the state trajectory will be at $t = T_f + \tau^j_{max}$ (for example, through simulations using the process model), and check whether the state at this time is within the fault-recovery region of a given fallback configuration. If not, then either an alternative fallback configuration, for which the fault-recovery region contains the state at the end of the delay, should be activated, or a shutdown maybe unavoidable. The availability of several fallback control loops, however, is limited by process design considerations which dictate, for example, how many variables can be used for control. Figure 3 summarizes the overall fault-tolerant control strategy for a two-unit plant.

Simulation Studies

In this section, we present two simulation studies that demonstrate the application of the proposed fault-tolerant control system design methodology to two chemical processes. In the first application, a single chemical reactor example is considered to demonstrate the idea of reconfiguring the local control system in the event of failures on the basis of the stability regions of the constituent control configurations, and how overall communication delays impact the reconfiguration logic. In the second application, a cascade of two chemical reactors in series is considered to demonstrate how the issue of failure propagation between a multiunit plant is handled within the proposed methodology, and how the various interplays between the feedback, supervisory control and communication tasks are handled in the multiunit setting.

Application to a single chemical reactor

Consider a well-mixed, nonisothermal continuous stirred-tank reactor, where three parallel irreversible elementary exothermic reactions of the form $A \xrightarrow{k_1} B$, $A \xrightarrow{k_2} U$, and $A \xrightarrow{k_3} R$ take place, where A is the reactant species, B is the desired product, and U, R are undesired byproducts. The feed to the reactor consists of pure A at flow rate F, molar concentration C_{A0}, and temperature T_{A0}. Due to the nonisothermal nature of the reactions, a jacket is used to remove/provide heat to the reactor.

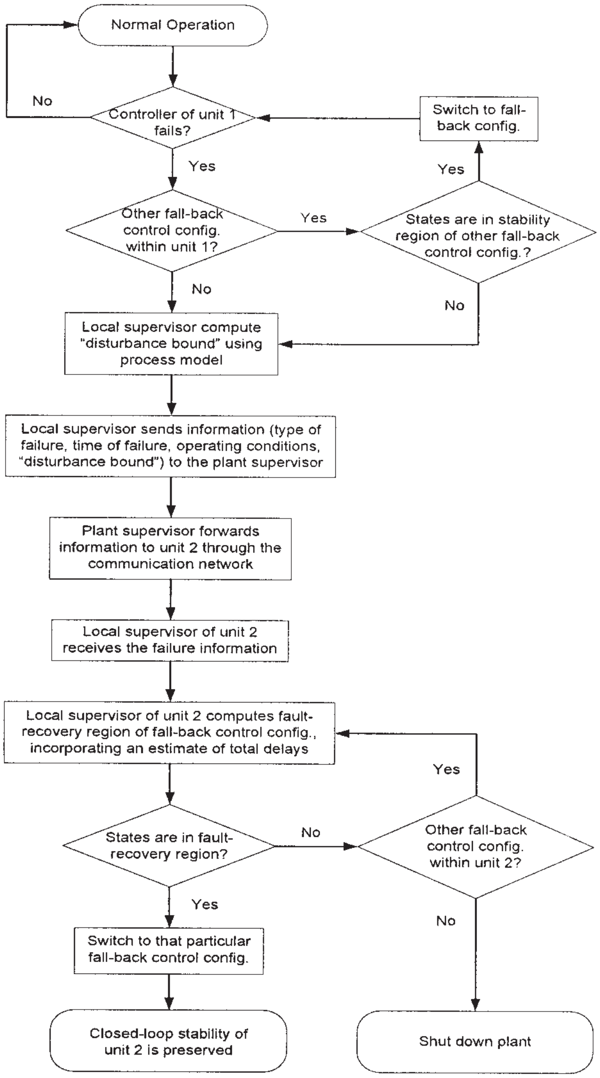
Normal Operation
No
Controller of unit 1 fails?
Yes
Other fall-back control config. within unit 1?
Yes
States are in stability region of other fall-back control config.?
Yes
Switch to fall-back config.
No
No
Local supervisor compute "disturbance bound" using process model
Local supervisor sends information (type of failure, time of failure, operating conditions, "disturbance bound") to the plant supervisor
Plant supervisor forwards information to unit 2 through the communication network
Local supervisor of unit 2 receives the failure information
Local supervisor of unit 2 computes fault-recovery region of fall-back control config., incorporating an estimate of total delays
Yes
States are in fault-recovery region?
No
Other fall-back control config. within unit 2?
Yes
No
Switch to that particular fall-back control config.
Closed-loop stability of unit 2 is preserved
Shut down plant

Figure 3. Summary of the fault-tolerant control strategy, for a two-unit plant, using communication networks.

Under standard modeling assumptions, a mathematical model of the process can be derived from material and energy balances, and takes the following form

$$\frac{dT}{dt} = \frac{F}{V}(T_{A0} - T) + \sum_{i=1}^{3} \frac{(-\Delta H_i)}{\rho c_p} R_i(C_A, T) + \frac{Q}{\rho c_p V}$$

$$\frac{dC_A}{dt} = \frac{F}{V}(C_{A0} - C_A) - \sum_{i=1}^{3} R_i(C_A, T)$$

$$\frac{dC_B}{dt} = -\frac{F}{V} C_B + R_1(C_A, T) \qquad (13)$$

where $R_i(C_A, T) = k_{i0}\exp(-E_i/RT)C_A$, C_A and C_B denote the concentrations of the species A and B, respectively, T denotes the temperature of the reactor, Q denotes the rate of heat input to the reactor, V denotes the volume of the reactor, ΔH_i, k_i, E_i, $i = 1, 2, 3$, denote the enthalpies, pre-exponential constants and activation energies of the three reactions, respectively, c_p and ρ denote the heat capacity and density of fluid in the reactor. The values of the process parameters and the corresponding steady-state values are given in Table 1. It was verified that under these conditions, the process model of Eq. 13 has three steady-states: two locally asymptotically stable, and one unstable at (T^s, C_A^s, C_B^s) = (388 K, 3.59 kmol/m^3, 0.41 kmol/m^3).

The control objective is to stabilize the reactor at the (open-loop) unstable steady-state. Operation at this point is typically sought to avoid high temperatures while, simultaneously, achieving reasonable reactant conversion. To accomplish this objective in the presence of control system failures, we consider the following manipulated input candidates:

1. Rate of heat input $u_1 = Q$, subject to the constraint $|Q| \leq u_{max}^1 = 2.7 \times 10^6$ KJ/hr.
2. Inlet stream temperature $u_2 = T_{A0} - T_{A0}^s$, subject to the constraint $|u_2| \leq u_{max}^2 = 100$ K.
3. Inlet reactant concentration $u_3 = C_{A0} - C_{A0}^s$, subject to the constraint $|u_3| \leq u_{max}^3 = 4$ kmol/m^3.

Each of the earlier manipulated inputs represents a unique control configuration (or control-loop) that, by itself, can stabilize the reactor using available measurements of the reactor temperature, reactant and product concentrations provided by the sensors. The sensors and control actuators of each configuration are connected to the unit supervisor (for example, a distant control room) over a communication network (see Figure 4). The first loop involving the heat input Q, as the manipulated variable will be considered as the primary control configuration. In the event of a total failure in this configuration, however, the supervisor will have to activate one of the other two fallback configurations in order to maintain closed-loop stability. The main question that we address in this simulation study is how can the supervisor determine which control loop to activate once failure is detected in the active configuration, and how overall communication delays influence this decision.

