

UNIVERSITY OF CALIFORNIA

Los Angeles

Integrated Fault Detection and Isolation and  
Fault-Tolerant Control of Nonlinear Process Systems

A dissertation submitted in partial satisfaction of the  
requirements for the degree Doctor of Philosophy  
in Chemical Engineering

by

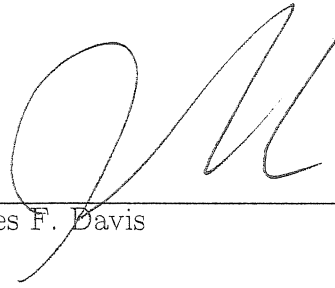
Charles W. McFall

2008

This page is intended blank and should not be use in the submission of the thesis.

It has the purpose of making the next page, the signature page, page ii.

The dissertation of Charles W. McFall is approved.



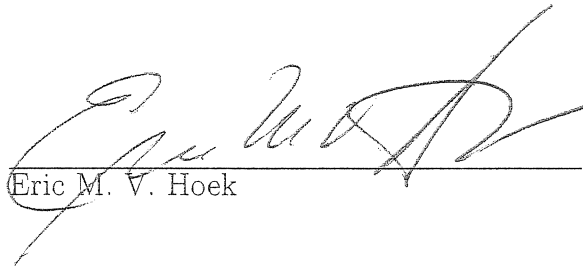
---

James F. Davis



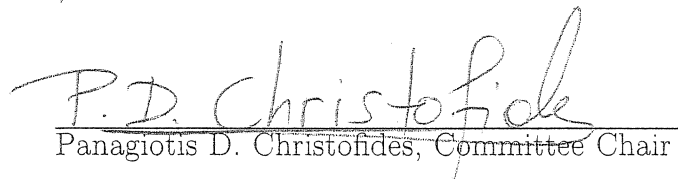
---

Gerassimos Orkoulas



---

Eric M. V. Hoek



---

Panagiotis D. Christofides, Committee Chair

University of California, Los Angeles

2008

# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Background on fault-detection and fault-tolerant control . . . . .	1
1.2	Dissertation objectives and structure . . . . .	5
<b>2</b>	<b>Integrated fault-detection and fault-tolerant control</b>	<b>10</b>
2.1	Introduction . . . . .	10
2.2	Preliminaries . . . . .	11
2.2.1	Process description . . . . .	11
2.2.2	Motivating example . . . . .	12
2.2.3	Bounded Lyapunov-based control . . . . .	15
2.3	State feedback case . . . . .	16
2.3.1	Simulation results . . . . .	23
2.4	Output feedback case . . . . .	29
2.4.1	Output feedback control . . . . .	30
2.4.2	Integrating fault-detection and fault-tolerant output feedback control . . . . .	33
2.4.3	Simulation results . . . . .	41

<b>3</b>	<b>Integrated fault-detection and isolation and fault-tolerant control</b>	<b>52</b>
3.1	Introduction . . . . .	52
3.2	Preliminaries . . . . .	53
3.3	State-feedback Fault-tolerant control . . . . .	55
3.3.1	State-feedback fault detection and isolation filter . . . . .	55
3.3.2	State-feedback fault-tolerant controller . . . . .	59
3.4	Output-feedback fault-tolerant control . . . . .	61
3.4.1	Output feedback controller . . . . .	61
3.4.2	Output-feedback fault detection and isolation filter . . . . .	64
3.4.3	Output-feedback fault detection and isolation and fault tolerant control . . . . .	66
3.5	Simulation examples . . . . .	69
<b>4</b>	<b>Fault-tolerant control of nonlinear process systems subject to sensor faults</b>	<b>86</b>
4.1	Introduction . . . . .	86
4.2	Preliminaries . . . . .	87
4.2.1	A chemical reactor example . . . . .	90
4.3	Stabilization subject to sensor failures . . . . .	92
4.3.1	Reconfiguration law . . . . .	92
4.3.2	Application to the chemical reactor . . . . .	96
4.4	Stabilization subject to sensor data losses . . . . .	97
4.4.1	Modeling sensor data loss . . . . .	98
4.4.2	Analyzing closed-loop stability . . . . .	100

4.4.3	Control of a chemical reactor subject to sensor data loss . . .	108
4.5	Fault-tolerant control subject to sensor data losses . . . . .	110
4.5.1	Reconfiguration law . . . . .	110
4.5.2	Fault-tolerant control of a chemical reactor . . . . .	113
4.5.3	Fault-tolerant control of a polyethylene reactor subject to sensor data loss . . . . .	114
4.6	Conclusions . . . . .	126
<b>5</b>	<b>Fault-tolerant control of a reverse osmosis desalination process</b>	<b>127</b>
5.1	Introduction . . . . .	127
5.2	Process description and modeling . . . . .	129
5.3	Fault-detection and isolation and fault-tolerant control . . . . .	131
5.3.1	Constrained feedback controller synthesis . . . . .	133
5.3.2	Characterization of stability regions . . . . .	134
5.3.3	Fault-detection and isolation filter design . . . . .	135
5.3.4	Fault-tolerant supervisory switching logic . . . . .	136
5.4	Simulation results . . . . .	137
5.5	Conclusions . . . . .	140
<b>6</b>	<b>Control and monitoring of a high recovery reverse osmosis desalination process</b>	<b>141</b>
6.1	Introduction . . . . .	141
6.2	Process description and modeling . . . . .	142
6.3	Reverse osmosis process model solution algorithm . . . . .	147
6.4	Feedback controller synthesis . . . . .	149

6.5	Fault detection and isolation and fault tolerant control . . . . .	155
6.5.1	Fall-back control configurations . . . . .	155
6.5.2	Fault detection and isolation filters . . . . .	156
6.5.3	Fault-tolerant supervisory switching logic . . . . .	157
6.6	Simulation results . . . . .	158
6.6.1	Large time varying disturbance . . . . .	158
6.6.2	Actuator failures . . . . .	172
6.7	Conclusions . . . . .	181
<b>7</b>	<b>Fault-Detection and Isolation for Nonlinear Process Systems Using Asynchronous Measurements</b>	<b>183</b>
7.1	Introduction . . . . .	183
7.2	FDI using asynchronous measurements: Problem formulation and so- lution . . . . .	184
7.2.1	Class of nonlinear systems . . . . .	184
7.2.2	Modeling of asynchronous measurements . . . . .	186
7.2.3	Asynchronous state observer . . . . .	187
7.2.4	Design of fault-detection and isolation filter . . . . .	188
7.3	Application to a polyethylene reactor with asynchronous measurements	193
7.3.1	Process and measurement modeling . . . . .	193
7.3.2	Design of the asynchronous state observers . . . . .	196
7.3.3	Design of the state feedback controller . . . . .	199
7.3.4	Design of FDI/FTC scheme . . . . .	200
7.3.5	Closed-loop process simulation results . . . . .	202

7.4	Conclusions . . . . .	213
<b>8</b>	<b>Conclusions</b>	<b>214</b>
	Bibliography . . . . .	219



# List of Figures

2.1	A schematic of the CSTR showing the three candidate control configurations. . . . .	14
2.2	Integrated fault-detection and fault-tolerant control design: state feedback case. . . . .	20
2.3	Evolution of the closed-loop state profiles under the switching rule of Eq.2.8 subject to failures in control systems 1 and 2 (solid line) and under arbitrary switching (dashed line). . . . .	25
2.4	Evolution of the closed-loop (a) temperature and (b) concentration under the switching rule of Eq.2.8 subject to failures in control systems 1 and 2 (solid lines) and under arbitrary switching (dashed lines). . .	26
2.5	Evolution of the closed-loop residual under the fault-detection filter for (a) control configuration 1 and (b) control configurations 2 and 3 under the switching rule of Eq.2.8 subject to failures in control systems 1 and 2 (solid lines) and under arbitrary switching (dashed lines). . . . .	27
2.6	Manipulated input profiles under (a) control configuration 1, (b) control configuration 2, and (c) control configuration 3 under the switching rule of Eq.2.8 subject to failures in control systems 1 and 2 (solid lines) and under arbitrary switching (dashed lines). . . . .	28

2.7	Integrated fault-detection and fault-tolerant control design under output feedback. . . . .	34
2.8	Evolution of the closed-loop (a) temperature (solid line), estimate of temperature (dash-dotted line) and the temperature profile generated by the filter (dashed line) and (b) concentration (solid line), estimate of concentration (dash-dotted line) and the concentration profile generated by the filter (dashed line) under control configuration 1 when the fault detection filter is initialized at $t = 0.005$ minutes. . . . .	44
2.9	Evolution of (a) the residual and (b) the manipulated input profile for the first control configuration when the fault detection filter is initialized at $t = 0.005$ minutes. . . . .	45
2.10	Evolution of the closed-loop (a) temperature (solid line), estimate of temperature (dash-dotted line) and the temperature profile generated by the filter (dashed line) and (b) concentration (solid line), estimate of concentration (dash-dotted line) and the concentration profile generated by the filter (dashed line) under the switching rule of Eq.2.22 subject to failures in control systems 1 and 2. . . . .	46
2.11	Evolution of the closed-loop state trajectory under the switching rule of Eq.2.22 subject to failures in control systems 1 and 2, using an appropriate fault-detection filter (solid line) and in the absence of a fault-detection filter (dashed line). . . . .	47
2.12	Evolution of the residual for (a) the first control configuration and (b) the second control configuration. . . . .	49

2.13	Evolution of the closed-loop (a) temperature (solid line), estimate of temperature (dash-dotted line) and the temperature profile generated by the filter (dashed line) and (b) concentration (solid line), estimate of concentration (dash-dotted line) and the concentration profile generated by the filter (dashed line) under the switching rule of Eq.2.22 subject to failures in control systems 1 and 2 in the absence of a fault-detection filter. . . . .	50
2.14	Manipulated input profiles under (a) control configuration 1, (b) control configuration 2, and (c) control configuration 3 under the switching rule of Eq.2.22 subject to failures in control systems 1 and 2 in the presence (solid lines) and absence (dashed lines) of a fault-detection filter. . . . .	51
3.1	A schematic of two CSTRs operating in series. . . . .	70
3.2	Evolution of reactor one closed-loop temperature profile under the switching rule of Theorem 3.3 (solid line) and in the absence of fault-tolerant control (dashed line) subject to simultaneous failures in both the heating jackets. . . . .	74
3.3	Evolution of reactor two closed-loop temperature profile under the switching rule of Theorem 3.3 (solid line) and in the absence of fault-tolerant control (dashed line) subject to simultaneous failures in both the heating jackets. . . . .	75
3.4	Evolution of reactor one closed-loop reactant concentration profile under the switching rule of Theorem 3.3 (solid line) and in the absence of fault-tolerant control (dashed line) subject to simultaneous failures in both the heating jackets. . . . .	75

3.5	Evolution of reactor two closed-loop reactant concentration profile under the switching rule of Theorem 3.3 (solid line) and in the absence of fault-tolerant control (dashed line) subject to simultaneous failures in both the heating jackets. . . . .	76
3.6	Evolution of residuals $e_{1,1}$ (solid line) and $e_{2,1}$ (dashed line) corresponding to the manipulated inputs in the first reactor. . . . .	76
3.7	Evolution of residuals $e_{3,1}$ (solid line) and $e_{4,1}$ (dashed line) corresponding to the manipulated inputs in the second reactor. . . . .	77
3.8	Evolution of the closed-loop temperature (solid line), estimate of temperature (dash-dotted line), and the temperature profile generated by the FDI filter (dashed line) with fault-tolerant control in place. Evolution of the temperature (dotted line) without fault-tolerant control in place. . . . .	79
3.9	Evolution of the residual corresponding to $Q_1$ for before switching ( $k = 1$ , solid line), and $Q_3$ after switching ( $k = 2$ , dashed line). A fault is declared when $e_{1,1}$ reaches the threshold at 0.1. . . . .	79
3.10	Evolution of the residual corresponding to $Q_2$ for before switching ( $k = 1$ , solid line), and after switching ( $k = 2$ , dashed line). No fault is declared. . . . .	80
3.11	(a) Temperature profile of reactor two with reconfiguration (solid line) and without reconfiguration (dotted line), (b) $Q_1$ residual profile, and (c) $Q_2$ residual profile (note fault detection at time $t = 40.79 \text{ min}$ ). . . . .	82
3.12	(a) Temperature profile of reactor two with reconfiguration (solid line) and without reconfiguration (dotted line), (b) $Q_1$ residual profile, and (c) $Q_2$ residual profile (note fault detection at time $t = 41.33 \text{ min}$ ). . . . .	84

4.1	A schematic of the CSTR showing the three candidate control configurations. . . . .	91
4.2	Evolution of the state profile under configuration 2 (dashed line) followed by loss of measurements (dotted line) and upon recovery reactivating configuration 2 (dash-dotted line), closed-loop stability is not preserved; however, switching to configuration 1 (solid line) preserves closed-loop stability. . . . .	96
4.3	Closed-loop system in the (a) absence, and (b) presence of sensor data losses. . . . .	99
4.4	Evolution of the state trajectory under control configuration 2 in the presence of sensor data loss (defined over a finite interval) at a rate of 0.4 (dashed line), sensor data loss (defined over an infinite interval) at a rate of 0.05 (dash-dotted line) and sensor data loss (defined over a finite interval) at a rate of 0.1 (solid line). . . . .	109
4.5	Manipulated input profile under control configuration 2 in the presence of sensor data loss (defined over a finite interval) at a rate of 0.4 (dashed line), sensor data loss (defined over an infinite interval) at a rate of 0.05 (dash-dotted line) and sensor data loss (defined over a finite interval) at a rate of 0.1 (solid line). . . . .	109
4.6	Evolution of the state trajectory: At $t = 13.5$ minutes the data loss rate goes up to 0.35 under configuration 2 (solid line). Keeping with configuration 2 (dotted line) or switching to configuration 3 (dashed line) does not preserve stability, while switching to configuration 1 (dash-dotted line) preserves stability. . . . .	114

4.7	Manipulate input profiles: At $t = 13.5$ minutes the data loss rate goes up to 0.35 under configuration 2 (solid line), switching to configuration 3 does not preserve stability (dashed line), while switching to configuration 1 (dash-dotted line) preserves stability. . . . .	115
4.8	Industrial gas phase polyethylene reactor system. . . . .	116
4.9	Evolution of the closed-loop state profiles under primary control configuration under continuous measurements (solid lines) and sensor data loss rate of 0.75 (dotted lines). . . . .	120
4.10	Evolution of the manipulated input profiles under primary control configuration under continuous measurements. . . . .	121
4.11	Evolution of the manipulated input profiles under primary control configuration with sensor data loss rate of 0.75. . . . .	121
4.12	Evolution of the closed-loop state profiles under the primary configuration with the data loss rate increasing from 0.75 to 0.80 at 0.97 hours. . . . .	122
4.13	Evolution of the closed-loop state profiles under the reconfiguration law of Eq.4.12 with the data loss rate increasing from 0.75 to 0.80 at 0.97 hours. . . . .	124
4.14	Evolution of the closed-loop input profiles under the reconfiguration law of Eq.4.12 with the data loss rate increasing from 0.75 to 0.80 at 0.97 hours. . . . .	125
5.1	Single membrane unit reverse osmosis desalination process. . . . .	129

5.2	Evolution of the closed-loop state profiles under fault-tolerant control (dashed line) and without fault tolerant-control (solid line). FTC recovers the desired brine flow, $v_3$ . . . . .	139
5.3	Evolution of the closed-loop pressure profile under fault tolerant control (dashed line) and without fault tolerant control (solid line). FTC recovers the desired operating pressure. . . . .	140
6.1	Single membrane unit high recovery reverse osmosis desalination process. The two actuated valves, retentate valve and bypass valve, act as manipulated inputs. . . . .	143
6.2	An expanded view of a spiral wound membrane module and typical concentration and velocity profiles inside the module. . . . .	144
6.3	Disturbance on feed concentration versus time, this large time-varying disturbance on the RO system is added to the nominal $C_f$ value. . . .	149
6.4	Bypass velocity, $v_b$ , profiles versus time; Open-loop (solid line), closed-loop feedback control without disturbance measurements (dotted line). . . . .	150
6.5	Retentate velocity, $v_r$ , profiles versus time; Open-loop (solid line), closed-loop feedback control without disturbance measurements (dotted line). The dotted line nearly overlaps the solid line. . . . .	150
6.6	Internal pressure, $P$ , profiles versus time; Open-loop (solid line), closed-loop feedback control without disturbance measurements (dotted line). The dotted line nearly overlaps the solid line. . . . .	151
6.7	Product velocity, $v_p$ , profiles versus time; Open-loop (solid line), closed-loop feedback control without disturbance measurements (dotted line). . . . .	151

6.8	Bypass velocity, $v_b$ , profiles versus time; Open-loop (solid line) and PI control with $P$ and $v_r$ as controlled outputs (dashed line). . . . .	159
6.9	Retentate velocity, $v_r$ , profiles versus time; Open-loop (solid line) and PI control with $P$ and $v_r$ as controlled outputs (dashed line). . . . .	160
6.10	Internal pressure, $P$ , profiles versus time; Open-loop (solid line) and PI control with $P$ and $v_r$ as controlled outputs (dashed line). . . . .	160
6.11	Product velocity, $v_p$ , profiles versus time; Open-loop (solid line) and PI control with $P$ and $v_r$ as controlled outputs (dashed line). . . . .	161
6.12	Manipulated inputs for the PI controller with $P$ and $v_r$ as the controlled outputs. Control action applied to $e_{v1}$ and $e_{v2}$ are the solid and dashed lines, respectively. . . . .	161
6.13	Manipulated inputs for the Lyapunov-based feedback controller with no feed-forward compensation with $v_b$ and $v_r$ as the controlled outputs. Control actions applied to $e_{v1}$ and $e_{v2}$ are the solid and dashed lines, respectively. . . . .	163
6.14	Bypass velocity, $v_b$ , profiles versus time; Open-loop (solid line) and under feed-forward/feedback control with $v_b$ and $v_r$ as controlled outputs (dashed line). . . . .	163
6.15	Retentate velocity, $v_r$ , profiles versus time; Open-loop (solid line) and under feed-forward/feedback control with $v_b$ and $v_r$ as controlled outputs (dashed line). . . . .	164
6.16	Internal pressure, $P$ , profiles versus time; Open-loop (solid line) and under feed-forward/feedback control with $v_b$ and $v_r$ as controlled outputs (dashed line). . . . .	164