Following the proposed methodology, we initially synthe-

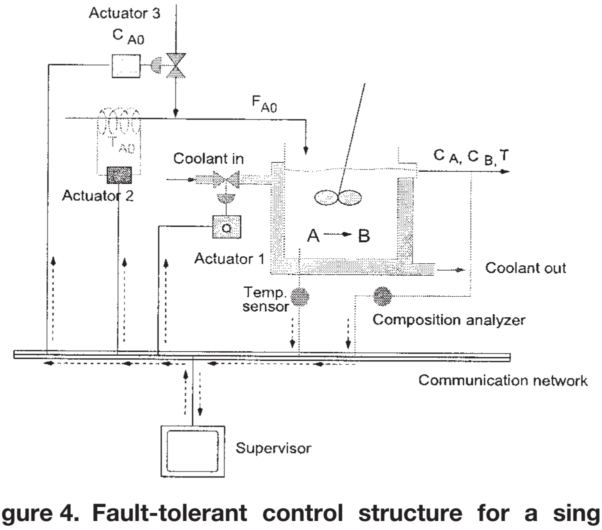

Figure 4. Fault-tolerant control structure for a single unit operation, integrating supervisory and feedback control over a communication network.

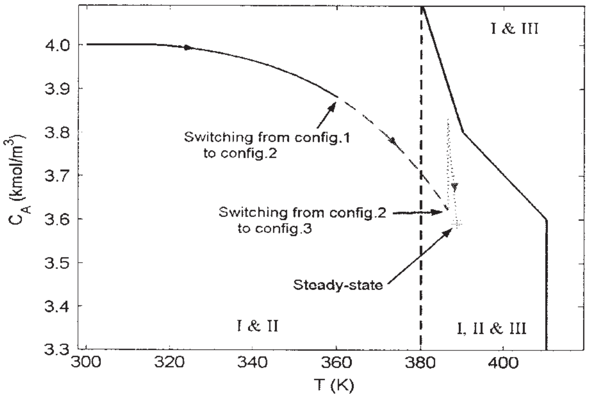

Figure 5. Stability regions of the three control configurations (I, II, III) considered for the chemical reactor example of Eq. 13.

size, for each control configuration, a feedback controller that enforces asymptotic closed-loop stability in the presence of actuator constraints. This task is carried out on the basis of the process input/output dynamics. While our control objective is to achieve full-state stabilization, auxiliary process outputs are introduced here to facilitate transforming the system of Eq. 13 into a form more suitable for explicit controller synthesis. In the case of the process of Eq. 13, a further simplification can be obtained by noting that C_B does not affect the evolution of either T or C_A and, therefore, the controller design can be addressed on the basis of the T and C_A equations only. A controller that stabilizes the (T, C_A) subsystem also stabilizes the entire closed-loop system. For the first configuration with $u_1 = Q$, we consider the output $y_1 = (C_A - C_A^s)/C_A^s$. This choice yields a relative degree of $r_1 = 2$ for the output with respect to the manipulated input. The coordinate transformation (in error variables form) takes the form: $e_1 = (C_A - C_A^s)/C_A^s$, $e_2 = (F/V)(C_{A0} - C_A)/C_A^s - \Sigma_{i=1}^3 k_{i0}\exp(-E_i/RT)C_A/C_A^s$. For the second configuration with $u_2 = T_{A0} - T_{A0}^s$, we choose the output $y_2 = (C_A - C_A^s)/C_A^s$, which yields the same relative degree as in the first configuration $r_2 = 2$, and the same coordinate transformation. For the third configuration, with $u_3 = C_{A0} - C_{A0}^s$, we choose the output $y_3 = (T - T^s)/T^s$, which yields a relative degree of $r_3 = 2$, and a coordinate transformation of the form: $e_1 = (T - T^s)/T_s$, $e_2 = (F/V)(T_{A0} - T)/T^s + \Sigma_{i=1}^3 [(-\Delta H_i)/\rho c_p T_s]R_i(C_A, T) + Q/\rho c_p V T_s$.

Note that since our objective is full-state stabilization, the choice of the output in each case is really arbitrary. However, to facilitate the controller design and subsequent stability analysis, we have chosen in each case an output that produces a system of relative degree 2. For each configuration, the corresponding state transformation yields a system, describing the input/output dynamics, of the following form

$$\begin{aligned}\dot{e} &= Ae + l_k(e) + b\alpha_k u_k \\ &:= \bar{f}_k(e) + \bar{g}_k(e)u_k, \quad k = 1, 2, 3 \end{aligned} \qquad (14)$$

where $A = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}$, $b = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$, $l_k(\cdot) = L_{f_k}^2 h_k(x)$, $\alpha_k(\cdot) = L_{g_k}L_{f_k}h_k(x)$, $h_k(x) = y_k$ is the output associated with the k-th configuration, $x = [x_1 \quad x_2]^T$ with $x_1 = (T - T^s)/T^s$, $x_2 = (C_A - C_A^s)/C_A^s$, and the functions $f_k(\cdot)$ and $g_k(\cdot)$ can be obtained by rewriting the (T, C_A) model equations in Eq. 13 in the form of Eq. 1. The explicit forms of these functions are omitted for brevity. Using a quadratic Lyapunov function of the form $V_k = e^T P_k e$, where P_k is a positive-definite symmetric matrix that satisfies the Riccati inequality $A^T P_k + P_k A - P_k b b^T P_k < 0$, we synthesize, for each control-loop, a bounded nonlinear feedback control law of the form of Eqs. 3 and 4 and characterize the associated stability region with the aid of Eq. 5. Figure 5 depicts the stability region, in the (T, C_A) space, for each configuration. The stability region of configuration 1 includes the entire area of the plot. The stability region of configuration 2 is the entire area to the left of the solid line, while the stability region of configuration 3 covers the area to the right of the dashed vertical line. The desired steady-state is depicted with an asterisk that lies in the intersection of the three stability regions.

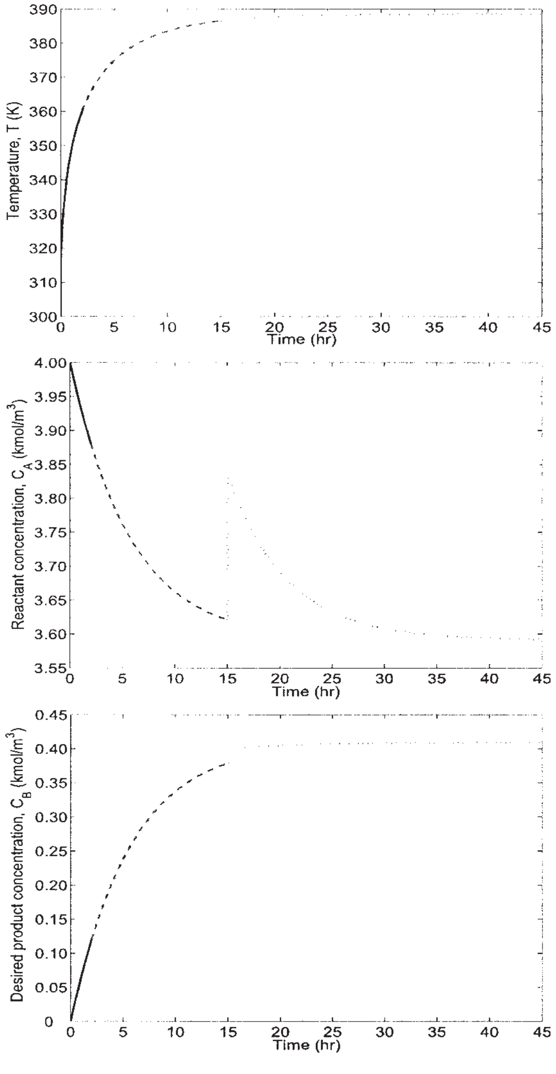

Figure 6. Evolution of the closed-loop state profiles under repeated control system failures and subsequent switching by the supervisor from configuration 1 (solid lines) to configuration 2 (dashed lines) to configuration 3 (dotted lines).

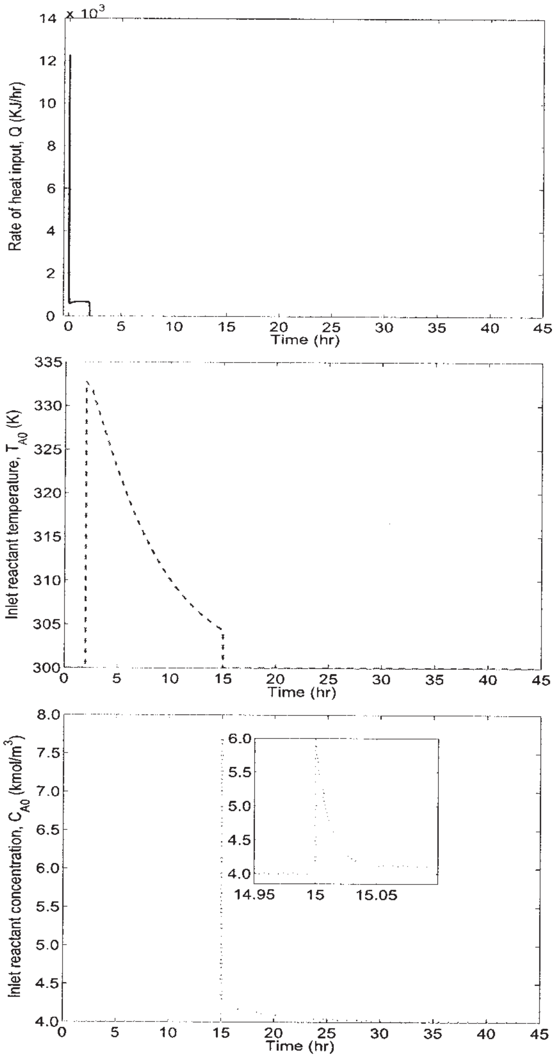

Figure 7. Manipulated input profiles for each control configuration as the supervisor switches from configuration 1 to configuration 2 at t = 2 h, and from configuration 2 to configuration 3 at t = 15 h.

We first consider the case when no time-delays are involved, and the supervisor can switch immediately between the different control loops in the event of failures. To this end, the reactor is initialized at $T(0) = 300$ K, $C_A(0) = 4.0$ kmol/m^3, $C_B(0) = 0.0$ kmol/m^3, using the Q control configuration, and the supervisor proceeds to monitor the evolution of the closed-loop trajectory. As shown by the solid parts of the closed-loop trajectory in Figure 5, the state profiles in Figure 6, and the rate of heat input profile in Figure 7, the controller proceeds to drive the closed-loop trajectory towards the desired steady-state until the actuator of the Q-configuration experiences a total failure after 2.0 h of startup (simulated by fixing $Q = 0$ for all $t \geq 2.0$ h). From the solid part of the trajectory in Figure 5, it is clear that the failure of the primary control configuration occurs when the closed-loop trajectory is within the stability region of the second control configuration, and outside the stability region of the third control configuration. Therefore, on the basis of the switching logic, the supervisor immediately activates the second configuration, with T_{A0} as the manipulated input. The result is shown by the dashed parts of the closed-loop trajectory in Figure 5, the state profiles in Figure 6 and the inlet stream temperature profile in Figure 7 where it is seen that, upon switching to the T_{A0} configuration, the corresponding controller continues to drive the closed-loop trajectory closer to the desired steady-state. At $t = 15.0$ h, we consider another total failure in the control actuators of the T_{A0} configuration (simulated by fixing T_{A0} for all $t \geq 15.0$ h). From the dashed part of the trajectory in Figure 5, it is clear that this failure occurs when the closed-loop trajectory is within the stability region of the third configuration. Therefore, the supervisor immediately activates the third control configuration, with C_{A0} as the manipulated input, which then successfully stabilizes the reactor at the desired steady-state (see the dotted parts of the closed-loop trajectory in Figure 5, the state profiles in Figure 6, and the inlet reactant concentration in Figure 7).

To demonstrate the effect of delays on the implementation of the switching logic, we consider an overall delay, between the supervisor and the constituent control configurations, of $\tau_{max} = 8.0$ min (accounting for possible delays in fault-detection, control computations, network transmission and actuator activation). In this case, the reactor is initialized at $T(0) = 300$ K, $C_A(0) = 4.0$ kmol/m^3, $C_B(0) = 0$ kmol/m^3 under the first control configuration (with Q as the manipulated input). The actual failure of this configuration occurs at $t = 10$ h which, as can be seen from Figure 8, is a time when the closed-loop state trajectory is within the intersection of all three stability regions. In the absence of delays, this suggests that switching to either configuration 2 or 3 should preserve closed-loop stability. We observe, however, from Figure 9 that, when the delay is present, activation of configuration 3 leads to instability (dotted profiles), while activation of configuration 2 achieves stabilization at the desired steady-state (dashed profiles). The reason is the fact that, for the time period between the actual failure ($t = 10$ h), and the activation of the backup configuration ($t = 10.13$ h), the process evolves in an open-loop fashion leading the trajectory to move out of the intersection zone such that at $t = 10.13$ h the state is within the

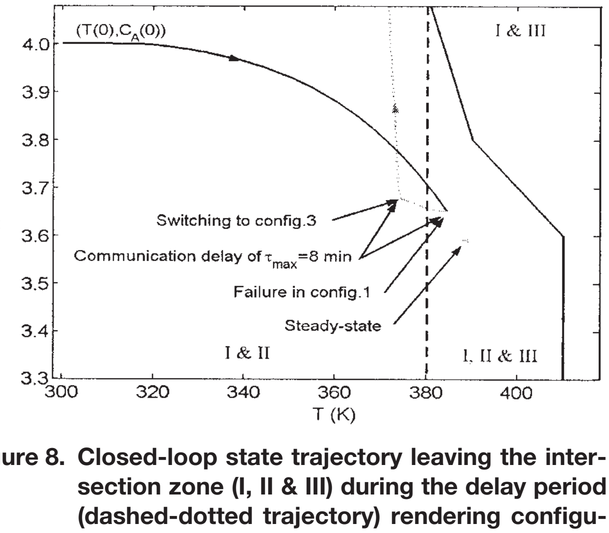

Figure 8. Closed-loop state trajectory leaving the intersection zone (I, II & III) during the delay period (dashed-dotted trajectory) rendering configuration 3 destabilizing (dotted trajectory).