6.17	Product velocity, $v_p$ , profiles versus time; Open-loop (solid line) and under feed-forward/feedback control with $v_b$ and $v_r$ as controlled outputs (dashed line). . . . .	165
6.18	Manipulated inputs for the feed-forward/feedback controller with $v_b$ and $v_r$ as the controlled outputs. Control actions applied to $e_{v1}$ and $e_{v2}$ are the solid and dashed lines, respectively. . . . .	166
6.19	Bypass velocity, $v_b$ , profiles versus time; Open-loop (solid line) and under feed-forward/feedback control with $P$ and $v_r$ as controlled outputs (dash-dotted line). . . . .	167
6.20	Retentate velocity, $v_r$ , profiles versus time; Open-loop (solid line) and under feed-forward/feedback control with $P$ and $v_r$ as controlled outputs (dash-dotted line). . . . .	168
6.21	Internal pressure, $P$ , profiles versus time; Open-loop (solid line) and under feed-forward/feedback control with $P$ and $v_r$ as controlled outputs (dash-dotted line). . . . .	168
6.22	Product velocity, $v_p$ , profiles versus time; Open-loop (solid line) and under feed-forward/feedback control with $P$ and $v_r$ as controlled outputs (dash-dotted line). . . . .	169
6.23	Manipulated inputs for the feed-forward/feedback controller with $P$ and $v_r$ as the controlled outputs. Control actions applied to $e_{v1}$ and $e_{v2}$ are the solid and dashed lines, respectively. . . . .	171
6.24	Retentate velocity, $v_r$ , profile versus time; subject to a failure in $e_{v2}$ (solid line) and with FDIFTC recovery (dotted line). . . . .	173
6.25	System pressure, $P$ , profile versus time; subject to a failure in $e_{v2}$ (solid line) and with FDIFTC recovery (dotted line). . . . .	174

6.26	Residual corresponding to the bypass valve versus time. No fault is detected on the bypass valve. The solid line is under the primary control configuration, and the dotted line is under the fall-back configuration.	174
6.27	Residual corresponding to the retentate valve versus time. A fault is detected in this valve at $t = 592 \text{ min}$ . The solid line is under the primary control configuration, and the dotted line is under the fall-back configuration. . . . .	175
6.28	Retentate velocity, $v_r$ , profile versus time; subject to a failure in $e_{v1}$ (solid line) and with FDIFTC recovery (dotted line). . . . .	176
6.29	System pressure, $P$ , profile versus time; subject to a failure in $e_{v1}$ (solid line) and with FDIFTC recovery (dotted line). . . . .	177
6.30	Residual corresponding to the bypass valve versus time. A fault is detected in this valve at $t = 592 \text{ min}$ . The solid line is under the primary control configuration, and the dotted line is under the fall-back configuration. . . . .	177
6.31	Residual corresponding to the retentate valve versus time. No fault is detected on the retentate valve. The solid line is under the primary control configuration, and the dotted line is under the fall-back configuration. . . . .	178
6.32	Retentate velocity, $v_r$ , profile versus time; subject to a failure in $e_{v1}$ . . . . .	179
6.33	System pressure, $P$ , profile versus time; subject to a failure in $e_{v1}$ . . . . .	179
6.34	Residual corresponding to the bypass valve versus time. The residual is exceeded at $t = 592 \text{ min}$ . . . . .	180

6.35	Residual corresponding to the retentate valve versus time. The residual is exceeded at $t = 592 \text{ min}$ . . . . .	180
7.1	Asynchronous sampling times $t_{k,[I_n]}$ (star), $t_{k,[M_1]}$ (cross), and $t_{k,Y}$ (circle) with a fault $d_1$ at $t = 0.5 \text{ hr}$ . . . . .	204
7.2	State trajectories of the closed-loop system without fault-tolerant control (circle/solid) and with appropriate fault-detection and isolation and fault-tolerant control where the fall-back control configuration is activated (star/dotted) with a fault $d_1$ at $t = 0.5 \text{ hr}$ . . . . .	205
7.3	Fault-detection and isolation residuals for the closed-loop system with a fault $d_1$ at $t = 0.5 \text{ hr}$ . The fault is detected immediately, but isolation occurs at $t = 0.59 \text{ hr}$ when all three asynchronous states have reported a residual below their detection threshold. This signals a type one fault, and we can isolate the source of this fault as $d_1$ . . . . .	205
7.4	Manipulated input for the closed-loop system without fault-tolerant control (solid) and with appropriate fault-tolerant control where the fall-back control configuration is activated (dotted) with a fault $d_1$ at $t = 0.5 \text{ hr}$ . . . . .	206
7.5	Asynchronous sampling times $t_{k,[I_n]}$ (star), $t_{k,[M_1]}$ (cross), and $t_{k,Y}$ (circle) with a fault $d_2$ at $t = 0.5 \text{ hr}$ . . . . .	207
7.6	State trajectories of the closed-loop system with a fault $d_2$ at $t = 0.5 \text{ hr}$ . . . . .	207

7.7 Fault-detection and isolation residuals for the closed-loop system with a fault  $d_2$  at  $t = 0.5$  hr. The fault is detected when residual for  $Y$  exceeds the threshold. Subsequently,  $T$  and  $[M_1]$  exceed their thresholds. When any asynchronous residual violates the threshold this indicates that the fault is in the set of type two faults;  $d_2$  or  $d_4$ . . . . . 208

7.8 Manipulated input for the closed-loop system with a fault  $d_2$  at  $t = 0.5$  hr. . . . . 208

7.9 Asynchronous sampling times  $t_{k,[In]}$  (star),  $t_{k,[M_1]}$  (cross), and  $t_{k,Y}$  (circle) with a fault  $d_3$  at  $t = 0.5$  hr. . . . . 209

7.10 State trajectories of the closed-loop system with a fault  $d_3$  at  $t = 0.5$  hr. 210

7.11 Fault-detection and isolation residuals for the closed-loop system with a fault  $d_3$  at  $t = 0.5$  hr. A fault is detected immediately when residual for  $T_{g1}$  exceeds the threshold. Subsequently, none of the asynchronous residuals exceed their thresholds, indicating that the fault source can be isolated as  $d_3$ . . . . . 210

7.12 Manipulated input for the closed-loop system with a fault  $d_3$  at  $t = 0.5$  hr. . . . . 211

7.13 Asynchronous sampling times  $t_{k,[In]}$  (star),  $t_{k,[M_1]}$  (cross), and  $t_{k,Y}$  (circle) with a fault  $d_4$  at  $t = 0.5$  hr. . . . . 211

7.14 State trajectories of the closed-loop system with a fault  $d_4$  at  $t = 0.5$  hr. 212

7.15 Fault-detection and isolation residuals for the closed-loop system with a fault  $d_4$  at  $t = 0.5$  hr. The fault is detected when residual for  $[M_1]$  exceeds the threshold. Subsequently,  $T$  and  $[In]$  exceed their thresholds. When any asynchronous residual violates the threshold this indicates the fault is in the set of type two faults;  $d_2$  or  $d_4$ . . . . . 212

7.16 Manipulated input for the closed-loop system with a fault $d_4$ at $t =$ 0.5 <i>hr</i> . . . . .	213
--	-----

# List of Tables

3.1	Process parameters and steady-state values for the chemical reactors of Eq.3.18. . . . .	71
5.1	Process parameters and steady-state values . . . . .	132
6.1	Process parameters and steady-state values . . . . .	152
7.1	Polyethylene reactor example process variables. . . . .	197
7.2	Polyethylene reactor noise parameters . . . . .	198

## VITA

- 2004 Bachelor of Science, Chemical Engineering  
Washington State University, WA
- 2006 Master of Science, Chemical Engineering  
University of California, Los Angeles, CA
- 2008 Doctor of Philosophy, Chemical Engineering  
University of California, Los Angeles, CA

## PUBLICATIONS AND PRESENTATIONS

1. Bartman, A., C. McFall, P. D. Christofides and Y. Cohen, "Model Predictive Control of Feed Flow Reversal in a Reverse Osmosis (RO) Membrane Desalination Process," California-Nevada AWWA Spring Conference, Hollywood, California, 2008.
2. Bartman, A., C. McFall, P. D. Christofides and Y. Cohen, "Model Predictive Control of Feed Flow Reversal in a Reverse Osmosis Desalination Process," AIChE Annual Meeting, to be presented, Philadelphia, Pennsylvania, 2008.
3. Bartman, A., C. McFall, P. D. Christofides and Y. Cohen, "Model Predictive Control of Feed Flow Reversal in a Reverse Osmosis Desalination Process," *J. Proc. Contr.*, in press.
4. Gani, A., P. Mhaskar, C. McFall, P. D. Christofides and J. F. Davis, "Fault-Tolerant Process Control: Handling Asynchronous Sensor Behavior," AIChE Annual Meeting, paper 494e, San Francisco, California, 2006.

5. McFall, C., A. Bartman, P. D. Christofides and Y. Cohen, "Control and Monitoring of a High-Recovery Reverse-Osmosis Desalination Process," *Ind. & Eng. Chem. Res.*, **47**, 13 pages, 2008.
6. McFall, C., A. Bartman P. D. Christofides and Y. Cohen, "Control of Reverse Osmosis Desalination at High Recovery," *Proceedings of the American Control Conference*, 2241-2247, Seattle, Washington, 2008, (**Best Presentation in Session Award for Charles McFall**).
7. McFall, C., A. Gani, P. Mhaskar, P. D. Christofides and J. F. Davis, "Fault-tolerant Control of Nonlinear Process Systems: Handling Sensor Malfunctions," AICHE Annual Meeting, paper 374f, Cincinnati, Ohio, 2005.
8. McFall, C., A. Gani, P. Mhaskar, P. D. Christofides and J. F. Davis, "Fault-Tolerant Output Feedback Control of Multivariable Nonlinear Processes," AICHE Annual Meeting, paper 654c, San Francisco, California, 2006.
9. McFall, C., A. Gani, P. Mhaskar, P. D. Christofides and J. F. Davis, "Fault-tolerant Process Control: Nonlinear FDI and Reconfiguration," AICHE Annual Meeting, paper 553g, Cincinnati, Ohio, 2005.
10. McFall, C., D. Muñoz de la Peña, B. Ohran, P. D. Christofides and J. F. Davis, "Fault-Detection and Isolation and Fault-Tolerant Control of Nonlinear Process Systems Using Asynchronous Measurements," AICHE Annual Meeting, to be presented, Philadelphia, Pennsylvania, 2008.
11. McFall, C., D. Muñoz de la Peña, B. Ohran, P. D. Christofides and J. F. Davis, "Fault Detection and Isolation and Fault-Tolerant Control of Nonlinear Process Systems Using Asynchronous Measurements," *Ind. & Eng. Chem. Res.*, submitted.



12. McFall, C., P. D. Christofides and J. F. Davis, "Integrated Fault Detection and Isolation and Fault-Tolerant Control of Process Systems," 15th Southern California Nonlinear Control Workshop, University of Southern California, January 2008.
13. McFall, C., P. D. Christofides, Y. Cohen and J. F. Davis, "Fault Detection and Control of a Reverse Osmosis Desalination Process," AIChE Annual Meeting, paper 444c, Salt Lake City, Utah, 2007.
14. McFall, C., P. D. Christofides, Y. Cohen and J. F. Davis, "Fault-Tolerant Control of a Reverse-Osmosis Desalination Process," *Proceedings of 8th IFAC Symposium on Dynamics and Control of Process Systems - Volume 3*, 163-168, Cancun, Mexico, 2007.
15. Mhaskar, P., A. Gani, N. H. El-Farra, C. McFall, P. D. Christofides and J. F. Davis, "Integrated Fault Detection and Fault-Tolerant Control of Process Systems," *AIChE J.*, **52**, 2129-2148, 2006.
16. Mhaskar, P., A. Gani, C. McFall, P. D. Christofides and J. F. Davis, "Fault-Tolerant Control of Nonlinear Process Systems Subject to Sensor Faults," *AIChE J.*, **53**, 654-668, 2007.
17. Mhaskar, P., A. Gani, C. McFall, P. D. Christofides and J. F. Davis, "Fault-Tolerant Control of Nonlinear Systems Subject to Sensor Data Losses," *Proceedings of 45th IEEE Conference on Decision and Control*, 3498-3505, San Diego, California, 2006.
18. Mhaskar, P., C. McFall, A. Gani, P. D. Christofides and J. F. Davis, "Fault-tolerant Control of Nonlinear Systems: Fault Detection and Isolation and Controller Reconfiguration," *Proceedings of the American Control Conference*, 5115-

5122, Minneapolis, Minnesota, 2006 (**Best Presentation in Session Award for Prashant Mhaskar**).

19. Mhaskar, P., C. McFall, A. Gani, P. D. Christofides and J. F. Davis, "Isolation and Handling of Actuator Faults in Nonlinear Systems," *Automatica*, **44**, 53-62, 2008.
20. Ohran, B., A. Gani, P. Mhaskar, C. McFall, P. D. Christofides and J. F. Davis, "Uniting Data - and Model-Based Fault-Detection Filters for Fault-Tolerant Control of Process Systems," AIChE Annual Meeting, paper 125e, San Francisco, California, 2006.

## ABSTRACT OF THE DISSERTATION

Integrated Fault Detection and Isolation and  
Fault-Tolerant Control of Nonlinear Process Systems

by

Charles W. McFall

Doctor of Philosophy in Chemical Engineering

University of California, Los Angeles, 2008

Professor Panagiotis D. Christofides, Chair

Chemical process operations rely extensively on highly automated control systems in order to deal with increasingly stringent requirements of safety, environmental sustainability, and profitability. Automation, however, adds a layer of complexity to a chemical process that may lead to additional faults (e.g., failures in the actuators, sensors or in the controllers) potentially causing a host of safety, environmental and economic problems. Management of abnormal situations, such as automation faults, is a major challenge in the chemical and process industries since abnormal situations account annually for at least 10 billion USD in lost revenue in the US alone. Despite the major industrial importance of the problems of detecting, isolating and handling process/control system faults in a unified and efficient manner, these problems have been traditionally addressed in isolation, thereby significantly limiting the range of practical applicability and performance of the available solutions.

This work develops a general and practical framework for the design of automated

fault-tolerant control systems that seamlessly integrate the tasks of fault-detection and isolation and control system reconfiguration for fault handling. Working with general nonlinear dynamic models of chemical processes, we design nonlinear dynamic filters that allow for timely detection and isolation of actuator/control system faults using limited plant measurements. The key idea is to design a fault-detection and isolation scheme for nonlinear process systems that decouples the effect of a fault on all process variables except one. This allows fault detection and isolation for nonlinear chemical processes even with highly coupled variables. The nonlinear dynamic filters are coupled with suitable control system reconfiguration strategies which achieve quick fault recovery and guarantee closed-loop system stability. In addition, fault-tolerant control methods are developed to deal explicitly with the practical issues of limited control actuator capacity, model uncertainty and disturbances, measurement noise and sensor faults. We present applications of the proposed fault-tolerant control system design framework to: a) a chemical plant consisting of two reactors in series, b) a high recovery reverse osmosis desalination plant, and c) a gas-phase polyethylene reactor.

# Chapter 1

## Introduction

### 1.1 Background on fault-detection and fault-tolerant control

Modern-day chemical plants involve a complex arrangement of processing units connected, in series and/or in parallel, and highly integrated with respect to material and energy flows through recycle streams and to information flows through tightly interacting control systems. Increasingly faced with the requirements of safety, reliability and profitability, chemical plant operation is relying extensively on highly automated process control systems. Automation, however, adds a layer of complexity to a chemical process that may lead to additional faults (e.g., defects/malfunctions in process equipment, sensors and actuators, failures in the controllers or in the control loops) potentially causing a host of economic, environmental, and safety problems that can seriously degrade the operating efficiency of the plant if not addressed within a time appropriate to the context of the process dynamics. Examples include physical damage to the plant equipment, increase in the wasteful use of raw material and energy resources, increase in the downtime for process operation resulting in significant production losses, and jeopardizing personnel and environmental safety. Management

of abnormal situations is a major challenge in the chemical industry since abnormal situations account annually for at least \$10 billion in lost revenue in the US alone [77]. These considerations provide a strong motivation for the development of methods and strategies for the design of advanced fault-tolerant control structures that ensure an efficient and timely response to enhance fault recovery, prevent faults from propagating or developing into total failures, and reduce the risk of safety hazards. Providing responsive actions comparable to those of an experienced human operator freeing the operator to do more strategic process analysis. Given the geographically-distributed, interconnected nature of the plant units and the large number of distributed sensors and actuators typically involved [99], the success of a fault-tolerant control method requires efficient fault detection, control designs that account for the complex nonlinear dynamics and constraints, and a high-level supervisor that coordinates the overall plant response to achieve fault-tolerant control.

Fault-tolerant control has been an active area of research for the past ten years, and has motivated many research studies in the context of aerospace engineering applications (see, e.g., [80, 101]) that are based on the underlying assumption of the availability of more control configurations than is required. Under this assumption, the reliable control approach dictates use of all the control loops at the same time so that failure of one control loop does not lead to the failure of the entire control structure (e.g., [96]). Using only as many control loops as is required at a time, is often motivated by economic considerations (to save on unnecessary control action), and in this case, fault-tolerant control can be achieved through control-loop reconfiguration. Recently, fault-tolerant control has gained increasing attention in the context of chemical process control; however, the available results are mostly based on the assumption of a linear process description (e.g., [6, 95]), and do not account for

complexities such as control constraints or the unavailability of state measurements.

In process control, given the complex dynamics of chemical processes (e.g., nonlinearities, uncertainties and constraints) the success of any fault-tolerant control method 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 quick detection of process faults, and (3) the design of supervisory switching schemes that orchestrate the transition from the failed control configuration to available well-functioning fall-back configurations to ensure fault-tolerance. 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, advances in the analysis and control of hybrid process systems and advances in fault detection. In the remainder of this section, we will briefly review the state-of-the-art in these areas, as pertinent to the focus of this dissertation.

The highly nonlinear behavior of many chemical processes has motivated extensive research on nonlinear process control. Chemical process nonlinearities can arise from the first principles process model, bounds on manipulated inputs, controller elements, or complex process interactions. Excellent reviews of results in the area of nonlinear process control can be found, for example, in [7, 98, 43]; for a more recent review, see [13]. The problems caused by input constraints have motivated numerous studies on the dynamics and control of systems subject to input constraints. Important contributions in this area include results on optimization-based control methods such as model predictive control (e.g., [36, 59, 30]), Lyapunov-based control (e.g., [54, 91, 46, 51, 24, 25]) and hybrid predictive control (e.g., [28, 64]).

The occurrence of faults in chemical processes and subsequent switching to fall-

back control configurations naturally leads to the superposition of discrete events on the underlying continuous process dynamics thereby making a hybrid systems framework a natural setting for the analysis and design of fault-tolerant control structures. Proper coordination of the switching between multiple (or redundant) actuator/sensor configurations provides a means for fault-tolerant control. However, at this stage, despite the large and growing body of research work on a diverse array of hybrid system problems (e.g., [40, 38, 45, 17, 26]), the use of a hybrid system framework for the study of fault-tolerant control problems for nonlinear systems subject to constraints has received limited attention. In a previous work [27], a hybrid systems approach to fault-tolerant control was employed where, under the assumption of full state measurements and knowledge of the fault, stability region-based reconfiguration is implemented to achieve fault-tolerant control.