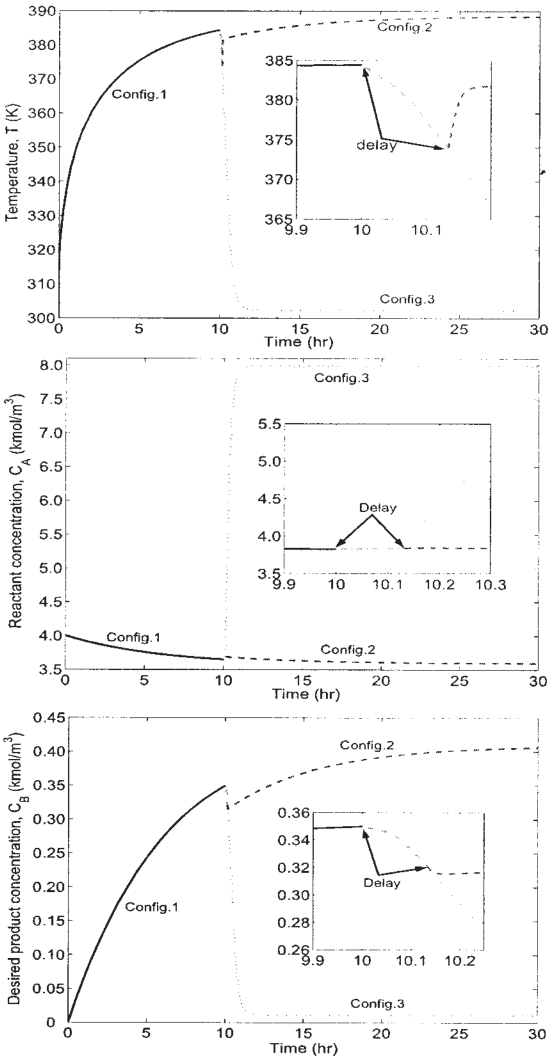

Figure 9. Evolution of the closed-loop state profiles when configuration 1 (solid lines) fails at *t* = 10 h, and an overall delay of τ_{max} = 8.0 min elapses before the backup configuration is activated.

Activation of configuration 2 preserves closed-loop stability (dashed lines) while activation of configuration 3 leads to instability (dotted lines).

stability region of configuration 2 and outside that of configuration 3 (see Figure 8). The corresponding manipulated input profiles are shown in Figure 10. To activate the correct configuration in this case, the supervisor needs to predict where the state trajectory will be at the end of the communication delay period.

Application to two chemical reactors in series

In this section, we revisit the two chemical reactors in series of Eq. 2, introduced earlier in the motivating example section, to illustrate the implementation of the proposed fault-tolerant control methodology. To this end, the reactors are initialized at $(T_1(0), C_{A1}(0))$ = (300 K, 4.0 kmol/m^3), and $(T_2(0), C_{A2}(0))$ = (440 K, 4.0 kmol/m^3). Under normal operating conditions (with no failures), each reactor is controlled by manipulating the rate of heat input, using a bounded nonlinear control law of the form of Eqs. 3 and 4.

For the first CSTR, the controller design procedure is the same as the one used for the Q configuration in the previous simulation example. For the second CSTR, we design the controller on the basis of the temperature equation only. Specifically, a quadratic function of the form $V_2 = \frac{1}{2}a_2(x_2^{(1)})^2$, where $x_2^{(1)} = (T_2 - T_2^s)/T_2^s$, is used to design the controller and estimate the resulting stability region using Eq. 5. The values of the controller tuning parameters are chosen as a_2 = 0.5 and ρ_2 = 0.0001. Figure 11 (solid profiles) and Figure 12 show the resulting closed-loop state and manipulated input profiles when the controllers are implemented without failure for both reactors. We observe that each controller successfully stabilizes the corresponding reactor at the desired steady-state.

Consider now a total failure in the actuators of both control systems (Q_1 and Q_2) at T_f = 5 min. In this case, both reactors

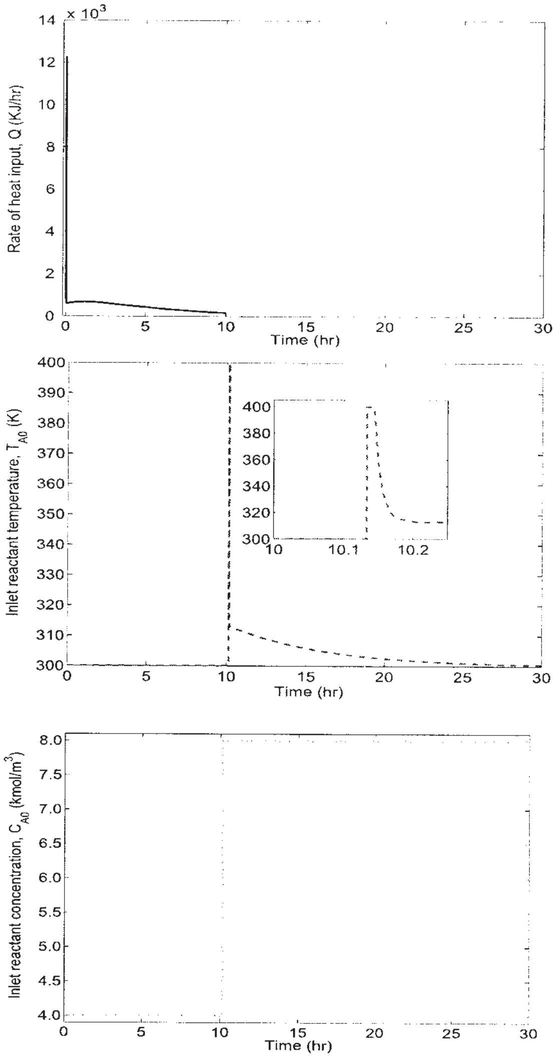

Figure 10. Manipulated input profiles when configuration 1 fails at *t* = 10 h, and an overall delay of τ_{max} = 8.0 min elapses before the backup configuration is activated.

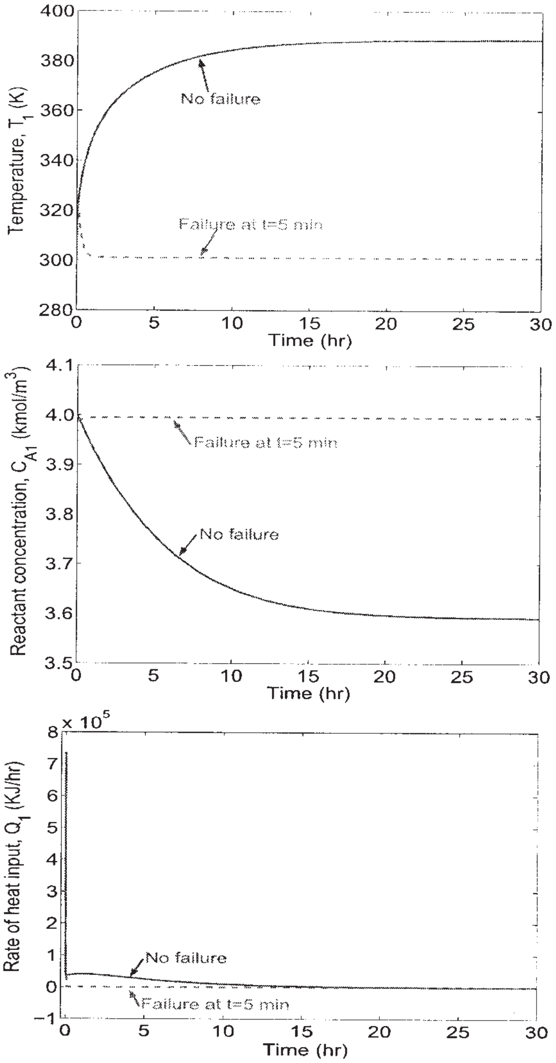

Figure 11. Evolution of the closed-loop state and manipulated input profiles for CSTR 1 under a well-functioning control system (solid), and when the control actuator fail at t = 5 min (dashed).

revert to their open-loop mode of behavior and, consequently, if no fallback control configuration is activated, the states move away from the desired steady-state, as shown by the dashed lines in Figure 11 for the first reactor, and Figure 13 for the second reactor (note that C_{A03} remains fixed for all times since it is not used as a manipulated variable in the prefailure configuration). As stated in the motivating example subsection, we assume that the controller failure in the first reactor is permanent; and our objective is to prevent the propagation of this effect to the second reactor. A permanent failure in the first unit could be the result of lack of sufficient fallback configurations, or because failure occurs at a time when the state is outside the stability regions of all the available configurations for this unit.

Using the proposed methodology, the supervisor of CSTR 1, at the failure time, runs both open-loop and closed-loop simulations using the process model of CSTR 1 to estimate the size of the disturbance affecting CSTR 2, and transmits this information to the local supervisor of CSTR 2 through the communication network. The maximum disturbance size is proportional to the largest discrepancy (after the failure time) between the values of C_{A1}, T_1 in the well-functioning (solid lines in Figure 11) and in the failed (dashed lines in Figure 11) modes. Using this information, the local supervisor of CSTR 2 designs a robust control law of the form of Eqs. 7–10 to stabilize CSTR 2, using the available fallback configuration with (Q_2, C_{A03}) as the manipulated inputs, and constructs the associated fault-recovery region for this configuration. The controller design procedure involves rewriting the process model of CSTR 2 in Eq. 2 in the form of Eq. 6, using the dimensionless variables, $x_i^{(1)} = (T_i - T_i^s)/T_i^s$, $x_i^{(2)} = (C_{Ai} - C_{Ai}^s)/C_{Ai}^s$, $i = 1, 2$, and with the states of CSTR 1 redefined as the disturbance variables $\theta_1(t) = [\theta_1^{(1)}(t) \quad \theta_1^{(2)}(t)]^T$, where $\theta_1^{(1)}(t) = (F_1T_1^s/V_2T_2^s)(x_1^{(1)}(t) + 1)$ and $\theta_1^{(2)}(t) = (F_1C_{A1}^s/V_2C_{A2}^s)(x_1^{(2)}(t) + 1)$, for all $t \geq T_f$. Then, using a quadratic function of the form $V_2 = \frac{1}{2}a_2(x_2^{(1)})^2 + \frac{1}{2}a_2(x_2^{(2)})^2$, the controller of Eqs. 7–10 is constructed and its fault-recovery region is computed with the

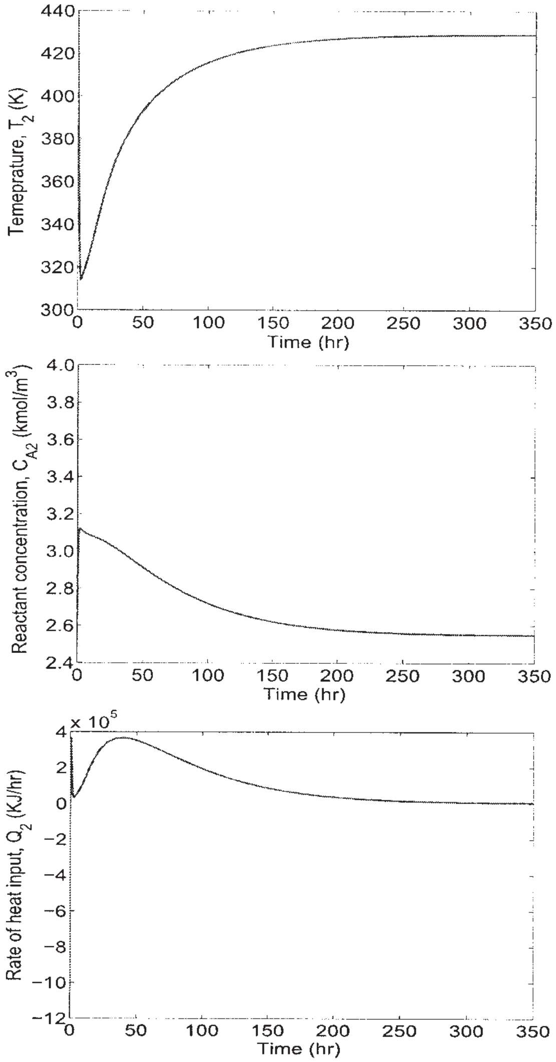

Figure 12. Evolution of the closed-loop state and manipulated input profiles for CSTR 2 under a well-functioning control system.

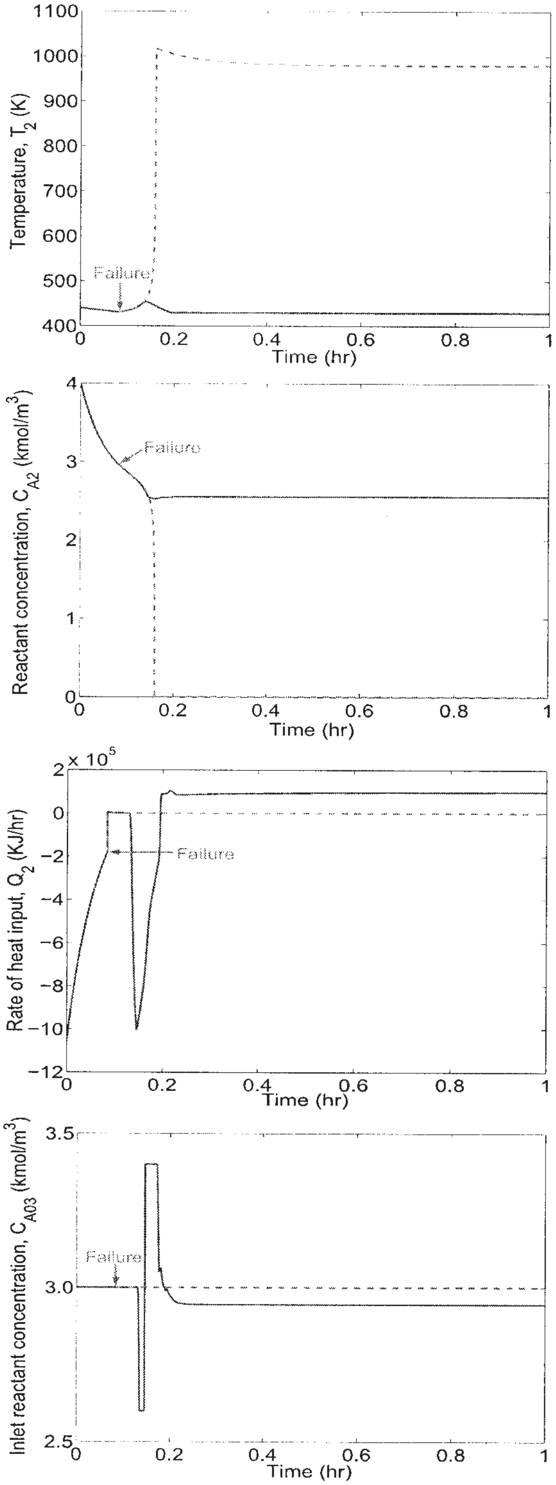

Figure 13. Evolution of the closed-loop state and manipulated input profiles for CSTR 2 when the controller of the fallback configuration (Q_2, C_{A03}) is activated immediately after the failure (solid lines), and the open-loop state and input profiles resulting when the fallback configuration is not activated after the failure (dashed lines).