Existing results on the design of fault-detection filters include those that use past plant-data and those that use fundamental process models for the purpose of fault-detection filter design. Statistical and pattern recognition techniques for data analysis and interpretation (e.g., [52, 84, 78, 23, 74, 22, 16, 90, 3, 100]) use past plant-data to construct indicators that identify deviations from normal operation to detect faults. The problem of using fundamental process models for the purpose of detecting faults has been studied extensively in the context of linear systems [58, 31, 32, 61]; and more recently, some results in the context of nonlinear systems have been derived [86, 18].

Close examination of the existing literature indicates a lack of general and practical methods for the design of integrated fault-detection and fault-tolerant control structures for chemical plants accounting explicitly for actuator/controller failures, process nonlinearities and input constraints.



## 1.2 Dissertation objectives and structure

The objective of this dissertation is to develop a general and practical framework for the design of automated nonlinear fault-tolerant control systems that seamlessly integrate the tasks of fault-detection and isolation and control system reconfiguration for fault handling. The basic idea is that of detecting faults and orchestrating logic-based switching between multiple constrained control configurations, each characterized by different manipulated inputs and a different region of closed-loop stability. The switching policy, which is based on the information provided by the fault-detection filter and stability regions, is implemented by a supervisor that receives and transmits information to the feedback system and activates/deactivates the appropriate control configurations in a way that ensures fault-tolerance. The design of fault-detection filters for single-input systems is shown for both the state-feedback and output-feedback cases. The design of fault-detection and isolation filters for multi-input systems is also shown for both state-feedback and output-feedback cases. The design of fault detection and isolation filters is a key contribution of this work. Finally, the implementation of the proposed approaches are demonstrated through several multi-unit chemical process examples.

The rest of the dissertation is organized as follows: Chapter 2 considers the problem of implementing fault-tolerant control to single input nonlinear processes with input constraints subject to control actuator failures is considered in chapter two. An approach predicated upon the idea of integrating fault-detection, feedback and supervisory control is presented and demonstrated. For the processes under consideration, a family of candidate control configurations, characterized by different manipulated inputs, is first identified. For each control configuration, a Lyapunov-based controller that enforces asymptotic closed-loop stability in the presence of constraints, is

designed, and the constrained stability region, associated with it, is explicitly characterized. A fault-detection filter is used to compute the expected closed-loop behavior in the absence of faults. Deviations of the process states from the expected closed-loop behavior are used to detect faults. A switching policy is then derived, on the basis of the stability regions, to orchestrate the activation/deactivation of the constituent control configurations in a way that guarantees closed-loop stability in the event that a failure is detected. Simulation studies are presented to demonstrate the implementation and evaluate the effectiveness of the proposed fault-tolerant control scheme.

Chapter 3 considers integrated fault-detection and isolation and fault tolerant control (FDI/FTC) for single-input single-output nonlinear process systems with input constraints subject to control actuator failures. An approach predicated upon the idea of integrating fault-detection, feedback and supervisory control is presented and demonstrated in the context of chemical process simulation studies with state feedback and output feedback. Chapter 3 proposes a general framework of FDI/FTC for multi-input multi-output nonlinear process systems with input constraints subject to control actuator failures. The integrated FDI/FTC approach is demonstrated through multi-unit chemical process simulations with state feedback and output feedback.

Chapter 4 considers the problem of nonlinear process control subject to input constraints and sensor faults (complete failure or intermittent unavailability of measurements). A fault-tolerant controller is designed that utilizes reconfiguration (switching to an alternate control configuration) in a way that accounts for the process non-linearity, input constraints and the occurrence of sensor faults. To clearly illustrate the importance of accounting for input constraints, we first consider the problem of sensor faults that necessitate sensor recovery to maintain closed-loop stability. We

address the problem of determining which control configuration should be activated (reactivating the primary control configuration may not preserve stability) after the sensor is rectified based on stability region characterizations for the candidate control configurations. We then consider the problem of asynchronous measurements, i.e., of intermittent unavailability of measurements. To address this problem, the stability region (that is, the set of initial conditions starting from where closed-loop stabilization under continuous availability of measurements is guaranteed) as well as the maximum allowable data loss rate are calculated to preserve closed-loop stability for the primary and the candidate backup configurations. This characterization is utilized in identifying the occurrence of a destabilizing sensor fault and in activating a suitable backup configuration that preserves closed-loop stability. The proposed method is illustrated using a chemical process example and demonstrated via application to a polyethylene reactor.

Chapters 5 and 6 develop model-based nonlinear control structures for reverse osmosis desalination systems while accounting for such practical issues as sampled and noisy measurements, large time-varying disturbances, and actuator faults. To accomplish this a detailed mathematical model of a high-recovery RO plant is first developed. The dynamic reverse osmosis process model developed for the purpose of process control is a key contribution of this work. This model adequately describes the evolution of process states in time, and it also accounts for the spatial variation of total dissolved solids (TDS) and flow-rate inside the membrane units. Under high recovery operation the gradients along the length of the membrane unit can be quite significant. As fluid flows axially along the module the bulk concentration increases, the flow rate decreases, and the local permeate flux decreases [20]. The model developed in the present work includes appropriate differential equations in space that

account for these gradients. A Lyapunov-based nonlinear controller is then applied to this high recovery RO model. One of the main objectives of a controller in high recovery RO is to reject disturbances caused by feed water variation. Feed disturbances could cause undesired fluctuations in the product flow rate or the internal pressure. To accomplish disturbance rejection, the control law includes both feedback and feed-forward components (i.e., measurement of feed concentration fluctuations). The feed water stream concentration can easily be measured in practice, so the first set of simulations presented in this work explore the ability of the proposed control method to reject such disturbances. Another objective is to detect and isolate actuator faults as soon as possible. Additionally, illustrative examples demonstrate how fault-detection and isolation (FDI) and fault tolerant control (FTC) can be applied to reverse osmosis systems, and how appropriate action can be taken to maintain desired system operation when a fault occurs in the control system.

Chapter 7 addresses the problem of fault-detection and isolation and fault-tolerant control when some process variable measurements are available at regular sampling intervals, and the remaining states are measured at an infrequent asynchronous rate. First, a fault-detection and isolation (FDI) scheme that employs model-based techniques is proposed that allows for the isolation of faults. The proposed FDI scheme provides detection and isolation of any fault that enters into the differential equation of only a synchronously measured state, and grouping of faults that enter into the differential equation of any asynchronously measured state. For a fully coupled process system, fault-detection occurs shortly after a fault takes place, and fault isolation, limited by the arrival of asynchronous measurements, occurs when asynchronous measurements become available. Fault-tolerant control methods with a supervisory control component are then employed to achieve stability in the presence of fail-

ures. Numerical simulations of a polyethylene reactor are performed to demonstrate the applicability and performance of the proposed fault-detection and isolation and fault-tolerant control scheme in the presence of asynchronous measurements.

Chapter 8 summarizes the research contributions of this doctoral dissertation in terms of fault-detection and isolation and fault tolerant control, reverse osmosis water desalination systems, and FDI for process systems that rely on asynchronously measured state variables.

## Chapter 2

# Integrated fault-detection and fault-tolerant control

### 2.1 Introduction

Close examination of the existing literature indicates a lack of general and practical methods for the design of integrated fault-detection and fault-tolerant control structures for chemical plants accounting explicitly for actuator/controller failures, process nonlinearities and input constraints. Motivated by these considerations the problem of implementing fault-detection and fault-tolerant control to single input nonlinear processes with input constraints subject to control actuator failures is considered in this chapter. An approach predicated upon the idea of integrating fault-detection, feedback and supervisory control is presented and demonstrated. To illustrate the main idea behind the proposed approach, we first assume availability of measurements of all the process state variables. For the processes under consideration, a family of candidate control configurations, characterized by different manipulated inputs, is first identified. For each control configuration, a Lyapunov-based controller that enforces asymptotic closed-loop stability in the presence of constraints, is designed,

and the constrained stability region, associated with it, is explicitly characterized. A fault-detection filter is used to compute the expected closed-loop behavior in the absence of faults. Deviations of the process states from the expected closed-loop behavior are used to detect faults. A switching policy is then derived, on the basis of the stability regions, to orchestrate the activation/deactivation of the constituent control configurations in a way that guarantees closed-loop stability in the event that a failure is detected. Often, in chemical process applications, all state variables are not available for measurement. To deal with the problem of lack of process state measurements, a nonlinear observer is designed to generate estimates of the states, which are then used to implement the state feedback controller and the fault-detection filter. A switching policy is then derived to orchestrate the activation/deactivation of the constituent control configurations in a way that accounts for the estimation error. Finally, simulation studies are presented to demonstrate the implementation and evaluate the effectiveness of the proposed fault-tolerant control scheme.

## 2.2 Preliminaries

### 2.2.1 Process description

We consider a class of continuous-time, single-input nonlinear processes with constraints on the manipulated input, represented by the following state-space description:

$$\begin{aligned} \dot{x}(t) &= f(x(t)) + g_{k(t)}(x(t))(u_{k(t)} + m_{k(t)}), & y_m &= h_m(x) \\ k(t) &\in \mathcal{K} = \{1, \dots, N\}, \quad N < \infty, & |u_{k(t)}| &\leq u_{max}^k \end{aligned} \quad (2.1)$$

where  $x(t) \in \mathbb{R}^n$  denotes the vector of process state variables,  $y_m \in \mathbb{R}$  denotes the measured variable,  $u_k(t) \in [-u_{max}^k, u_{max}^k] \subset \mathbb{R}$  denotes the constrained manipulated input associated with the  $k$ -th control configuration and  $m_{k(t)} \in \mathbb{R}$  denotes the fault in the  $k$ -th control configuration. For each value that  $k$  assumes in  $\mathcal{K}$ , the process is

controlled via a different manipulated input which defines a given control configuration.

It is assumed that the origin is the equilibrium point of the nominal process (i.e.,  $f(0) = 0$ ),  $g_k(x) \neq 0 \forall x \in \mathbb{R}^n$ , and that the vector functions  $f(\cdot)$  and  $g_k(\cdot)$  are sufficiently smooth, for all  $k$ , on  $\mathbb{R}^n$ . Throughout the manuscript, a function  $\beta(r, s)$  is said to belong to class  $\mathcal{KL}$  if, for each fixed  $s$ , the mapping  $\beta(\cdot, s)$  belongs to class  $\mathcal{K}$  (a continuous function  $\alpha(\cdot)$  is said to belong to class  $\mathcal{K}$  if it is strictly increasing and  $\alpha(0) = 0$ ) and for each fixed  $r$ , the mapping  $\beta(r, \cdot)$  is decreasing, and  $\beta(r, s) \rightarrow 0$  as  $s \rightarrow \infty$ ; see also [49]. The notation  $\|\cdot\|$  is used to denote the standard Euclidean norm of a vector, the notation  $|\cdot|$  is used to denote the absolute value of a scalar and  $x'$  denotes the transpose of  $x$  and the notation  $R = [r_1 \ r_2]$  is used to denote the augmented vector  $R \in \mathbb{R}^{m+n}$  comprising of the vectors  $r_1 \in \mathbb{R}^m$  and  $r_2 \in \mathbb{R}^n$ . The notation  $L_f h$  denotes the standard Lie derivative of a scalar function  $h(\cdot)$  with respect to the vector function  $f(\cdot)$  and the notation  $x(T^+)$  denotes the limit of the trajectory  $x(t)$  as  $T$  is approached from the right, i.e.,  $x(T^+) = \lim_{t \rightarrow T^+} x(t)$ . Throughout the manuscript, we assume that for any  $|u_k| \leq u_{max}^k$  the solution of the system of Eq.2.1 exists and is continuous for all  $t$ .

### 2.2.2 Motivating example

To motivate our fault-tolerant control design methodology, we introduce in this subsection a benchmark chemical reactor example that will be used to illustrate the design and implementation of the fault-tolerant control structure. To this end, consider a well-mixed, non-isothermal continuous stirred tank reactor (see Fig. 2.1) 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. Under standard modeling assumptions, a mathemat-



ical model of the process can be derived from material and energy balances and takes the following form:

$$\begin{aligned}
\frac{dT}{dt} &= \frac{F}{V}(T_{A0} - T) + \sum_{i=1}^3 \frac{(-\Delta H_i)}{\rho c_p} k_{i0} \exp\left(\frac{-E_i}{RT}\right) C_A + \frac{Q}{\rho c_p V} \\
\frac{dC_A}{dt} &= \frac{F}{V}(C_{A0} - C_A) - \sum_{i=1}^3 k_{i0} \exp\left(\frac{-E_i}{RT}\right) C_A \\
\frac{dC_B}{dt} &= -\frac{F}{V}C_B + k_{10} \exp\left(\frac{-E_1}{RT}\right) C_A
\end{aligned} \tag{2.2}$$

where  $C_A$  and  $C_B$  denote the concentrations of the species  $A$  and  $B$ ,  $T$  denotes the temperature of the reactor,  $Q$  denotes the rate of heat input/removal from the reactor,  $V$  denotes the volume of the reactor,  $\Delta H_i$ ,  $k_i$ ,  $E_i$ ,  $i = 1, 2, 3$ , denote the enthalpies, pre-exponential constants and activation energies of the three reactions, respectively,  $c_p$  and  $\rho$  denote the heat capacity and density of the reactor, respectively. The values of the process parameters and the corresponding steady-state values can be found in [27]. It was verified that under these conditions, the process of Eq.2.2 has three steady-states (two locally asymptotically stable and one unstable at  $(T_s, C_{As}, C_{Bs}) = (388.57 \text{ K}, 3.59 \text{ kmol/m}^3, 0.41 \text{ kmol/m}^3)$ ).

The control objective considered here is the one of stabilizing the reactor at the (open-loop) unstable steady-state. Operation at this point is typically sought to avoid high temperature, while simultaneously achieving reasonable conversion. To accomplish this objective in the presence of control system failures, we consider as manipulated inputs the rate of heat input,  $u_1 = Q$ , subject to the constraints  $|Q| \leq u_{max}^1 = 748 \text{ KJ/s}$ , the inlet stream temperature,  $u_2 = T_{A0} - T_{A0s}$ , subject to the constraints  $|u_2| \leq u_{max}^2 = 100 \text{ K}$ , with  $T_{A0s} = 300 \text{ K}$  and the inlet reactant concentration,  $u_3 = C_{A0} - C_{A0s}$ , subject to the constraints  $|u_3| \leq u_{max}^3 = 4 \text{ kmol/m}^3$ , with  $C_{A0s} = 4 \text{ kmol/m}^3$ .

Each of the above manipulated inputs, together with measurements of reactor tem-

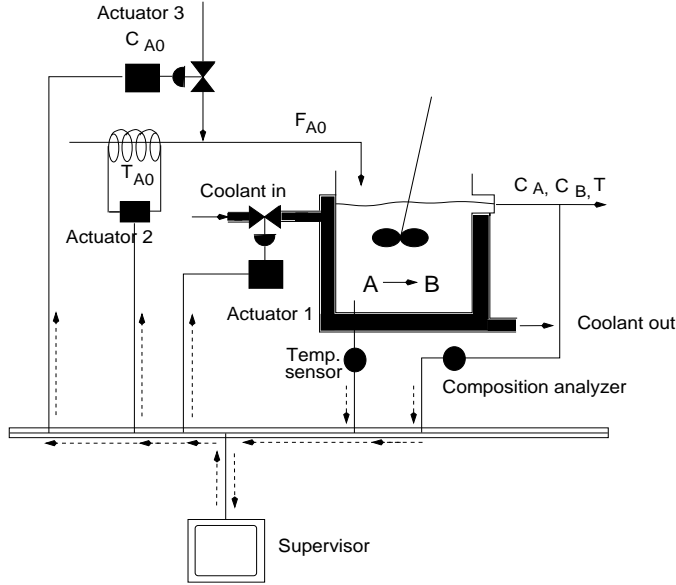


Figure 2.1: A schematic of the CSTR showing the three candidate control configurations.

perature and/or concentration, represents a unique control configuration (or control-loop) that, by itself, can stabilize the reactor. In the event of some failure in the primary configuration (involving the heat input,  $Q$ ), the important questions that arise include how can the supervisor detect this fault (note that measurements of the manipulated input variable are not available), and which control loop to activate once failure is detected in the active configuration. The answer to the first question involves the design of an appropriate fault-detection filter. The approach that we will utilize to answer the second question, i.e., that of deciding which backup controller should be activated in the event of a fault, will be based on the stability regions under the individual control configuration. To this end, we next review a state feedback control design that allows for characterizing the constrained stability region under each control configuration. Note that this particular choice of the controller is presented only as an example to illustrate our results, and that any other controller design that allows for an explicit characterization of the constrained stability region can be used

instead. Note also, that while the above example will be used to illustrate the main ideas behind the proposed fault-detection and fault-tolerant control method, we also investigate in the simulation studies an application to a network of chemical reactors in the presence of uncertainty and measurement noise.

### 2.2.3 Bounded Lyapunov-based control

Consider the system of Eq.2.1, for which a family of control Lyapunov functions (CLFs),  $V_k(x)$ ,  $k \in \mathcal{K} \equiv \{1, \dots, N\}$  has been found (see below for a discussion on the construction of CLFs). Using each control Lyapunov function, we construct, using the results in [54] (see also [24]), the following continuous bounded control law:

$$u_k(x) = -\frac{L_f^* V_k(x) + \sqrt{(L_f^* V_k(x))^2 + (u_{max}^k \|(L_{g_k} V_k)(x)\|)^4}}{\|(L_{g_k} V_k)(x)\|^2 \left[1 + \sqrt{1 + (u_{max}^k \|(L_{g_k} V_k)(x)\|)^2}\right]} (L_{g_k} V_k)(x) \quad (2.3)$$

when  $(L_{g_k} V_k)(x) \neq 0$  and  $u_k(x) = 0$  when  $(L_{g_k} V_k)(x) = 0$ ,  $L_f^* V_k(x) = \frac{\partial V_k(x)}{\partial x} f(x) + \rho_k V_k(x)$ ,  $\rho_k > 0$  and  $L_{g_k} V_k(x) = \frac{\partial V_k(x)}{\partial x} g_k(x)$ . Let  $\Pi_k$  be the set defined by

$$\Pi_k(u_{max}^k) = \{x \in \mathbb{R}^n : L_f^* V_k(x) \leq u_{max}^k \|(L_{g_k} V_k)(x)\|\} \quad (2.4)$$

and assume that

$$\Omega_k := \{x \in \mathbb{R}^n : V_k(x) \leq c_k^{max}\} \subseteq \Pi_k(u_{max}^k) \quad (2.5)$$

for some  $c_k^{max} > 0$ . It can be shown, using standard Lyapunov arguments, that in the absence of faults ( $m_{k(t)} = 0$ ),  $\Omega_k$  provides an estimate of the stability region, starting from where the control law of Eq.2.3 guarantees asymptotic (and local exponential) stability of the origin of the closed-loop system under each control configuration. This implies that there exist class  $\mathcal{KL}$  functions  $\beta_i$ ,  $i = 1, \dots, N$ , such that  $\|x(t)\| \leq \beta_i(\|x(0)\|, t)$ . We will use this property later in the design of the output feedback controllers.