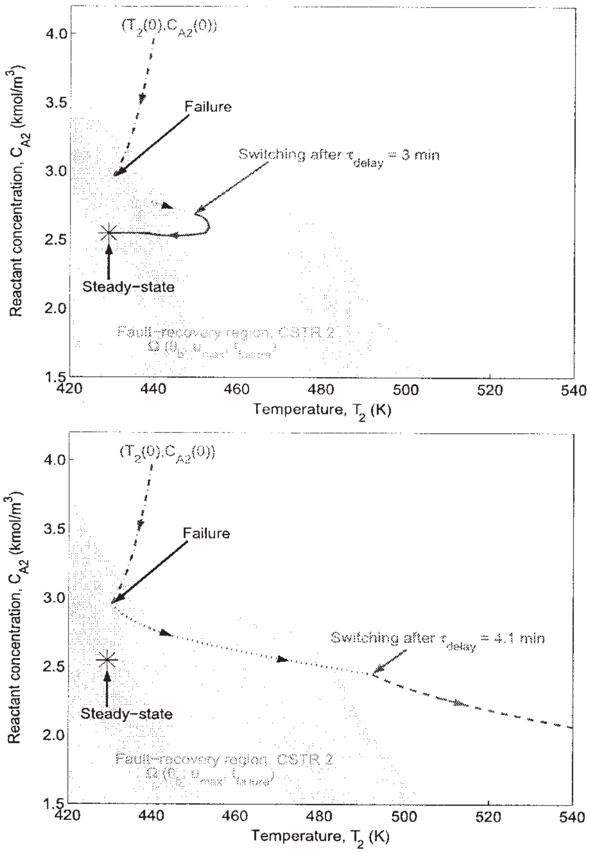

Figure 14. Fault-recovery region of the fallback control configuration (Q_2, C_{A03}) for CSTR 2, with constraints $|Q_2| \leq 2.8 \times 10^6$ KJ/hr and $|C_{A03} - C^s_{A03}| \leq 0.4$ kmol/m^3 when failure occurs at T_f = 5 min.

Activation of the fallback configuration after a 3 min delay preserves closed-loop stability (top plot), while activation after 4.1 min delay fails to ensure fault-tolerance (bottom plot).

aid of Eq. 11. The disturbance bound is computed as $\theta_b^1 = \sup_{t \geq T_f} \|\theta_1(t)\|$. The values of the controller tuning parameters are selected to be $a_2 = 0.5$, $\rho_2 = 0.0001$, $\chi_2 = 2.0001$ and $\phi_2 = 0.0001$. The fault-recovery region is depicted by the shaded area in Figure 14.

From Figure 14, we observe that the failure occurs when the states of CSTR 2 are within the fault-recovery region. Therefore, assuming no delays in the fault-detection, computations and communication processing (that is, instantaneous switching), when the fallback controllers are activated, closed-loop stability is preserved and the closed-loop states converge close to the desired steady-state as shown by the solid lines in Figure 13.

When delay effects are taken into account, we see from Figure 14 (top plot) that if an overall delay of 3 min (accounting for delays in fault-detection, controller computations, information transmission and actuator activation) elapses between the failure and the activation of the (Q_2, C_{A03}) configuration—during this delay, CSTR 2 evolves in an open-loop mode as indicated by the dotted line in Figure 14 (top plot)—the state at the end of the delay still resides within the fault-recovery region and, therefore, closed-loop stability is

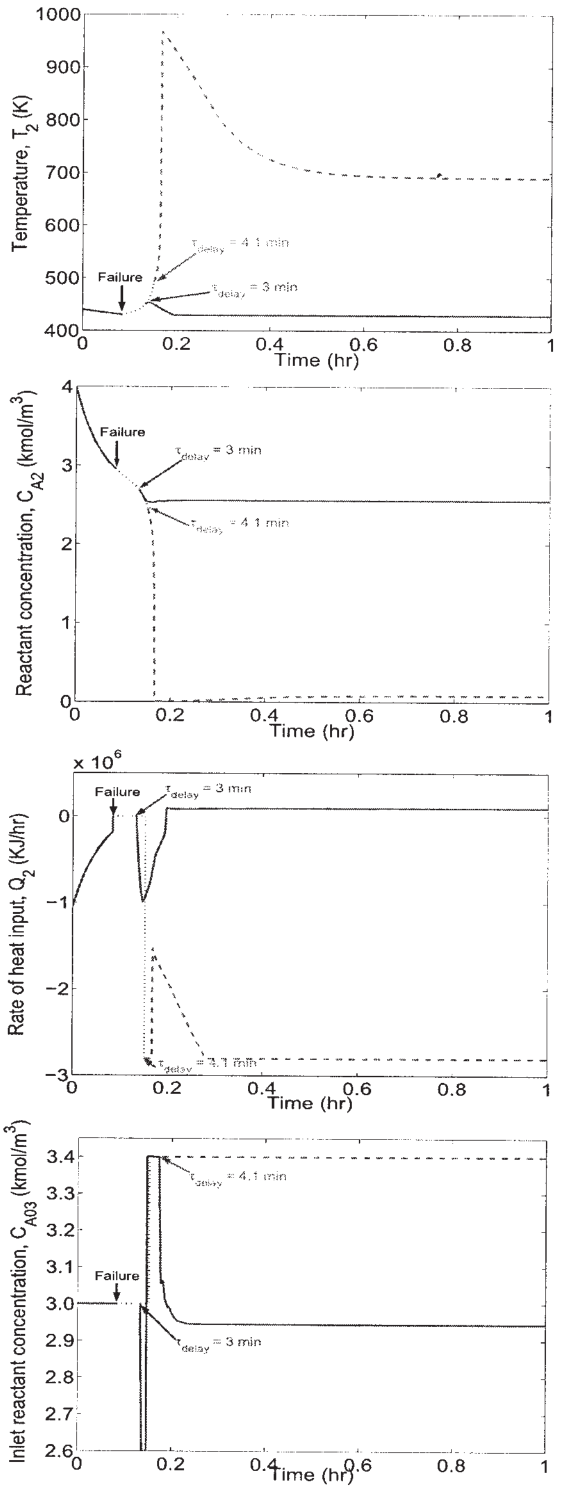

Figure 15. Evolution of the closed-loop state and input profiles when the failure occurs at T_f = 5 min, and the fallback configuration (Q_2, C_{A03}), with constraints $|Q_2| \leq 2.8 \times 10^6$ KJ/hr and $|C_{A03} - C^s_{A03}| \leq 0.4$ kmol/m^3 is activated after a total delay of 3 min (solid lines), and after a total delay of 4.1 min (dashed lines).

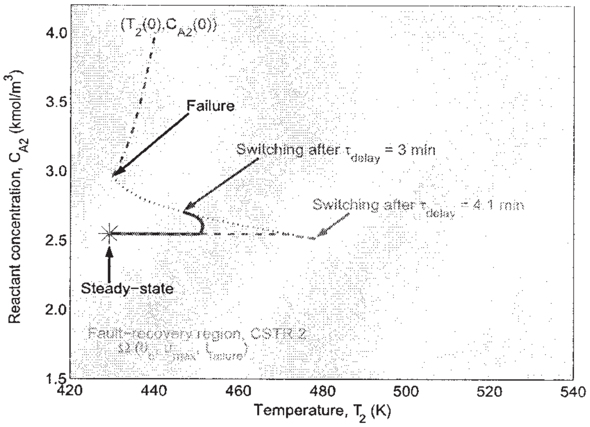

Figure 16. Fault-recovery region of the fallback control configuration (Q_2, C_{A03}) for CSTR 2, with constraints $|Q_2| \leq 1.4 \times 10^7$ KJ/hr and $|C_{A03} - C^s_{A03}| \leq 2.0$ kmol/m^3 when failure occurs at T_f = 5 min.

Activation of the fallback configuration after a delay of either 3 min or 4.1 min ensures fault-tolerance.

preserved by switching to the (Q_2, C_{A03}) configuration at the end of the delay. The corresponding state and input profiles are shown by the solid lines in Figures 14 and 15. By contrast, we see from the bottom plot in Figure 14 that when an overall delay of 4.1 min is considered, the state at the end of the delay lies outside the fault-recovery region; hence, the fallback configuration cannot stabilize the system at the desired steady-state, as can be seen from the dashed lines in Figures 14 and 15.

Examination of Figure 14 provides useful insights into how the tradeoff between the controller design, switching and communication logics can be managed to ensure fault-tolerance. For example, the picture suggests that with a larger fault-recovery region, even large delays maybe tolerated by switching to this particular configuration. A larger region can be obtained by relaxing the constraints. Figure 16 shows the resulting fault-recovery region for the (Q_2, C_{A03}) configuration when the constraints are relaxed to $|Q_2| \leq u^{Q_2}_{max} = 1.4 \times 10^7$ KJ/hr and $|C_{A03} - C_{A03s}| \leq u^{C_{A03}}_{max} = 2.0$ kmol/m^3. In this case, the fault-recovery region includes the entire area of the plot. As a result, activation of the fallback configuration, whether after 3 min or 4.1 min from the failure time, stabilizes the reactor since the state at the end of the delay in both cases is contained within the fault-recovery region. Figure 17 shows the corresponding closed-loop state and input profiles of CSTR 2 for both scenarios.

Conclusions

In this work, we presented a methodology for the design of fault-tolerant control systems for chemical plants with distributed interconnected processing units. Bringing together tools from Lyapunov-based nonlinear control and hybrid systems theory, the approach is based on a hierarchical architecture that integrates lower-level feedback control of the individual units with upper-level logic-based supervisory control over communication networks. The local control system for each unit consists of a family of control configurations for each of which a stabilizing feedback controller is designed, and the stability

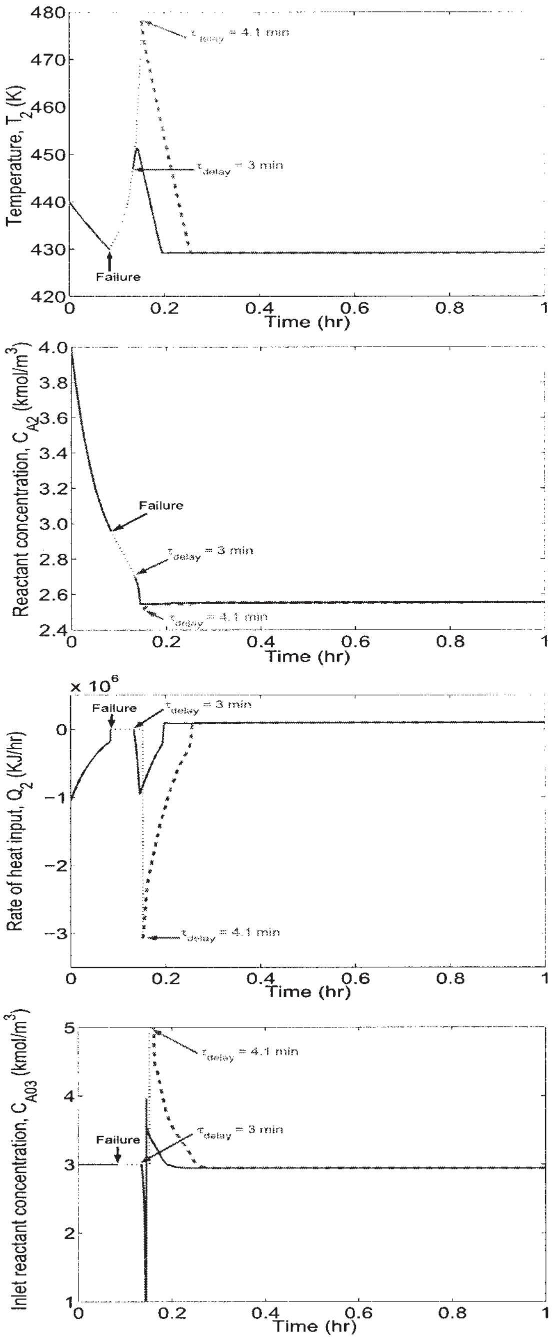

Figure 17. Evolution of the closed-loop state and manipulated input profiles when the failure occurs at T_f = 5 min and the fallback configuration (Q_2, C_{A03}), with constraints $|Q_2| \leq 1.4 \times 10^7$ KJ/hr and $|C_{A03} - C^s_{A03}| \leq 2.0$ kmol/m^3 is activated after a total delay of 3 min (solid lines), and after a total delay of 4.1 min (dashed lines).

region is explicitly characterized. The actuators and sensors of each configuration are connected, via a local communication network, to a local supervisor that orchestrates switching between the constituent configurations, on the basis of the stability regions, in the event of failures. The local supervisors communicate, through a plant-wide communication network, with a plant supervisor responsible for monitoring the different units and coordinating their responses in a way that minimizes the propagation of failure effects. The communication logic is designed to ensure efficient transmission of information between units while also respecting the inherent limitations in network resources by minimizing unnecessary network usage and accounting explicitly for the effects of possible delays due to fault-detection, control computations, network communication and actuator activation. Explicit guidelines for managing the various interplays between the coupled tasks of feedback control, fault-tolerance and communication were provided. The efficacy of the proposed approach was demonstrated through chemical process examples.

Acknowledgments

Financial support from NSF, CTS-0129571, is gratefully acknowledged.