Referring to the above controller design, it is important to make the following remarks. First, a general procedure for the construction of CLFs for nonlinear systems of the form of Eq.2.1 is currently not available. Yet, for several classes of nonlinear systems that arise commonly in the modeling of engineering applications, it is possible to exploit system structure to construct CLFs (see, for example, [53, 33]). Second, given that a CLF,  $V_k$ , has been obtained for the system of Eq.2.1, it is important to clarify the essence and scope of the additional assumption that there exists a level set,  $\Omega_k$ , of  $V_k$  that is contained in  $\Pi_k$ . Specifically, the assumption that the set,  $\Pi_k$ , contains an invariant subset around the origin, is necessary to guarantee the existence of a set of initial conditions for which closed-loop stability is guaranteed (note that even though  $\dot{V}_k < 0 \forall x \in \Pi_k \setminus \{0\}$ , there is no guarantee that trajectories starting within  $\Pi_k$  remain within  $\Pi_k$  for all times). Moreover, the assumption that  $\Omega_k$  is a level set of  $V_k$  is made only to simplify the construction of  $\Omega_k$ . This assumption restricts the applicability of the proposed control method because a direct method for the construction of a CLF with level sets contained in  $\Pi_k$  is not available. However, the proposed control method remains applicable if the invariant set  $\Omega_k$  is not a level set of  $V_k$  but can be constructed in some other way (which, in general, is a difficult task). Note also that possibly larger estimates of the stability region can be computed using constructive procedures such as Zubov's method [21] or by using a combination of several Lyapunov functions.

## 2.3 State feedback case

### State feedback fault-tolerant control

Consider the system of Eq.2.1, where all process states are available as measurements, i.e.,  $h_m(x) = x$ , and without loss of generality, assume that it starts operating using

control configuration  $i$ , under the controller of Eq.2.3. At some unknown time,  $T_i^f$ , a fault occurs in the first control configuration such that for all  $t \geq T_i^f$ ,  $m_i = -u_i$ , i.e., control configuration  $i$  fails. The problems at hand are those of detecting that a fault has occurred and, upon detection, to decide which of the available backup configurations should be implemented in the closed-loop to achieve fault-tolerant control. To this end, we consider a fault-detection filter and a switching logic of the form:

$$\dot{w}(t) = f_f(w, x), \quad r(t) = h_f(w, x), \quad k(t) = \varphi(r, w, x) \quad (2.6)$$

where  $w \in \mathbb{R}^n$  is the state of the filter,  $r(t) \in \mathbb{R}$  is a residual that indicates the occurrence of a fault, and is the output of the filter,  $f_f \in \mathbb{R}^n$  is the vector field describing the evolution of the filter state  $w$ , and  $\varphi(r, w, x)$  is the switching logic that dictates which of the available control configurations should be activated.

The main idea behind the fault-tolerant control design is as follows: (1) use the available state measurements, the process model, and the computed control action to simulate the evolution of the closed-loop process in the absence of actuator faults, compare it with the actual evolution of the states, and use the difference between the two behaviors, if any, to detect faults, and (2) having detected the fault, activate a backup control configuration for which the closed-loop state is within its stability region estimate. To formalize this idea, consider the constrained system of Eq.2.1 for which a bounded controller of the form of Eq.2.3 has been designed for each control configuration, and the stability region,  $\Omega_j$ ,  $j = 1, \dots, N$  has been explicitly characterized. The fault-detection filter and the fault-tolerant control design are described in Theorem 2.1 below.

**Theorem 2.1:** *Let  $k(0) = i$  for some  $i \in \mathcal{K}$  and  $x(0) := x_0 \in \Omega_i$ . Set  $w(0) = x(0)$ , and consider the system*

$$\dot{w} = f(w) + g_i(w)u_i(w); \quad r = \|w - x\| \quad (2.7)$$

where  $w \in \mathbb{R}^n$  is the filter state and  $u_i(\cdot)$  is the feedback control law defined in Eq.2.3. Let  $T_i^f$  be such that  $m_i(t) = 0 \forall 0 \leq t \leq T_i^f$ , then  $r(T_i^{f+}) > 0$  if and only if  $m_i(T_i^f) \neq 0$ . Furthermore, let  $T_i^s$  be the earliest time such that  $r(t) > 0$ , then the following switching rule:

$$k(t) = \left\{ \begin{array}{ll} i, & 0 \leq t < T_i^s \\ j \neq i, & t \geq T_i^s, x(T_i^s) \in \Omega_j \end{array} \right\} \quad (2.8)$$

guarantees asymptotic stability of the origin of the closed-loop system.

**Proof of Theorem 2.1:** We split the proof of the theorem in two parts. In the first part we show that the filter detects a fault if and only if one occurs, and in the second part we establish closed-loop stability under the switching rule of Eq.2.8.

*Part 1:* Let  $x(T_i^f) := x_{T_i^f}$  and  $w(T_i^f) := w_{T_i^f}$  and consider

$$\dot{w}(T_i^f) - \dot{x}(T_i^f) = f(x_{T_i^f}) + g(x_{T_i^f})(u_i(x_{T_i^f}) + m_i(T_i^f)) - (f(w_{T_i^f}) + g(w_{T_i^f})u_i(w_{T_i^f})) \quad (2.9)$$

with  $m_i(T_i^f) \neq 0$ . Since  $w_{T_i^f} = x_{T_i^f}$ , we have that

$$f(x_{T_i^f}) + g(x_{T_i^f})(u_i(x_{T_i^f}) + m_i(T_i^f)) - (f(w_{T_i^f}) + g(w_{T_i^f})u_i(w_{T_i^f})) = g(x_{T_i^f})m_i(T_i^f) \quad (2.10)$$

Furthermore, since  $g(x_{T_i^f}) \neq 0$ , we have that

$$\dot{w}(T_i^f) - \dot{x}(T_i^f) = g(x_{T_i^f})m_i(T_i^f) \neq 0 \quad (2.11)$$

if and only if  $m_i(T_i^f) \neq 0$ . Since  $w_{T_i^f} - x_{T_i^f} = 0$  and  $\dot{w}(T_i^f) - \dot{x}(T_i^f) \neq 0$  if and only if  $m_i(T_i^f) \neq 0$ , we have that

$$w(T_i^{f+}) - x(T_i^{f+}) \neq 0 \quad (2.12)$$

or

$$r(T_i^{f+}) = \|w(T_i^{f+}) - x(T_i^{f+})\| > 0 \quad (2.13)$$

if and only if  $m_i(T_i^f) \neq 0$ .

*Part 2:* We prove closed-loop stability for the two possible cases; first if no switching occurs, and second if a switch occurs at a time  $T_i^s$ .

*Case 1:* The absence of a switch implies  $r_i(t) = 0$ . Furthermore,  $r_i(t) = 0 \implies x(t) = w(t)$ . Since  $x(0) = w(0) \in \Omega_i$ , and control configuration  $i$  is implemented for all times in this case, we have that asymptotic closed-loop stability is achieved.

*Case 2:* At time  $T_i^s$ , the supervisor switches to a control configuration  $j$  for which  $x(T_i^s) \in \Omega_j$ . From this time onwards, since configuration  $j$  is implemented in the closed-loop system for all times, and since  $x(T_i^s) \in \Omega_j$ , closed-loop stability follows. This completes the proof of Theorem 2.1.

The fault-detection filter and fault-tolerant controller are designed and implemented as follows (see also Fig.2.2):

- Given any  $x_0 \in \Omega_i$ , initialize the filter states as  $w(0) = x_0$  and integrate the filter dynamics using Eq.2.7.
- Compute the norm of the difference between the filter states and the process states,  $r(t) = \|w(t) - x(t)\|$  and if  $r(t) = 0$ , continue to implement control configuration  $i$ .
- At any time  $T_i^s$  that  $r(T_i^s) > 0$ , switch to a control configuration  $j \neq i$ , for which  $x(T_i^s) \in \Omega_j$  to achieve asymptotic stability of the origin of the closed-loop system.

Note that the fault-detection filter uses a replica of the process dynamics, and that the state of the filter  $w$  is initialized at the same value as the process states  $x(0)$ . In the absence of faults, the evolution of  $w(t)$  is identical to  $x(t)$ , and hence  $r(t) = 0$ . In the presence of faults, however, the effect of the fault is registered by a change in the evolution of the process, but not in that of the filter state (since the filter

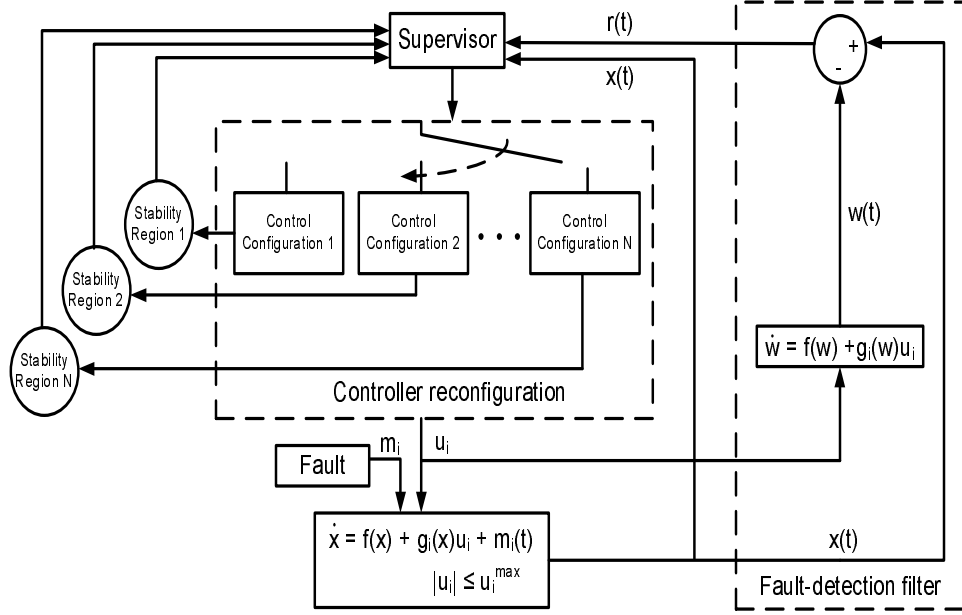


Figure 2.2: Integrated fault-detection and fault-tolerant control design: state feedback case.

state dynamics include the computed control action,  $u_i(w)$ , and not the implemented control action,  $u_i(w) + m_i$ . This change is detected by a change in the value of  $r(t)$  and declared as a fault. Note also, that the fact that the faults  $m_i$  appear as additive terms to the manipulated input variable is a natural consequence of focussing on the problem of detecting (through the design of appropriate fault-detection filters) and dealing (via reconfiguration) with faults in control actuators. The approach employed in the design of the fault-detection filter can also be used to detect faults that do not necessarily appear in the control actuators, as long as they influence the evolution of the state variables.

**Remark 2.1:** Once a fault is detected, the switching logic ensures that the backup control configuration that is implemented in the closed-loop is one that can guarantee closed-loop stability in the presence of constraints, and this is achieved by verifying that the state of the process, at the time that a fault is detected, is present in the constrained stability region of the candidate control configuration. Note that while



the bounded controller is used for a demonstration of the main ideas, other control approaches, that provide an explicit characterization of the set of initial conditions for which closed-loop stability is guaranteed (achieved, for example, via the use of the hybrid predictive approach [28] or via a Lyapunov-based model predictive control design [63]) can be used within the proposed framework. Note also that early detection of a fault enhances the chances that corrective action can be taken in time to achieve fault-tolerant control (Theorem 2.1 guarantees that a fault is detected as soon as it occurs). Specifically, it may happen that a fault occurs when the closed-loop state resides in the stability region of one of the backup configurations, but if the fault is not immediately detected, the destabilizing effect of the fault may drive the state outside the stability region of the backup configuration by the time a fault is detected (for a demonstration, see the simulation example).

In the event that the process state, at the time of the failure of the primary control configuration, lies in the stability region of more than one backup control configuration, additional performance considerations such as ease and/or cost of implementing one control configuration over another, can be used in choosing which control configuration should be implemented in the closed-loop system [66]. If the state at the time of a failure lies outside the stability region of all the backup controllers, then this indicates that the back up controllers do not have enough control action available and calls for increasing the allowable control action in the fall-back configurations. Note that the set of initial conditions starting from where a given control configuration can stabilize a steady state – the so-called null-controllable region – is fundamentally limited by the constraints on the available control action, and that different control laws typically provide estimates of the stability region which are subsets of the null-controllable region.

**Remark 2.2:** In the presence of plant model mismatch or unknown disturbances, the value of  $r(t)$  will be nonzero even in the absence of faults. The FDFTC problem in the presence of time varying disturbances with known bounds on the disturbances can be handled by (1) redesigning the filter to account for the disturbances; specifically, requiring that a fault be declared only if the value of  $r(t)$  increases beyond some threshold,  $\delta$ , where  $\delta$  accounts for the deviation of the plant dynamics from the nominal dynamics in the absence of faults (please see the simulation example for a demonstration of this idea in an application to a network of chemical reactors in the presence of uncertainty and measurement noise) and (2) by redesigning the controllers for the individual control configurations to mitigate the effect of disturbances on the process, and characterizing the robust stability regions and using them as criteria for deciding which backup controller should be implemented in the closed-loop. Note that while Theorem 2.1 presents the fault-detection filter and fault-tolerant control (FDFTC) design for a fault in the primary control configuration, extensions to faults in successive backup configurations are straightforward and involve similar filter designs for the active control configuration and a switching logic that orchestrates switching to the remaining control configurations.

**Remark 2.3:** While we illustrate our idea using a single input, extensions to multi-input systems are possible, and fault-detection filters can be designed in the same way, using a replica of the process dynamics. The case of multi-input systems, however, introduces an additional layer of complexity due to the need of identifying which particular manipulated input has failed, i.e., the additional problem of fault-isolation. For the purpose of presenting the integrated fault-detection and fault-tolerant control structure, we focus here on multiple control configurations, where each control configuration comprises of a single input that does not require the filter to perform the

additional task of fault-isolation. For a simple illustration of a fault-detection and isolation filter design, please see the simulation example.

**Remark 2.4:** Note that the fault-detection filter presented in Theorem 2.1 detects the presence of both complete and partial failures. Once a fault is detected, the control reconfiguration strategy is the same for both cases, and that is to shut down the faulty configuration and switch to some well-functioning fall-back configuration. Note that in the case of a partial failure, unless the faulty configuration is shut down, the backup control configurations will have to be redesigned to be robust with respect to the bounded disturbance generated by the faulty configuration (for the backup control configuration, the unmeasured actuator action of the faulty control configuration will act as a disturbance and will be bounded because of the fact that the actuator itself has a limited capacity and, therefore, even if the implemented control action is not the same as that prescribed by the controller, it cannot exceed the physical limitations and will remain bounded). By shutting down the faulty configuration, however, the source of the disturbance is eliminated and no controller redesign is needed for the backup control configurations.

### 2.3.1 Simulation results

In this subsection, we illustrate the implementation of the proposed fault-detection/fault-tolerant control methodology to the chemical reactor introduced as a motivating example. We first describe the controller design for the individual control configurations. Note that our objective is full state stabilization; however, to facilitate the controller design and subsequent stability analysis, we use a state transformation to transform the system of Eq.2.2.2 into the following one describing the input/output dynamics:

$$\dot{e} = Ae + l_k(e) + b\alpha_k u_k \quad := \quad \bar{f}_k(e) + \bar{g}_k(e)u_k \quad (2.14)$$

where  $e \in \mathbb{R}^n$  is the variable in transformed co-ordinate (for the specific transformations used for each control configuration, please see below),  $A = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}$ ,  $b = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$ ,  $l_k(\cdot) = L_{f_k}^2 h_k(x)$ ,  $\alpha_k(\cdot) = L_{g_k} L_{f_k} h_k(x)$ ,  $h_k(x) = y_k$  is the output associated with the  $k$ -th configuration,  $x = [x_1 \ x_2]^T$  with  $x_1 = T - T_s$ ,  $x_2 = C_A - C_{As}$ , and the functions  $f_k(\cdot)$  and  $g_k(\cdot)$  can be obtained by re-writing the  $(T, C_A)$  model equations in Eq.2.2 in the form of Eq.2.1. The explicit forms of these functions are omitted for brevity. 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  $A^T P_k + P_k A - P_k b b^T P_k < 0$ , is used for controller design. In particular:

1. For the first configuration with  $u_1 = Q$ , we consider the controlled output  $y_1 = C_A - C_{As}$ . The coordinate transformation (in error variables form) takes the form:  $e_1 = C_A - C_{As}$ ,  $e_2 = \frac{F}{V}(C_{A0} - C_A) - \sum_{i=1}^3 k_{i0} e^{\frac{-E_i}{RT}} C_A$  and yields a relative degree of two with respect to the manipulated input.
2. For the second configuration with  $u_2 = T_{A0} - T_{A0s}$ , we choose the output  $y_2 = C_A - C_{As}$  which yields the same relative degree as in the first configuration,  $r_2 = 2$ , and the same coordinate transformation.
3. For the third configuration with  $u_3 = C_{A0} - C_{A0s}$ , a coordinate transformation of the form used for configurations 1 and 2 above does not yield a sufficiently large estimate of the stability region, we therefore choose a candidate Lyapunov function of the form  $V_3(x) = x' P x$ , where  $P > 0$  and  $x = [T - T_s \ C_A - C_{As}]'$  with  $P = \begin{bmatrix} 0.011 & 0.019 \\ 0.019 & 0.101 \end{bmatrix}$ .

Fig.2.3 depicts the stability region, in the  $(T, C_A)$  space, for each configuration. The desired steady-state is depicted with an asterisk that lies in the intersection of the three stability regions. The reactor as well as the fault-detection filter for

the first control configuration is initialized at  $T(0) = 330 \text{ K}$ ,  $C_A(0) = 3.6 \text{ kmol/m}^3$ ,  $C_B(0) = 0.0 \text{ kmol/m}^3$ , using the  $Q$ -control configuration, and the supervisor proceeds to monitor the evolution of the closed-loop trajectory.