Literature Cited

1. Willsky AS. A survey of design methods for failure detection in dynamic systems. *Automatica.* 1998;12:601–611.
2. Yang GH, Zhang SY, Lam J, Wang J. Reliable control using redundant controllers. *IEEE Trans Autom Contr.* 1988;43:1588–1593.
3. Bao J, Zhang WZ, Lee PL. Passivity-based decentralized failure-tolerant control. *Ind & Eng Chem Res.* 2002;41:5702–5715.
4. Patton RJ. Fault-tolerant control systems: the 1997 situation. *Proceedings of the IFAC Symposium SAFEPROCESS 1997.* Hull, U.K.; 1997: 1033–1054.
5. Blanke M, Izadi-Zamanabadi R, Bogh SA, Lunau CP. Fault-tolerant control systems—a holistic view. *Contr Eng Prac.* 1997;5:693–702.
6. Mahmoud M, Jiang J, Zhang Y. Active fault tolerant control systems: stochastic analysis and synthesis. In: Lecture Notes in Control and Information Sciences. vol. 287. Heidelberg, Germany: Springer-Verlag; 2003:1–187.
7. Kazantzis N, Kravaris C. Energy-predictive control: a new synthesis approach for nonlinear process control. *Chem Eng Sci.* 1999;54:1697–1709.
8. Kapoor N, Daoutidis P. Stabilization of nonlinear processes with input constraints. *Comp & Chem Eng.* 2000;24:9–21.
9. El-Farra NH, Christofides PD. Integrating robustness, optimality, and constraints in control of nonlinear processes. *Chem Eng Sci.* 2001;56: 1841–1868.
10. El-Farra NH, Christofides PD. Bounded robust control of constrained multivariable nonlinear processes. *Chem Eng Sci.* 2003;58:3025–3047.
11. Valluri S, Soroush M. A non-linear controller design method for processes with saturating actuators. *Inter J Contr* 2003;76:698–716.
12. El-Farra NH, Christofides PD. Switching and feedback laws for control of constrained switched nonlinear systems. In: Tomlin CJ, Greenstreet MR, eds. Lecture Notes in Computer Science. vol. 2289. Berlin, Germany: Springer-Verlag; 2002. 164–178.
13. El-Farra NH, Christofides PD. Coordinating feedback and switching for control of hybrid nonlinear processes. *AIChE J.* 2003;49:2079–2098.
14. El-Farra NH, Christofides PD. Coordinated feedback and switching for control of spatially-distributed processes. Comp & Chem Eng. 2004; 28:111–128.
15. El-Farra NH, Lou Y, Christofides PD. Fault-tolerant control of fluid dynamic systems: coordinated feedback and switching. *Comp & Chem Eng.* 2003;27:1913–1924.
16. Bemporad A, Morari M. Control of systems integrating logic, dynamics and constraints. *Automatica.* 1999;35:407–427.
17. El-Farra NH, Mhaskar P, Christofides PD. Hybrid predictive control of nonlinear systems: method and applications to chemical processes. *Inter J Rob & Non Contr.* 2004;14:199–225.
18. Yamalidou EC, Kantor J. Modeling and optimal control of discrete-event chemical processes using Petri nets. Comp & Chem Eng. 1990; 15:503–519.
19. Barton PI, Pantelides CC. Modeling of combined discrete/continuous processes. *AIChE J.* 1994;40:966–979.

20. Garcia-Onorio V, Ydstie BE. Distributed, asynchronous and hybrid simulation of process networks using recording controllers. *Inter J Rob & Non Contr.* 2004;14:227–248.
21. Harjunkoski I, Jain V, Grossmann IE. Hybrid mixed-integer/constrained logic programming strategies for solving scheduling and combinatorial optimization problems. *Comp & Chem Eng.* 2000;24:337–343.
22. Grossmann IE, van den Heever SA, Harjukoski I. Discrete optimization methods and their role in the integration of planning and scheduling. In: Proceedings of 6th International Conference on Chemical Process Control. Tucson, AZ; 2001. p. 124–152.
23. Hespanha JP, Morse AS. Stability of switched systems with average dwell time. In: Proceedings of 38th IEEE Conference on Decision and Control. Phoenix, AZ; 1999. p. 2655–2660.
24. Decarlo RA, Branicky MS, Petterson S, Lennartson B. Perspectives and results on the stability and stabilizability of hybrid systems. Proceedings of the IEEE. 2000;88:1069–1082.
25. Ydstie EB. New vistas for process control: integrating physics and communication networks. *AIChE J. 2002;48:422–426.*
26. Walsh GC. Ye H, Bushnell LG. Stability analysis of networked control systems. *IEEE Trans Contr Syst Tech.* 2002;10: 438–446.
27. Montestruque LA, Antsaklis PJ. On the model-based control of networked systems. *Automatica.* 2003;39:1837–1843.
28. Tipsuwan Y, Chow MY. Control methodologies in networked control systems. *Contr Eng Prac.* 2003;11:1099–1111.
29. Patankar R. A model for fault-tolerant networked control system using TTP/C communication. In: Proceedings of American Control Conference. Denver, CO; 2003. p. 533–537.
30. Xu Y, Hespanha J. Communication logics for networked control systems. In: Proceedings of American Control Conference. Boston, MA; 2004. p. 572–577.
31. Frank PM, Ding X. Survey of robust residual generation and evaluation methods in observer-based fault detection systems. *J Proc Contr.* 1997;7:403–424.
32. De Persis C, Isidori A. A geometric approach to nonlinear fault detection and isolation. *IEEE Trans Automat Contr.* 2001;46: 853–865.
33. Whiteley JR, Davis JF. Qualitative interpretation of sensor patterns. *IEEE Expert.* 1992;8:54–63.
34. Harris TJ, Boudreau F, MacGregor JF. Performance assessment of multivariable feedback controllers. *Automatica.* 1996;32:1505–1518.
35. Davis JF, Piovoso ML, Kosanovich K, Bakshi B. Process data analysis and interpretation. *Advances in Chemical Engineering.* 1999;25:1–103.
36. Nounou MN, Bakshi BR, Goel PK, Shen X. Bayesian principal component analysis. *J Chemometrics.* 2002;16:576–595.
37. Aradhye HB, Davis JF, Bakshi BR. ART-2 and multiscale ART-2 for on-line process fault detection—validation via industrial case studies and monte carlo simulation. *Annual Reviews in Control.* 2002;26:113–127.
38. Venkatasubramanian V, Rengaswamy R, Yin K, Kavuri SN. Review of process fault diagnosis—parts I, II, III. *Comp & Chem Eng.* 2003; 27:293–346.
39. Lin Y, Sontag Ed. A universal formula for stabilization with bounded controls. *Syst & Contr Lett.* 1991;16:393–397.
40. El-Farra NH, Mhaskar P, Christofides PD. Uniting bounded control and MPC for stabilization of constrained linear systems. *Automatica.* 2004;40:101–110.
41. Mhaskar P, El-Farra NH, Christofides PD. Hybrid predictive control of process systems. *AIChE J.* 2004;50:1242–1259.
42. Mhaskar P, El-Farra NH, Christofides PD. Robust hybrid predictive control of nonlinear systems. *Automatica.* 2005;41:209–217.
43. Freeman RA, Kokotovic PV. Robust nonlinear control design: state-space and lyapunov techniques. Boston: Birkhauser; 1996.
44. Sepulchre R, Jankovic M, Kokotovic P. Constructive nonlinear control. Berlin-Heidelberg: Springer-Verlag; 1997.
45. Dubljević S, Kazantzis N. A new Lyapunov design approach for nonlinear systems based on Zubov's method. *Automatica.* 2002;38: 1999–2007.
46. Brockett R. Minimum attention control. In: *Proceedings of 36th Conference on Decision and Control.* San Diego, CA; 1997;2628–2632.

Manuscript received May 15, 2004, and revision received Oct. 20, 2004.