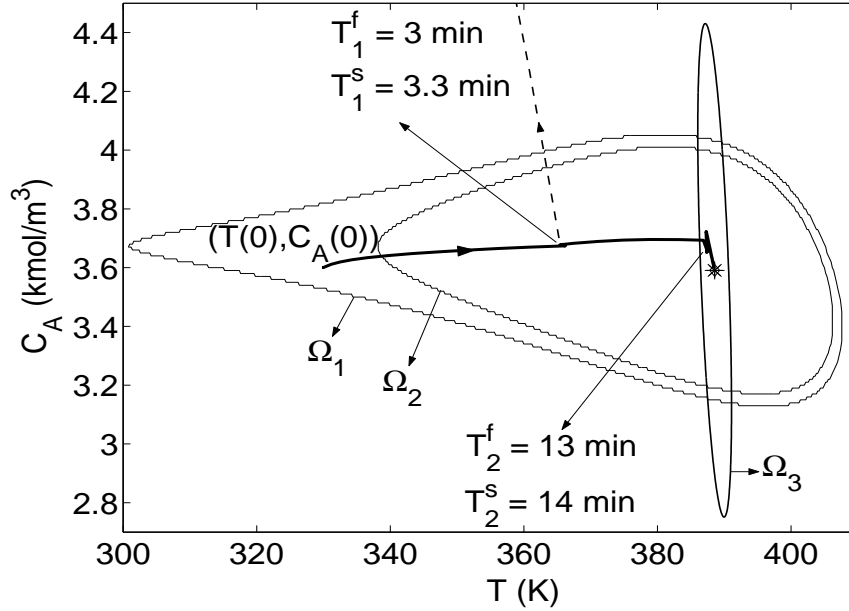
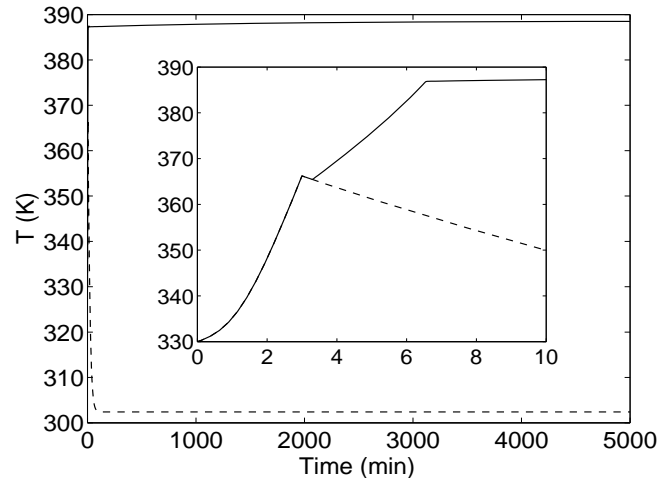
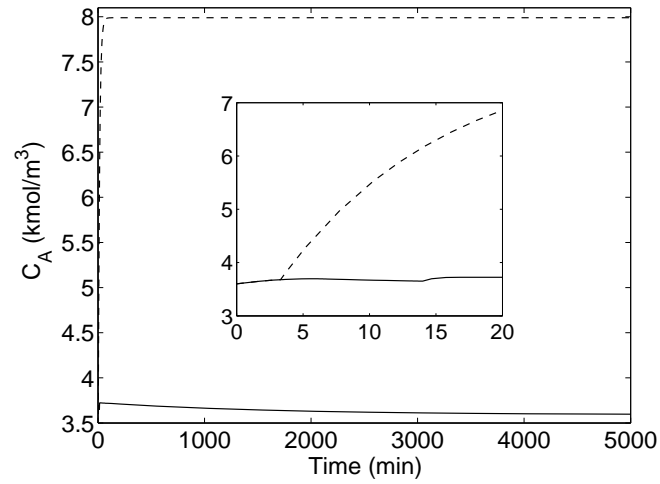


Figure 2.3: Evolution of the closed-loop state profiles under the switching rule of Eq.2.8 subject to failures in control systems 1 and 2 (solid line) and under arbitrary switching (dashed line).

As shown by the solid lines in Figs.2.3-2.4, the controller proceeds to drive the closed-loop trajectory towards the desired steady-state, up until the  $Q$ -configuration fails after 3 minutes of reactor startup (see Fig.2.6a). As can be seen in Fig.2.5a, at this time the value of  $r_1(t)$  becomes non-zero and the fault-detection filter detects this fault. If the supervisor switches arbitrarily, and in particular, switches to backup configuration 3, closed-loop stability is not achieved (dashed lines in Figs.2.3-2.4). Note that this happens because the closed-loop state is outside the stability region of the third control configuration, and even though the third control configuration does not encounter a fault ( $r_3(t) = 0$ ; see dashed line in Fig.2.5b), the limited control



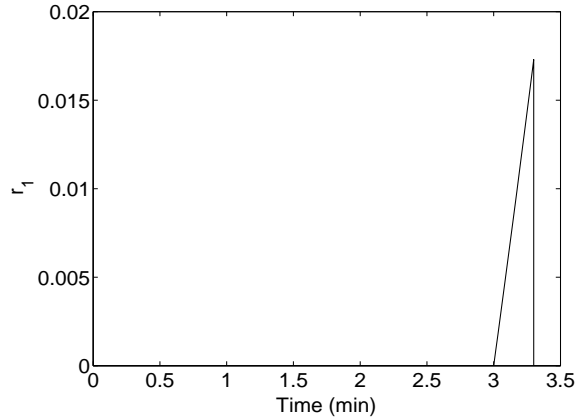
(a)



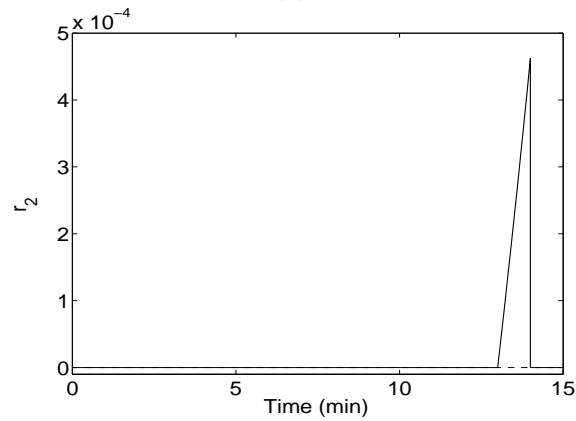
(b)

Figure 2.4: Evolution of the closed-loop (a) temperature and (b) concentration under the switching rule of Eq.2.8 subject to failures in control systems 1 and 2 (solid lines) and under arbitrary switching (dashed lines).

action available in this configuration is unable to achieve closed-loop stability. On the basis of the switching logic of Eq.2.8, the supervisor activates the second configuration (with  $T_{A0}$  as the manipulated input, see Fig.2.6b), which continues to drive the state trajectory closer to the desired steady-state.



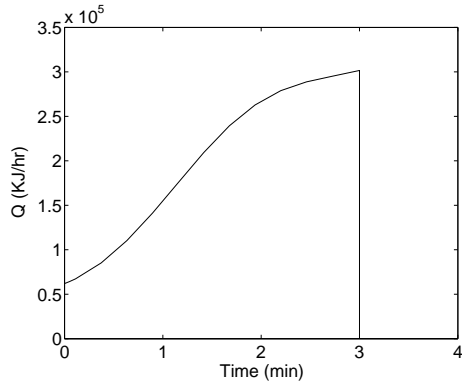
(a)



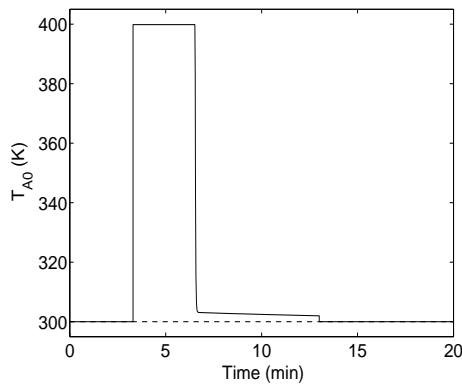
(b)

Figure 2.5: Evolution of the closed-loop residual under the fault-detection filter for (a) control configuration 1 and (b) control configurations 2 and 3 under the switching rule of Eq.2.8 subject to failures in control systems 1 and 2 (solid lines) and under arbitrary switching (dashed lines).

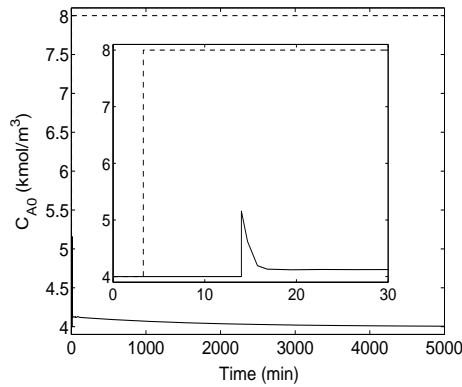
To demonstrate the implementation of the proposed FDFTC strategy when faults occur in successive control configurations, we consider the case when a second failure



(a)



(b)



(c)

Figure 2.6: Manipulated input profiles under (a) control configuration 1, (b) control configuration 2, and (c) control configuration 3 under the switching rule of Eq.2.8 subject to failures in control systems 1 and 2 (solid lines) and under arbitrary switching (dashed lines).



occurs (this time in the  $T_{A0}$ -configuration) at  $t = 13$  minutes. Once again, the filter detects this failure via an increase in the value of  $r_2(t)$  (solid line in Fig.2.5b) using the fault-detection filter for control configuration 2. From Fig.2.3, it is clear that the failure of the second control configuration 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, see Fig.2.6c) which finally stabilizes the reactor at the desired steady-state.

## 2.4 Output feedback case

The feedback controllers, the fault-detection filters and the switching rules in the previous section were designed under the assumption of availability of measurements of all the process states. The unavailability of full state measurements has several implications. First, it necessitates generating estimates of the states to be used in conjunction with both the state feedback controller and the fault-detection filter. The state estimates, however, contain errors, and this results in a difference between the expected closed-loop behavior of the measured variables (computed using the state estimates) and the evolution of the measured variables, even in the absence of actuator faults. The fault-detection filter has to be redesigned to account for this fact so that it does not treat this difference to be an indicator of an actuator fault (i.e., to prevent a false alarm). Also, the switching logic has to account for the fact that the supervisor can monitor only the state estimates and needs to make inferences about the true values of the states using the state estimates.

In the remainder of this section, we first review an output feedback controller design, proposed in [25], based on a combination of a high-gain observer and a state feedback controller (see also [57, 47, 48, 89, 12] for results on observer designs and

output feedback control for unconstrained nonlinear systems) and characterize the stability properties of the closed-loop system under output feedback control. Then, we present the fault-detection filter and fault-tolerant controller and demonstrate its application via a simulation example.

### 2.4.1 Output feedback control

To facilitate the design of a state estimator with the required convergence properties, we make the following assumption:

**Assumption 2.1:** *For each  $i \in \mathcal{K}$ , there exists a set of coordinates*

$$\begin{bmatrix} \xi_i \end{bmatrix} = \begin{bmatrix} \xi_i^1 \\ \xi_i^2 \\ \vdots \\ \xi_i^n \end{bmatrix} = \chi_i(x) = \begin{bmatrix} h_m(x) \\ L_f h_m(x) \\ \vdots \\ L_f^{n-1} h_m(x) \end{bmatrix} \quad (2.15)$$

such that the system of Eq. 2.1 takes the form

$$\begin{aligned} \dot{\xi}_i^1 &= \xi_i^2 \\ &\vdots \\ \dot{\xi}_i^{n-1} &= \xi_i^n \\ \dot{\xi}_i^n &= L_f^n h_m(\chi_i^{-1}(\xi)) + L_{g_i} L_f^{n-1} h_m(\chi_i^{-1}(\xi))(u_{i(t)} + m_{i(t)}) \end{aligned} \quad (2.16)$$

where  $L_{g_i} L_f^{n-1} h_m(x) \neq 0$  for all  $x \in \mathbb{R}^n$ . Also,  $\xi_i \rightarrow 0$  if and only if  $x \rightarrow 0$ .

We note that the change of variables is invertible, since for every  $x$ , the variable  $\xi_i$  is uniquely determined by the transformation  $\xi_i = \chi_i(x)$ . This implies that if one can estimate the values of  $\xi_i$  for all times, using an appropriate state observer, then we automatically obtain estimates of  $x$  for all times, which can be used to implement the state feedback controller. The existence of such a transformation will facilitate the design of high-gain observers which will be instrumental in preserving the same closed-loop stability properties achieved under full state feedback.

Proposition 2.1 below presents the output feedback controller used for each mode and characterizes its stability properties. The proof of the proposition, which invokes singular perturbation arguments (for a result on input-to-state stability with respect to singular perturbations, and further references, see [14]), is a special case of the proof of Theorem 2.2 in [25], and is omitted for brevity. To simplify the statement of the proposition, we first introduce the following notation. We define  $\alpha_i(\cdot)$  as a class  $\mathcal{K}$  function that satisfies  $\alpha_i(\|x\|) \leq V_i(x)$ . We also define the set  $\Omega_{b,i} := \{x \in \mathbb{R}^n : V_i(x) \leq \delta_{b,i}\}$ , where  $\delta_{b,i}$  is chosen such that  $\beta_i(\alpha_i^{-1}(\delta_{b,i}), 0) < \alpha_i^{-1}(c_i^{max})$ , where  $\beta_i(\cdot, \cdot)$  is a class  $\mathcal{KL}$  function and  $c_i^{max}$  is a positive real number defined in Eq.2.5.

**Proposition 2.1:** *Consider the nonlinear system of Eq.2.1, for a fixed mode,  $k(t) = i$ , and with  $m_i(t) \equiv 0$ , under the output feedback controller:*

$$\begin{aligned} \dot{\tilde{y}} &= \begin{bmatrix} -L_i a_1^{(i)} & 1 & 0 & \cdots & 0 \\ -L_i^2 a_2^{(i)} & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ -L_i^n a_n^{(i)} & 0 & 0 & \cdots & 0 \end{bmatrix} \tilde{y} + \begin{bmatrix} L_i a_1^{(i)} \\ L_i^2 a_2^{(i)} \\ \vdots \\ L_i^n a_n^{(i)} \end{bmatrix} y_m \\ u_i &= u_i^c(\hat{x}, u_i^{max}) \end{aligned} \quad (2.17)$$

where  $u_i^c$  is defined in Eq.2.3, the parameters,  $a_1^{(i)}, \dots, a_n^{(i)}$  are chosen such that the polynomial  $s^n + a_1^{(i)} s^{n-1} + a_2^{(i)} s^{n-2} + \dots + a_n^{(i)} = 0$  is Hurwitz,  $\hat{x} = \chi_i^{-1}(\text{sat}(\tilde{y}))$ ,  $\text{sat}(\cdot) = \min\{1, \zeta_{max,i}/|\cdot|\}(\cdot)$ , with  $\zeta_{max,i} = \beta_\zeta(\delta_{\zeta,i}, 0)$  where  $\beta_\zeta$  is a class  $\mathcal{KL}$  function and  $\delta_{\zeta,i}$  is the maximum value of the norm of the vector  $[h_m(x) \cdots L_{f_i}^{n-1} h_m(x)]$  for  $V_i(x) \leq c_i^{max}$  and let  $\epsilon_i = 1/L_i$ . Then, given  $\Omega_{b,i}$ , there exists  $\epsilon_i^* > 0$  such that if  $\epsilon_i \in (0, \epsilon_i^*]$ ,  $x(0) \in \Omega_{b,i}$ , and  $\|\tilde{y}(0)\| \leq \delta_{\zeta,i}$ , the origin of the closed-loop system is asymptotically (and locally exponentially) stable. Furthermore, given any positive real numbers,  $e_{m,i}$  and  $T_i^b$ , there exists a real positive number  $\epsilon_i^{**}$  such that if  $\epsilon_i \in (0, \epsilon_i^{**}]$  then  $\|x(t) - \hat{x}(t)\| \leq e_{m,i}$  for all  $t \geq T_i^b$ .

The state observer in Eq.2.17 ensures sufficiently fast convergence that is nec-

essary for the implementation of both the state feedback controller (and preserving its stability properties under output feedback control), and the fault-detection filter. The most important feature of this estimator (and one that will be used in the fault-detection filter design) is that the estimation error is guaranteed to fall below a certain value in a small period of time,  $T_i^b$ , which can be chosen arbitrarily small by sufficiently increasing the observer gain. This requirement or constraint on the error dynamics is needed even when other estimation schemes, such as moving horizon observers, are used (for example, see [71, 81]). For such observers, however, it is difficult in general to obtain a transparent relationship between the tunable observer parameters and the error decay rate.

Due to the lack of full state measurements, the supervisor can rely only on the available state estimates to decide whether switching at any given time is permissible, and, therefore, needs to make reliable inferences regarding the position of the states based upon the available state estimates. Proposition 2.2 below establishes the existence of a set,  $\Omega_{s,i} := \{x \in \mathbb{R}^n : V_i(x) \leq \delta_{s,i}\}$ , such that once the state estimation error has fallen below a certain value (note that the decay rate can be controlled by adjusting  $L_i$ ), the presence of the state within the output feedback stability region,  $\Omega_{b,i}$ , can be guaranteed by verifying the presence of the state estimates in the set  $\Omega_{s,i}$ . A similar approach was employed in the construction of the output feedback stability regions  $\Omega_{b,i}$  and the regions for the state estimates  $\Omega_{s,i}$  in the context of output feedback control of linear systems in [62].

**Proposition 2.2:** *Given any positive real number  $\delta_{b,i}$ , there exist positive real numbers  $e_{m,i}^*$  and  $\delta_{s,i}$  such that if  $\|x - \hat{x}\| \leq e_{m,i}$ , where  $e_{m,i} \in (0, e_{m,i}^*]$ , and  $V_i(\hat{x}) \leq \delta_{s,i}$ , then  $V_i(x) \leq \delta_{b,i}$ .*

**Proof of Proposition 2.2:** From the continuity of the function  $V_i(\cdot)$ , we have that

for any positive real number  $e_{m,i}$ , there exists a positive real number  $\gamma_i$  such that  $\|x - \hat{x}\| \leq e_{m,i} \implies |V_i(x) - V_i(\hat{x})| \leq \gamma_i \implies V_i(x) \leq V_i(\hat{x}) + \gamma_i$ . Since  $\gamma_i$  can be made small by choosing  $e_{m,i}$  small, it follows that given any positive real number  $\delta_{b,i}$ , there exists a positive real number,  $e_{m,i}^*$ , such that for all  $e_{m,i} \in (0, e_{m,i}^*]$ ,  $\gamma_i < \delta_{b,i}$ . Now, let  $\delta_{s,i}$  be any positive real number that satisfies  $\delta_{s,i} + \gamma_i \leq \delta_{b,i}$ . Then if  $\|x - \hat{x}\| \leq e_{m,i} \leq e_{m,i}^*$  and  $V_i(\hat{x}) \leq \delta_{s,i}$ , we have  $V_i(x) \leq V_i(\hat{x}) + \gamma_i \leq \delta_{s,i} + \gamma_i \leq \delta_{b,i}$ . This completes the proof of the proposition.

Note that for the inference that  $\hat{x} \in \Omega_{s,i} \implies x \in \Omega_{b,i}$  to be useful in executing the switching, the set  $\Omega_{s,i}$  needs to be contained within  $\Omega_{b,i}$ . From Proposition 2.2, this can be ensured if  $e_{m,i}$  is sufficiently small, which in turn is ensured for all times greater than  $T_i^b$  provided that the observer gain is sufficiently large. In practice, use of a sufficiently high observer gain leads to an  $\Omega_{b,i}$  that is almost identical to  $\Omega_i$ , and furthermore, once the error has sufficiently decreased,  $\Omega_{s,i}$  can be taken to be almost equal to  $\Omega_{b,i}$ .

#### 2.4.2 Integrating fault-detection and fault-tolerant output feedback control

In this subsection we will present a fault-tolerant controller that uses the estimates generated by the high-gain observer for the implementation of the fault-detection filter, the state feedback controllers and the switching logic (see Fig.2.7). We proceed by first showing how the implementation of the design and implementation of the fault-detection filter should be modified to handle the absence of full state measurements. To this end, we consider the following system:

$$\begin{aligned} \dot{w}(t) &= f(w) + g_i(w)u_i(w) \\ r(t) &= \|\hat{x}(t) - w(t)\| \end{aligned} \tag{2.18}$$

Note that, as in the full state feedback case, the state equation for the filter in Eq.2.18 is a replica of the closed-loop state equation under full state feedback and in the ab-

sence of faults. However, because of the absence of full state measurements, the residual can only be defined in terms of the state estimates, not the actual states. The residual therefore provides a measure of the discrepancy between the evolution of the nominal closed-loop system (i.e., with no faults) under full state feedback and the evolution of the closed-loop state estimates under output feedback. Since the discrepancy can be solely due to estimation errors and not necessarily due to faults, it is important to establish a bound on the residual which captures the expected difference in behavior in the absence of faults. This bound, which is given in Proposition 2.3 below, will be useful as a threshold to be used by the supervisor in declaring when a fault has occurred and consequently when switching becomes necessary.

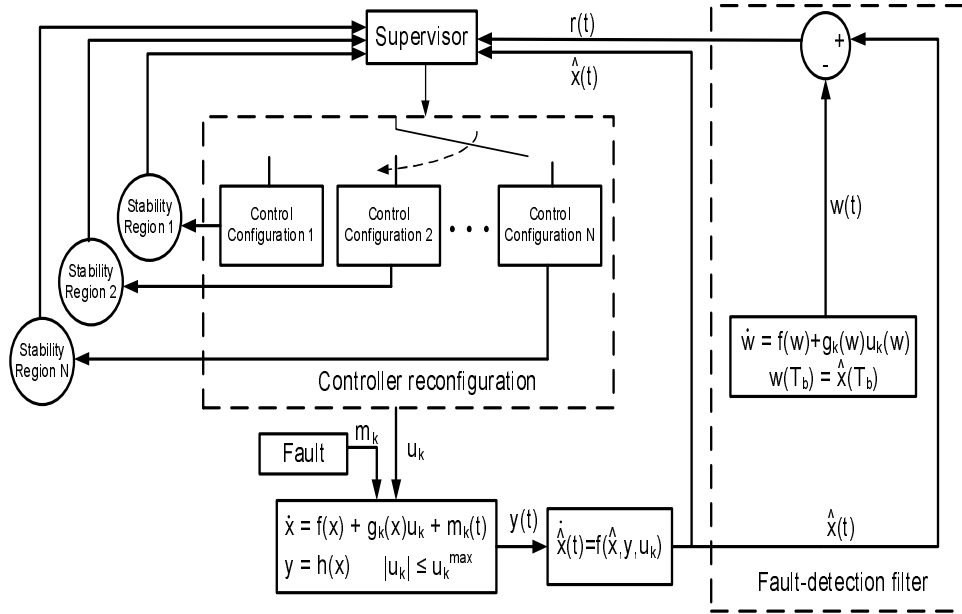


Figure 2.7: Integrated fault-detection and fault-tolerant control design under output feedback.

**Proposition 2.3:** Consider the nonlinear system of Eq.2.1, for a fixed mode,  $k(t) = i$ , and with  $m_i(t) \equiv 0$ , under the output feedback controller of Eq.2.17. Consider also the system of Eq.2.18. Then, given the set of positive real numbers  $\{\delta_{b,i}, \delta_{\zeta,i}, \delta_{m,i}, T_i^b\}$ ,

there exists a positive real number,  $\epsilon'_i > 0$ , such that if  $\epsilon_i \in (0, \epsilon'_i]$ ,  $V_i(x(0)) \leq \delta_{b,i}$ ,  $\|\tilde{y}(0)\| \leq \delta_{\zeta,i}$ ,  $w(T_i^b) = \hat{x}(T_i^b)$ , the residual satisfies a relation of the form  $r(t) \leq \delta_{m,i}$  for all  $t \geq T_i^b$ .

**Proof of Proposition 2.3:** Consider the system of Eq.2.1 with  $m_i(t) \equiv 0$  under the output feedback controller of Eq.2.17. From the result of Proposition 2.1, we have that given  $x(0) \in \Omega_{b,i}$  and any positive real number  $T_i^b$ , there exists a real positive number  $\epsilon_i^{**}$  such that  $\|x(t) - \hat{x}(t)\| \leq k_1 \epsilon_i$ , for all  $t \geq T_i^b$ ,  $\epsilon_i \in (0, \epsilon_i^{**}]$ , for some  $k_1 > 0$ , i.e.,  $x(t) = \hat{x}(t) + O(\epsilon_i)$ , where  $O(\epsilon_i)$  is the standard order of magnitude notation. Now, consider the following two systems for  $t \geq T_i^b$ :

$$\dot{x}(t) = f(x(t)) + g_i(x(t))u_i(\hat{x}(t)) \quad (2.19)$$

$$\dot{w}(t) = f(w(t)) + g_i(w(t))u_i(w(t)) \quad (2.20)$$

where  $w(T_i^b) = \hat{x}(T_i^b)$ . The system of Eq.2.20 is exactly the closed-loop system under full state feedback and has an asymptotically (and exponentially) stable equilibrium at the origin, for all initial conditions within  $\Omega_i$ . The system of Eq.2.19 is the closed-loop system under output feedback and (from Proposition 2.1) has an asymptotically (and locally exponentially) stable equilibrium at the origin, for all initial conditions within  $\Omega_{b,i} \subset \Omega_i$  and for all  $\epsilon_i \leq \epsilon_i^*$ . Since  $x(t) = \hat{x}(t) + O(\epsilon_i)$  for all  $t \geq T_i^b$ , we have that  $x(T_i^b) = \hat{x}(T_i^b) + O(\epsilon_i)$  and, when  $\epsilon_i = 0$ , the two systems of Eqs.2.19-2.20 become identical. Let  $F_i(\cdot) = f(\cdot) + g_i(\cdot)u_i(\cdot)$ , and  $x(T_i^b) = \hat{x}(T_i^b) + O(\epsilon_i) := \eta(\epsilon_i)$ , where  $\eta$  is a continuous function that depends smoothly on  $\epsilon_i$ , then we can write

$$\begin{aligned} \dot{x}(t) &= F_i(x(t), \epsilon_i), & x(T_i^b) &= \eta(\epsilon_i) \\ \dot{w}(t) &= F_i(w(t)), & w(T_i^b) &= \eta(0) \end{aligned} \quad (2.21)$$

It is clear from the above representation that the state equations for both the filter system and the closed-loop system, as well as their initial conditions at  $T_i^b$ , are identical when  $\epsilon_i = 0$ . Therefore, we can use the theory of regular perturbations (see

Chapter 8 in [49]) to establish the closeness of solutions between the two systems over the infinite time interval. In particular, since  $F_i(\cdot)$  is continuous and bounded on  $\Omega_{b,i}$ , and the  $w$ -system is exponentially stable, an application of the result of Theorem 8.2 in [49] yields that there exists  $\epsilon_i'' > 0$  such that for all  $\epsilon_i \in (0, \epsilon_i'']$ ,  $x(t) = w(t) + O(\epsilon_i)$  for all  $t \geq T_i^b$ . We therefore have that, for  $\epsilon_i \in (0, \min\{\epsilon_i^{**}, \epsilon_i''\}]$ ,  $r(t) = \|\hat{x}(t) - w(t)\| = \|\hat{x}(t) - x(t) + x(t) - w(t)\| \leq \|\hat{x}(t) - x(t)\| + \|x(t) - w(t)\| \leq (k_1 + k_2)\epsilon_i$  for all  $t \geq T_i^b$ . This implies that given any positive real number  $\delta_{m,i}$ , there exists  $\epsilon_i' > 0$  such that  $\|\hat{x}(t) - w(t)\| \leq \delta_{m,i}$  for all  $\epsilon_i \in (0, \epsilon_i']$ , for all  $t \geq T_i^b$ , where  $\epsilon_i' = \min\{\epsilon_i^{**}, \epsilon_i'', \delta_{m,i}/(k_1 + k_2)\}$ .

To summarize, we conclude that given the set of positive real numbers  $\{\delta_{b,i}, \delta_{\zeta,i}, \delta_{m,i}, T_i^b\}$ , there exists a positive real number,  $\epsilon_i' > 0$ , such that if  $\epsilon_i \in (0, \epsilon_i']$ ,  $V_i(x(0)) \leq \delta_{b,i}$ ,  $\|\tilde{y}(0)\| \leq \delta_{\zeta,i}$ ,  $w(T_i^b) = \hat{x}(T_i^b)$ , the residual satisfies a relation of the form  $r(t) \leq \delta_{m,i}$  for all  $t \geq T_i^b$ . This completes the proof of the proposition.

Note that the bound  $\delta_{m,i}$  can be chosen arbitrarily small by choosing the observer gain to be sufficiently large. Note also that, unlike the case of full state feedback, the fault-detection filter is initialized only after the passage of some short period of time,  $[0, T_i^b]$  (which can be chosen arbitrarily small by increasing the observer gain), to ensure that the closed-loop state estimates have converged sufficiently close to the true closed-loop states and thus – by setting the filter state  $w$  at this time equal to the value of the state estimate – ensure that the filter state is initialized sufficiently close to the true values of the state. From this point onwards, the filter simply integrates a replica of the dynamics of the process in the absence of errors. In the absence of actuator faults, the difference between the filter states and the process states is a function of the initial error, which can be bounded from above by a value that can be



made as small as desired by decreasing the initial error, which in turn can be done by appropriate choice of the observer parameters.

Having established a bound on the residual in the absence of faults, we are now ready to proceed with the design of the switching logic. To this end, consider the nonlinear system of Eq.2.1 where, for each control configuration, an output feedback controller of the form of Eq.2.17 is available and, given the desired output feedback stability regions  $\Omega_{b,i} \subset \Omega_i$ ,  $i = 1, \dots, N$ , as well as the desired values for  $\delta_{m,i}$ ,  $T_b^i$ , an appropriate observer gain has been determined (e.g.,  $\epsilon_i \leq \min\{\epsilon_i^*, \epsilon_i', \epsilon_i^{**}\}$  to guarantee both stability and satisfaction of the desired bound on the residual) and the sets  $\Omega_{s,i}$  (see Proposition 2.2) have been computed. The implementation of the fault-detection filter and fault-tolerant controller is described in Theorem 2.2 below.

**Theorem 2.2:** *Let  $k(0) = i$  for some  $i \in \mathcal{K}$ ,  $x(0) \in \Omega_{b,i}$ ,  $w(T_i^b) = \hat{x}(T_i^b)$ , and consider a fault for which  $r(T_i^s) \geq \delta_{m,i}$ , where  $T_i^s > T_i^b$  is the earliest time for which  $r(t) \geq \delta_{m,i}$ . Then under the switching rule*

$$k(t) = \left\{ \begin{array}{ll} i, & 0 \leq t < T_i^s \\ j \neq i, & t \geq T_i^s, \hat{x}(T_i^s) \in \Omega_j^s \end{array} \right\} \quad (2.22)$$

*the origin of the closed-loop system is asymptotically stable.*

**Proof of Theorem 2.2:** Consider the nonlinear system of Eq.2.1, under the output feedback controller of Eq.2.17, and the system of Eq.2.18, where  $k(0) = i$  for some  $i \in \mathcal{K}$ ,  $x(0) \in \Omega_{b,i}$ ,  $w(T_i^b) = \hat{x}(T_i^b)$ ,  $\epsilon_i \leq \min\{\epsilon_i^*, \epsilon_i', \epsilon_i^{**}\}$ , where  $\epsilon_i^*$ ,  $\epsilon_i^{**}$  were defined in Proposition 2.1 and  $\epsilon_i'$  was defined in Proposition 2.3. Since we consider only faults for which  $r(T_i^s) \geq \delta_m^i$ , where  $T_i^s > T_i^b$  is the earliest time for which  $r(t) \geq \delta_m^i$ , it follows that:

(a) in the absence of such faults, no switching takes place and configuration  $i$  is implemented for all times. Since  $x(0) \in \Omega_{b,i}$  and  $\epsilon_i \leq \epsilon_i^*$ , asymptotic closed-loop stability of the origin follows directly from Proposition 2.1.

(b) in the case that such faults take place, the earliest time a fault is detected is  $T_i^s > T_i^b$  and we have, from Eq.2.22, that  $k(t) = i$  for  $0 \leq t < T_i^s$ . From the stability of the  $i$ -th closed-loop system established in Proposition 2.1, we have that the closed-loop trajectory stays bounded within  $\Omega_{b,i}$  for  $0 \leq t < T_i^s$ . At time  $T_i^s$ , the supervisor switches to a control configuration  $j$  for which  $\hat{x}(T_i^s) \in \Omega_{s,j}$ . By design,  $\hat{x}(t) \in \Omega_{s,j} \implies x(t) \in \Omega_{b,j}$  for all  $t \geq T_i^s > T_i^b$ . From this point onwards, configuration  $j$  is implemented in the closed-loop system for all future times and, since  $x(T_i^s) \in \Omega_{b,j}$ , asymptotic closed-loop stability of the origin follows from the result of Proposition 2.1. This completes the proof of Theorem 2.2.

The design and implementation of the fault-detection filter and fault-tolerant controller proceed as follows:

1. Given the nonlinear process of Eq.2.1, identify the available control configurations,  $k = 1, \dots, N$ . For each configuration, design the output feedback controller of Eq.2.17, and for a given choice of the output feedback stability region,  $\Omega_{b,i}$ , determine a stabilizing observer gain,  $\epsilon_i^*$ .
2. Given any positive real numbers,  $\delta_{m,i}$  and  $T_i^b$ , determine the observer gain,  $\epsilon_i'$ , for which the maximum possible difference between the filter states and the state estimates, in the absence of faults, is less than the threshold  $\delta_{m,i}$  for all times greater than  $T_i^b$ .
3. Given the output feedback stability region,  $\Omega_{b,i}$ , determine the maximum error,  $e_{m,i}^*$ , and the set  $\Omega_{s,i}$  such that if  $\|x - \hat{x}\| \leq e_{m,i} \leq e_{m,i}^*$  (i.e., the error between the estimates and the true values of the states is less than  $e_{m,i}$ ) and  $\hat{x} \in \Omega_{s,i}$  (i.e., the state estimates belong to  $\Omega_{s,i}$ ), then  $x \in \Omega_{b,i}$  (i.e., the state belongs to the output feedback stability region).

4. For a choice of  $e_{m,i} \in (0, e_{m,i}^*]$  and given  $T_i^b$ , determine the observer gain,  $\epsilon_i^{**}$ , for which the maximum possible difference between the states and the state estimates, in the absence of faults, is less than the threshold  $e_{m,i}$  for all times greater than  $T_i^b$ . Set  $\epsilon_i := \min\{\epsilon_i^*, \epsilon_i', \epsilon_i^{**}\}$ . Note that this choice guarantees that by time  $T_i^b$ : (1) the residual is within the desired threshold and (2) the presence of  $\hat{x}$  within  $\Omega_{s,i}$  guarantees that  $x$  belongs to  $\Omega_{b,i}$ .
5. Initialize the closed-loop system such that  $x(0) \in \Omega_{b,i}$ , for some  $i \in \mathcal{K}$ , and start generating the state estimates  $\hat{x}(t)$ . At time  $T_i^b$ , initialize and start integrating the filter dynamics of Eq.2.18 with  $w(T_i^b) = \hat{x}(T_i^b)$ , where  $\hat{x}$  is the state estimate generated by the high-gain observer.
6. At the earliest time  $T_i^s > T_i^b$  that  $r(t) > \delta_{m,i}$  (implying that the difference between the expected evolution of the process states and the estimates of the process states is more than what can be accounted for by the error in the initialization of the filter states, implying that a fault has occurred), activate the backup configuration for which  $\hat{x}(T_i^s) \in \Omega_{s,j}$  (note that since  $t = T_i^s > T_i^b$ , we have that  $\|x(T_i^s) - \hat{x}(T_i^s)\| \leq e_{m,i}$ ; this together with  $\hat{x}(T_i^s) \in \Omega_{s,j}$  implies that  $x(T_i^s) \in \Omega_{b,j}$ , i.e., the state belongs to the stability region of configuration  $j$ ). Implement the backup configuration  $j$  to achieve closed-loop stability.

Theorem 2.2 considers faults that are “observable” from the filter’s residual, in the sense that if the residual in Eq.2.18 exceeds the allowable threshold  $\delta_{m,i}$  at any time, then the supervisor can conclude with certainty that a fault has occurred. On the other hand, if the residual does not exceed the allowable threshold, it might still be possible that some “unobservable” fault – the effect of which is within the filter threshold – has taken place. Note that in contrast to the case of full state feedback, the states in this case are only known up to a certain degree of accuracy.

Therefore, any fault that causes a difference in the closed-loop behavior that is within that margin of (i.e., indistinguishable from) the effect of the estimation error will, in principle, go undetected. This class of faults is not considered in Theorem 2.2 since its effect on closed-loop stability cannot be discerned from the behavior of the residual. This, however, is not a restriction since the observability threshold  $\delta_{m,i}$  is a design parameter and can be chosen arbitrarily small, thus rendering the possibility of major (i.e., destabilizing) faults that cannot be detected quite small. Ultimately, the choice of  $\delta_{m,i}$  reflects a fundamental tradeoff between the need to avoid false alarms that could be caused by estimation errors (this favors a relatively large threshold) and the need to minimize the possibility of some faults going undetected (this favors a relatively small threshold).

Note that for all times prior to  $T_i^b$ , the filter is inactive. Up-until this time, the state estimates have not yet converged close enough to the true values of the states, and no inference about the state of the system can be drawn by looking at the evolution of the state estimate, and therefore no inference about any possible faults can be drawn via the fault-detection filter. If a fault occurs within this time, the filter will detect its occurrence only after the time  $T_i^b$ . By choosing a larger value of the observer gain, however, the time  $T_i^b$  can be reduced further, if so desired. Note also that while we consider the problem of unavailability of some of the state variables as measurements, we do not consider the problem of sensor faults, i.e., we assume that the sensors do not malfunction both in the state and output feedback cases. In the event of availability of multiple measurements in a way that each of them can be used to estimate of the process states, the estimates of the states generated using the different measurements can be used to also detect sensor faults.

**Remark 2.5:** The central idea behind the model-based fault-detection filter design,

that of comparing the evolution of the process to the expected evolution of the process in the absence of faults, can also be used to design a rule-based fault-detection filter. One example of a rule-based fault-detection filter is to declare a fault if the state estimates, after a time  $T_i^b$ , touch the boundary of  $\Omega_{s,i}$ , indicating that the closed-loop states themselves may be about to escape the output feedback stability region  $\Omega_{b,i}$ . The rule-based fault detection filter, however, would be able to detect the fault only when the state estimates hit the boundary of  $\Omega_{s,i}$ , as opposed to the model-based fault detection filter, which detects a fault as soon as the effect of the fault on the closed-loop evolution goes beyond a prescribed threshold. This delay in a rule-based approach could result in the state escaping the stability region of the available backup configurations (see the simulation for an example). Also, it may happen that the fault causes the closed-loop process states evolving within  $\Omega_{s,i}$  to neither escape  $\Omega_{s,i}$  nor converge to the origin. The rule based fault-detection filter would not be able to detect such a fault. In contrast, the model-based fault-detection filter of Theorem 2.2, is able to detect faults that have an effect, up-to a desirable threshold, on the evolution of the closed-loop process. Note also that the model-based fault-detection filter of Theorem 2.2 and the rule-based fault-detection filter discussed above differ only in that the model-based filter of Theorem 2.2 uses a more quantitative knowledge of the closed-loop dynamics to predict the expected closed-loop trajectory, instead of using the qualitative knowledge that the fault-free closed-loop state trajectory does not escape the stability region.

### 2.4.3 Simulation results

In this subsection, we first illustrate the implementation of the proposed fault-tolerant control methodology to the chemical reactor introduced as a motivating example to clearly explain the main ideas behind the application of the proposed fault-detection

and fault-tolerant control method, and then demonstrate an application to a networked chemical reactor example, investigating issues such as uncertainty and measurement noise.

For the chemical reactor of the motivating example, Fig.2.11 depicts the stability region, in the  $(T, C_A)$  space, for each configuration. The desired steady-state is depicted with an asterisk that lies in the intersection of the three stability regions. For the first two control configurations, a state estimator of the form of Eq.2.17 is designed. For thresholds of  $\delta_m = 0.0172$  and  $0.00151$  in the fault detection filters, the parameters in the observer of Eq.2.17 are chosen as  $L_1 = L_2 = 100$ ,  $a_1^{(1)} = a_1^{(2)} = 10$  and  $a_2^{(1)} = a_2^{(2)} = 20$ . For the third configuration, the estimates,  $\hat{T}$ ,  $\hat{C}_A$  are generated as follows:

$$\begin{aligned} \frac{d\hat{T}}{dt} &= \frac{F}{V}(T_{A0} - \hat{T}) + \sum_{i=1}^3 \frac{(-\Delta H_i)}{\rho c_p} k_{i0} e^{\frac{-E_i}{R\hat{T}}} \hat{C}_A + \alpha_1(C_A - \hat{C}_A) \\ \frac{d\hat{C}_A}{dt} &= \frac{F}{V}(C_{A0} - \hat{C}_A) - \sum_{i=1}^3 k_{i0} e^{\frac{-E_i}{R\hat{T}}} \hat{C}_A + \alpha_2(C_A - \hat{C}_A) \end{aligned} \quad (2.23)$$

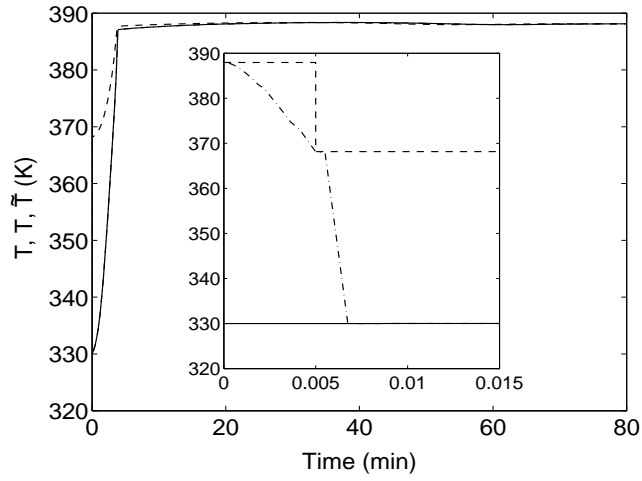
where  $\alpha_1 = -10^4$  and  $\alpha_2 = 10$ . The reactor is initialized at  $T(0) = 330 \text{ K}$ ,  $C_A(0) = 3.6 \text{ kmol/m}^3$ ,  $C_B(0) = 0.0 \text{ kmol/m}^3$ , using the  $Q$ -control configuration, while the state estimates are initialized at  $\hat{T}(0) = 390 \text{ K}$ ,  $\hat{C}_A(0) = 3.6 \text{ kmol/m}^3$  and the supervisor proceeds to monitor the evolution of the closed-loop estimates.

We first demonstrate the need to wait for a sufficient time before initializing the filter. To this end, consider the fault-detection filter initialized at  $t = 0.005$  minutes  $\equiv T_1^b$  at which time the state estimates (dash-dotted lines in Fig.2.8) have not converged to the true values (solid lines in Fig.2.8). As a result, the fault-detection filter shows a false alarm (see Fig.2.9a) by crossing the threshold even when control configuration 1 is functioning properly (see Fig.2.9b) and stabilizes the closed-loop system. Note that while the initialization of the filter at a time when the state estimates have not

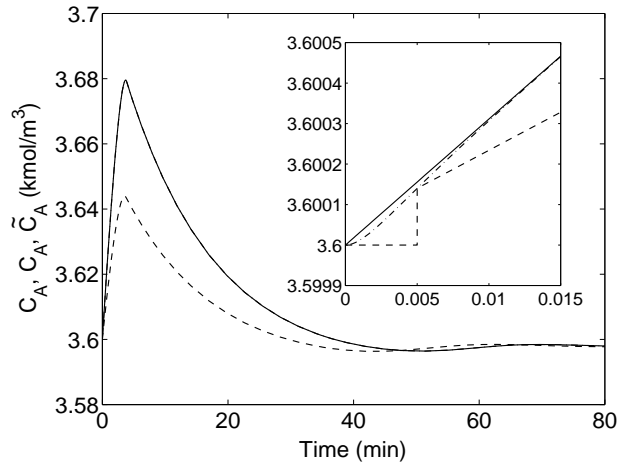
converged leads to the residual crossing the threshold, the residual eventually goes to zero as expected, since both the filter states and the closed-loop process states eventually stabilize and go to the same equilibrium point.

We now demonstrate the application of the fault-detection filter and fault-tolerant controller of Theorem 2.2. Starting from the same initial conditions, the estimates of  $T$  and  $C_A$  (dash-dotted lines in Figs.2.10a,b) converge very quickly to the true values of the states (solid lines in Figs.2.10a,b). The states in the fault-detection filter are initialized and set equal to the value of the state estimates at  $t = 0.01$  minutes  $\equiv T_1^b$ ; note that by this time the estimates have converged to the true values. By initializing the fault-detection filter appropriately, a false alarm is prevented (the value of  $r_1(t)$  does not hit the threshold in the absence of a fault after a time  $t = 0.01$  minutes, see Fig.2.12a). As shown by the solid lines in Fig.2.11, the controller proceeds to drive the closed-loop trajectory towards the desired steady-state, up until the  $Q$ -configuration fails after 3.0 minutes  $\equiv T_1^f$  of reactor startup (see solid lines in Fig.2.14a). Note that at this time, the value of  $r_1(t)$  becomes non-zero and hits the threshold at  $t = 3.3$  minutes  $\equiv T_1^s$ . From Fig.2.11, 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 of Eq.2.22, the supervisor activates the second configuration (with  $T_{A0}$  as the manipulated input). The result is shown by the solid line in Fig.2.11 where it is seen that upon switching to the  $T_{A0}$ -configuration, the corresponding controller continues to drive the state trajectory closer to the desired steady-state.

When a second failure occurs (this time in the  $T_{A0}$ -configuration) at  $t = 13.0$  minutes  $\equiv T_2^f$  (which is simulated by fixing  $T_{A0}$  for all  $t \geq 13.0$  minutes, see solid



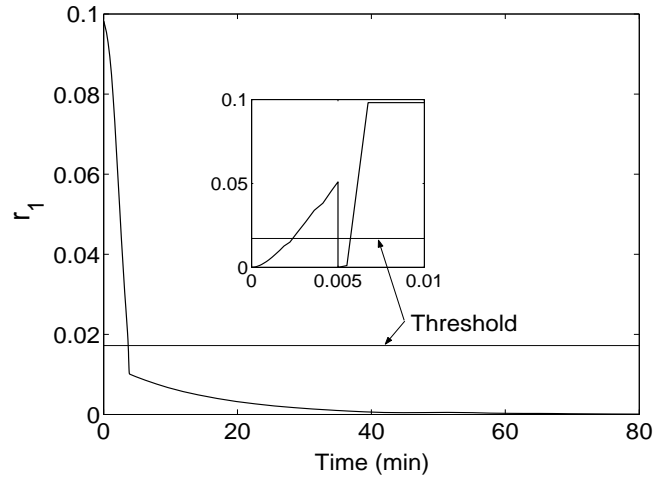
(a)



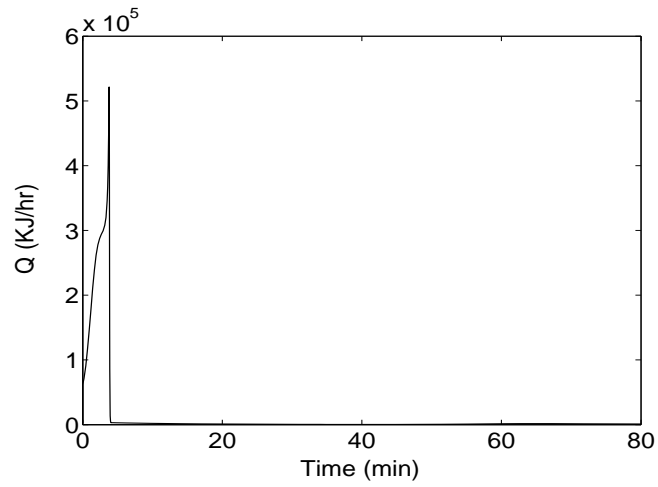
(b)

Figure 2.8: Evolution of the closed-loop (a) temperature (solid line), estimate of temperature (dash-dotted line) and the temperature profile generated by the filter (dashed line) and (b) concentration (solid line), estimate of concentration (dash-dotted line) and the concentration profile generated by the filter (dashed line) under control configuration 1 when the fault detection filter is initialized at  $t = 0.005$  minutes.



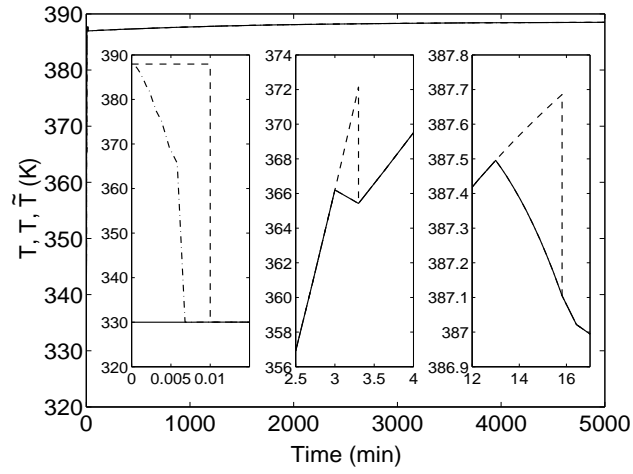


(a)

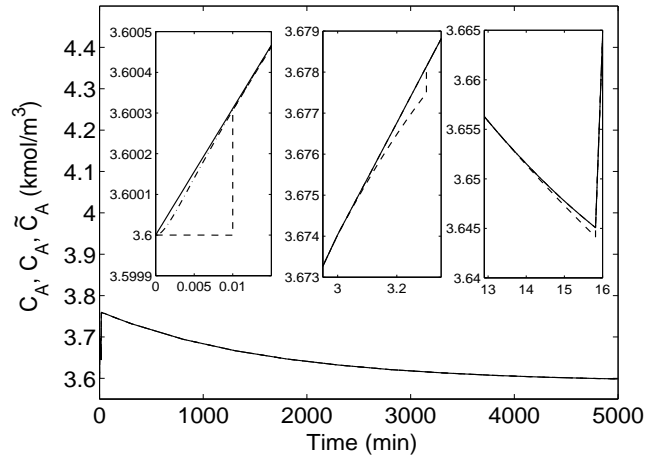


(b)

Figure 2.9: Evolution of (a) the residual and (b) the manipulated input profile for the first control configuration when the fault detection filter is initialized at  $t = 0.005$  minutes.



(a)



(b)

Figure 2.10: Evolution of the closed-loop (a) temperature (solid line), estimate of temperature (dash-dotted line) and the temperature profile generated by the filter (dashed line) and (b) concentration (solid line), estimate of concentration (dash-dotted line) and the concentration profile generated by the filter (dashed line) under the switching rule of Eq.2.22 subject to failures in control systems 1 and 2.

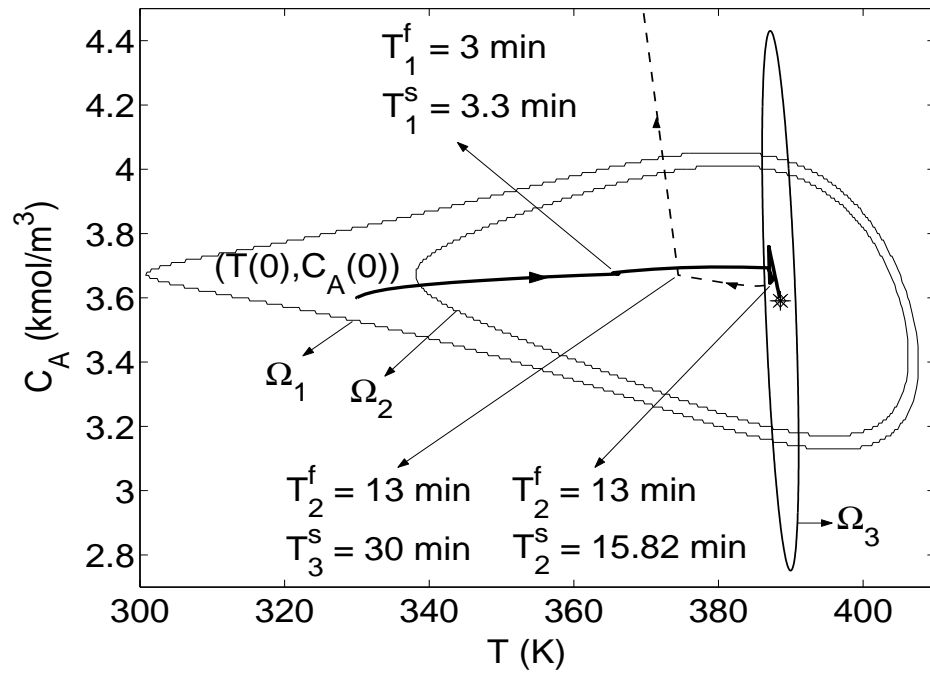
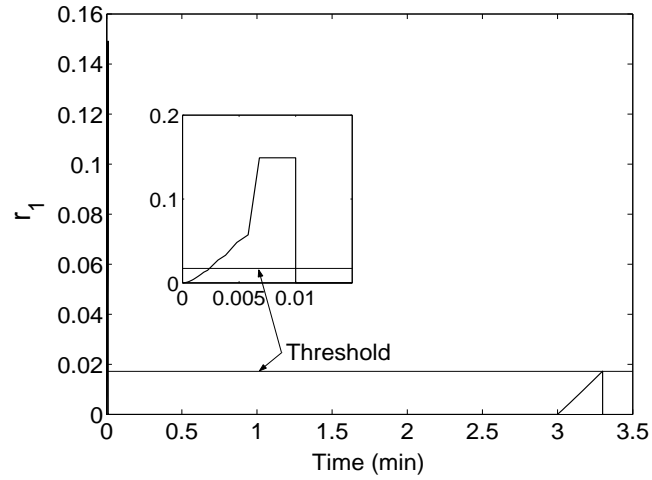
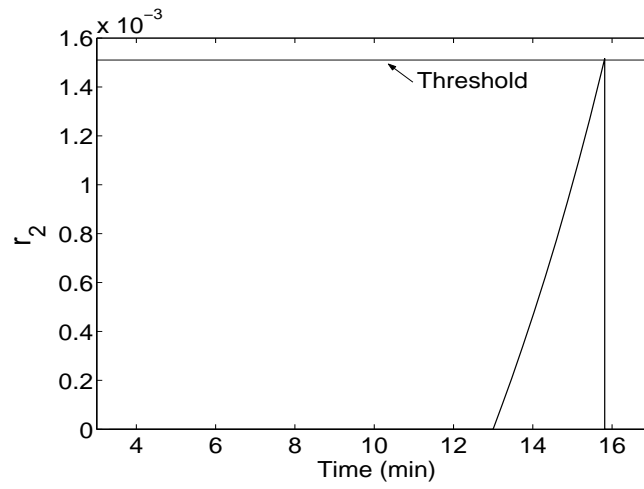


Figure 2.11: Evolution of the closed-loop state trajectory under the switching rule of Eq.2.22 subject to failures in control systems 1 and 2, using an appropriate fault-detection filter (solid line) and in the absence of a fault-detection filter (dashed line).

lines in Fig.2.14b) before the process has reached the steady state, the filter detects this failure via the value of  $r_2(t)$  hitting the threshold (see Fig.2.12b). From the solid line in Fig.2.11, it is clear that the failure of the second control configuration occurs when the closed-loop trajectory is within the stability region of the third configuration. However, if the fault-detection filter is not in place and the backup configuration is implemented late in the closed-loop (at  $t = 30$  minutes  $\equiv T_3^s$ ), by this time the state of the closed-loop system has moved out of the stability region of the third control configuration, and closed-loop stability is not achieved (see dashed line in Fig.2.11, see also Fig.2.13 and dashed lines in Fig.2.14). In contrast, when the fault-detection filter is in place, it detects a fault at  $t = 15.82$  minutes  $\equiv T_2^s$  and when the supervisor switches to configuration 3, closed-loop stability is achieved (see solid line in Fig.2.11).

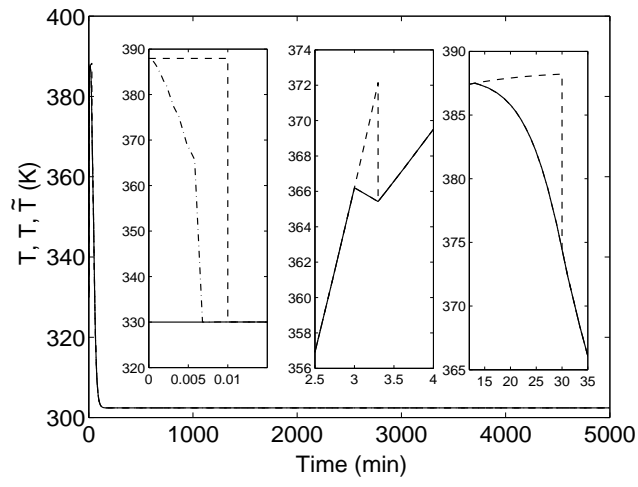


(a)

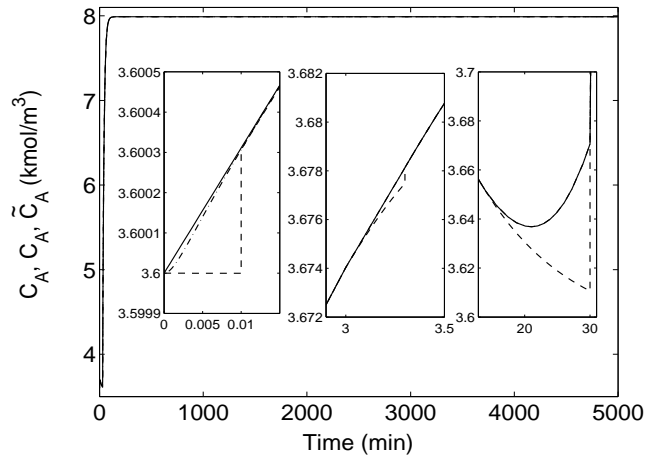


(b)

Figure 2.12: Evolution of the residual for (a) the first control configuration and (b) the second control configuration.

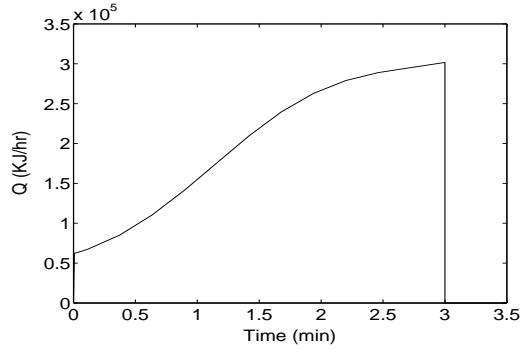


(a)

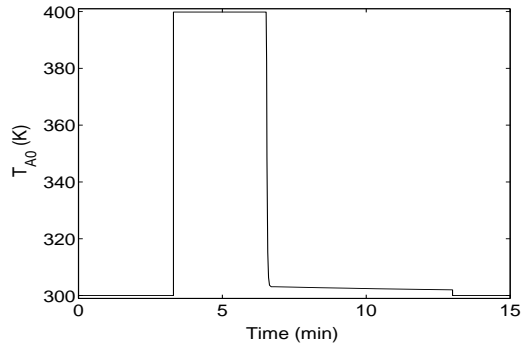


(b)

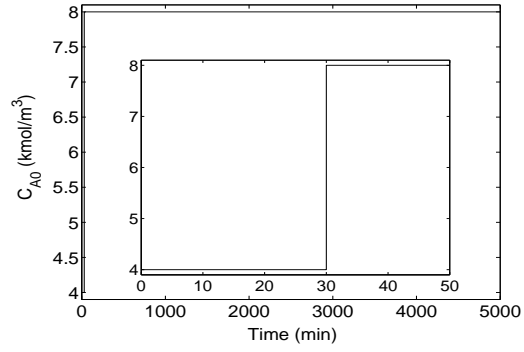
Figure 2.13: Evolution of the closed-loop (a) temperature (solid line), estimate of temperature (dash-dotted line) and the temperature profile generated by the filter (dashed line) and (b) concentration (solid line), estimate of concentration (dash-dotted line) and the concentration profile generated by the filter (dashed line) under the switching rule of Eq.2.22 subject to failures in control systems 1 and 2 in the absence of a fault-detection filter.



(a)



(b)



(c)

Figure 2.14: Manipulated input profiles under (a) control configuration 1, (b) control configuration 2, and (c) control configuration 3 under the switching rule of Eq.2.22 subject to failures in control systems 1 and 2 in the presence (solid lines) and absence (dashed lines) of a fault-detection filter.

## Chapter 3

# Integrated fault-detection and isolation and fault-tolerant control

### 3.1 Introduction

This chapter considers the problem of implementing fault tolerant control on a multi-input multi-output nonlinear system subject to multiple faults in the control actuators and constraints on the manipulated inputs. To illustrate some of the ideas behind the design of the fault-detection and isolation filter and subsequent reconfiguration, the case where all the states of the system are measured is first considered. The state measurements and the model are used to design filters that essentially capture the difference between the fault-free evolution and the evolution of the system to detect and isolate faults. Once a fault is detected and isolated, out of the available backup configurations, a configuration is chosen that 1) does not use the failed control actuator, and 2) guarantees the stability of the closed-loop system starting from the system state at the time of the failure. To be able to ascertain the second condition, Lyapunov-based controllers are used in designing the control laws for the individual control configurations which provide an explicit characterization of the set of ini-



tial conditions starting from where the closed-loop stability is guaranteed. The more complicated and realistic problem where all the system states are not measured is considered next. First, output-feedback controllers are designed that use a combination of state estimators and state-feedback controllers in a way that allows for an explicit characterization of the output-feedback stability region. The state estimates are employed in the design of the fault-detection and isolation filters, and also in devising the reconfiguration rule that determines which of the backup control configurations should be implemented in the closed-loop system. Finally, the implementation of the fault-detection and isolation filters and reconfiguration strategy is first illustrated via a state-feedback chemical reactor example, and then issues such as uncertainty, measurement noise, and applicability in an output-feedback setting are investigated in further chemical reactor examples.

### 3.2 Preliminaries

We consider nonlinear systems with input constraints, described by:

$$\begin{aligned} \dot{x} &= f(x) + G_{k(t)}(x)(u_{k(t)}(x) + \tilde{u}_{k(t)}(t)), & y(x) &= h(x) \\ u_k &\in \mathbf{U}_k, & k(t) &\in \mathcal{K} = \{1, \dots, N\}, & N < \infty \end{aligned} \tag{3.1}$$

where  $x \in \mathbb{R}^n$  denotes the vector of state variables,  $y \in \mathbb{R}^m$  denotes the vector of measured variables and  $u_{k(t)}(x) \in \mathbb{R}^m$  denotes the control action prescribed by the control law for the vector of constrained manipulated inputs under the  $k$ th configuration.  $\tilde{u}_{k(t)}$  denotes the unknown fault vector with  $u_{k(t)}(x) + \tilde{u}_{k(t)}$  taking values in a nonempty convex subset  $\mathbf{U}_k$  of  $\mathbb{R}^m$ , where  $\mathbf{U}_k = \{u_k + \tilde{u}_k \in \mathbb{R}^m : \|u_k + \tilde{u}_k\| \leq u_k^{max}\}$ ,  $\|\cdot\|$  is the Euclidean norm of a vector,  $u_k^{max} > 0$  is the magnitude of input constraints and  $f(0) = 0$ . The vector function  $f(x)$  and the matrices  $G_k(x) = [g_{1,k}(x) \cdots g_{m,k}(x)]$  are assumed to be sufficiently smooth on their domains of definition.  $k(t)$ , which takes values in the finite index set  $\mathcal{K}$ , represents a discrete state that indexes the

matrix  $G_k(\cdot)$  as well as the manipulated input  $u_k(\cdot)$  and the possible faults in the manipulated inputs  $\tilde{u}_k(\cdot)$ . For each value that  $k$  assumes in  $\mathcal{K}$ , the process is controlled via a different set of manipulated inputs which defines a given control configuration. The notation  $L_f h$  denotes the standard Lie derivative of a scalar function  $h(\cdot)$  with respect to the vector function  $f(\cdot)$  and the notation  $x(T^+)$  denotes the limit of the trajectory  $x(t)$  as  $T$  is approached from the right, i.e.,  $x(T^+) = \lim_{t \rightarrow T^+} x(t)$ . Throughout the manuscript, we assume that for any  $u_k \in \mathbf{U}_k$  the solution of the system of Eq.3.1 exists and is continuous for all  $t$ .

To illustrate some of the ideas behind the fault detection and isolation filter design and reconfiguration strategy, we begin by assuming that all the states are available as measurements. We next review one example of a state-feedback controller that provides an explicit estimate of the stability region for the closed-loop system subject to constraints (for more details on the controller design, and the proof, see [25] and [54]).

**Theorem 3.1 [25]:** *Consider the switched nonlinear system of Eq.3.1 for a configuration  $k$  for which a Control Lyapunov Function  $V_k$  exists, with  $\tilde{u}_k(t) \equiv 0$ , under state-feedback using the following bounded nonlinear feedback controller:*

$$u_k = -w_k(x, u_k^{max})(L_{G_k} V_k(x))^T \quad (3.2)$$

where  $w_k(x, u_k^{max}) =$

$$\begin{cases} \frac{\alpha_k(x) + \sqrt{\alpha_k^2(x) + (u_k^{max} \|b_k^T(x)\|)^4}}{\|b_k^T(x)\|^2 \left[1 + \sqrt{1 + (u_k^{max} \|b_k^T(x)\|)^2}\right]}, & b_k^T(x) \neq 0 \\ 0, & b_k^T(x) = 0 \end{cases} \quad (3.3)$$

with  $\alpha_k(x) = L_{f_k} V_k(x) + \rho_k V_k(x)$ ,  $\rho_k > 0$  and  $b_k(x) = L_{G_k} V_k(x)$ . Assume that the set  $\Phi_k(u_k^{max})$  of  $x$  satisfying

$$L_{f_k} V_k(x) + \rho_k V_k(x) \leq u_k^{max} \|(L_{G_k} V_k(x))^T\| \quad (3.4)$$

contains the origin and a neighborhood of the origin. Also, let  $\Omega_k(u_k^{max}) := \{x \in \mathbb{R}^n : V_k(x) \leq c_k^{max}\}$  be a level set of  $V_k$ , completely contained in  $\Phi_k$ , for some  $c_k^{max} > 0$ . Then for all  $x(0) \in \Omega_k(u_k^{max})$  the control law guarantees that the origin of the closed-loop system is asymptotically stable.

### 3.3 State-feedback Fault-tolerant control

In this section, we first consider the problem under state-feedback to illustrate the main idea behind the fault detection and isolation filter and fault-tolerant controller design.

#### 3.3.1 State-feedback fault detection and isolation filter

To be able to detect the occurrence of a fault in a control actuator via observing the state evolution, it is necessary that the control actuator influences the evolution of at least some of the states. To be able to isolate the occurrence of a fault, it becomes further necessary that the control actuator in question be the only one influencing at least some state. To understand this better, consider the following single state, two input example:  $\dot{x} = x + u_1(x) + \tilde{u}_1 + u_2(x) + \tilde{u}_2$ . As is clear from the equation, the faults in the manipulated inputs  $u_1$  and  $u_2$  effect the evolution of the state additively, i.e., as the sum  $(\tilde{u}_1 + \tilde{u}_2)$ . While it may be possible to detect that a fault has occurred in either  $u_1$  or  $u_2$  (if the faults do not cancel out each other, i.e., if  $\tilde{u}_1 + \tilde{u}_2 \neq 0$ ), it is not possible, in this case, to determine by observing the evolution of the system (and finding it to be different when compared to the expected evolution with  $\tilde{u}_1 = \tilde{u}_2 = 0$ ) whether  $\tilde{u}_1 \neq 0$  or  $\tilde{u}_2 \neq 0$ , or both. In other words, while it may be possible to detect the occurrence of a fault, it is not possible to isolate it. Below we formulate a verifiable assumption on the structure of the system of Eq.3.1 that allows for fault

detection and isolation.

**Assumption 3.1:** Consider the system of Eq.3.1 in configuration  $k$  under state-feedback. Then for every input  $u_{j,k}$ ,  $j = 1, \dots, m$ , there exists a unique state  $x_{i,k}$ ,  $i \in \{1, \dots, n\}$  such that with  $x_{i,k}$  as output, the relative degree of  $x_{i,k}$  with respect to  $u_{j,k}$  and only with respect to  $u_{j,k}$  is equal to 1.

Consider now the system of Eq.3.1 in configuration  $k$  for which Assumption 3.1 holds. Theorem 3.2 below formulates the fault detection and isolation filter and outlines its fault detection and isolation properties.

**Theorem 3.2:** Consider the system of Eq.3.1 in configuration  $k$  under the control law of Eq.3.2. Let the fault detection and isolation filter for the  $j$ th manipulated input in the  $k$ th configuration be described by

$$\begin{aligned}\dot{\tilde{x}}_{i,k} &= f_i(x_1, \dots, \tilde{x}_{i,k}, \dots, x_n) + g_{j,k}[i](x_1, \dots, \tilde{x}_{i,k}, \dots, x_n)u_{j,k}(x_1, \dots, \tilde{x}_{i,k}, \dots, x_n) \\ e_{i,k} &= \tilde{x}_{i,k} - x_i\end{aligned}\tag{3.5}$$

where  $g_{j,k}[i]$  denotes the  $i$ th element of the vector  $g_{j,k}$ ,  $\tilde{x}_{i,k}(0) = x_i(0)$  and the subscripts  $i, k$  refer to the  $i$ th state under the  $k$ th control configuration. Let  $T_{j,k}^f$  be the earliest time for which  $\tilde{u}_{j,k} \neq 0$ , then the fault detection and isolation filter of Eq.3.5 ensures that  $e_{i,k}(T_{j,k}^{f+}) \neq 0$ . Also,  $e_{i,k}(t) \neq 0$  only if  $\tilde{u}_{j,k}(s) \neq 0$  for some  $0 \leq s < t$ .

**Proof of Theorem 3.2:**

*Part 1:* We first show the *only if* part of the Theorem by contradiction. To this end, consider the equation describing the evolution of the  $i$ th state,  $x_i$  described by

$$\dot{x}_i = f_i(x) + g_{j,k}[i](x)(u_{j,k}(x) + \tilde{u}_{j,k}(t))\tag{3.6}$$

and let us assume that  $\tilde{u}_{j,k}(s) = 0$ , for all  $0 \leq s < t$ . Then for all  $0 \leq s < t$  Eq.3.6 reduces to

$$\dot{x}_i = f_i(x) + g_{j,k}[i](x)u_{j,k}(x)\tag{3.7}$$

Since  $x_i(0) = \tilde{x}_{i,k}(0)$ , we therefore have that  $\dot{x}_i(s) = \dot{\tilde{x}}_{i,k}(s)$  for  $s = 0$  and subsequently

for all  $0 \leq s < t$ . Therefore  $e_{i,k}(s) = 0$  for all  $0 \leq s < t$ , which leads to a contradiction. This means that the assumption that  $\tilde{u}_{j,k}(s) = 0$ , for all  $0 \leq s < t$  does not hold, i.e.,  $\tilde{u}_{j,k}(s) \neq 0$  for some  $0 \leq s < t$ . This completes the proof of the first part of the theorem.

*Part 2:* To prove the *if* part of the theorem, consider once again Eq.3.5 and Eq.3.6 with  $\tilde{u}_j^k(t) = 0$  for all  $t \leq T_k^f$ . Then following the line of reasoning as in Part 1, we get that  $x_i(T_{j,k}^f) = \tilde{x}_{i,k}(T_{j,k}^f)$ . Since  $\tilde{u}_{j,k}(T_{j,k}^f) \neq 0$ , we get that  $\dot{x}_i(T_{j,k}^f) \neq \dot{\tilde{x}}_{i,k}(T_{j,k}^f)$ , and therefore, that  $x_i(T_{j,k}^{f+}) \neq \tilde{x}_{i,k}(T_{j,k}^{f+})$ , i.e.,  $e_{i,k}(T_{j,k}^{f+}) \neq 0$ . This completes the proof of Theorem 3.2.

**Remark 3.1:** As stated in Theorem 3.2 above, the fault detection and isolation filter performs the task of detection as well as isolation. Specifically, the *if* part of the theorem characterizes the detection capabilities where the residual for a manipulated input becomes non-zero if a fault occurs in the given manipulated input. The *only if* part of the theorem allows isolation since a residual is non-zero only if a fault has occurred at some previous time in the given manipulated input. Note that in general it is possible that a fault-occurs for some time and disappears, and also the fault profile is such that after some time the evolution of the system becomes identical again to the fault-free system, in which case the residual would once again go back to zero. The immediate detection capability of the filter above, however, precludes the possibility that such a fault goes undetected.

**Remark 3.2:** Note that Assumption 3.1 is a sufficient condition that allows fault detection and isolation filter design, and can be readily relaxed. For instance, if the inputs influence the evolution of the states in an ‘upper triangular’ or ‘lower triangular’ form, fault detection and isolation is possible using the same idea as in Theorem 3.2 above. As an illustration, consider a two state two input system, of the

form

$$\begin{aligned}\dot{x}_1 &= f_1(x) + g_1[1](x)(u_1(x) + \tilde{u}_1(t)) \\ \dot{x}_2 &= f_2(x) + g_1[2](x)(u_1(x) + \tilde{u}_1(t)) + g_2[2](x)(u_2(x) + \tilde{u}_2(t))\end{aligned}\quad (3.8)$$

where  $f_i(\cdot)$  denotes the  $i$ th elements of the vector function  $f(\cdot)$  and  $g_i[j]$  denotes the  $j$ th element of the vector  $g_i$ . While this system does not satisfy Assumption 3.1, fault detection and isolation can still be achieved. Specifically, a filter design of the form of Eq.3.5 can be used to build a detection filter for the first manipulated input. The dynamics of the second filter can then be designed as

$$\begin{aligned}\dot{\tilde{x}}_2 &= f_2(x_1, \tilde{x}_2) + g_1[2](x_1, \tilde{x}_2)(u_1(x_1, \tilde{x}_2)) + g_2[2](x_1, \tilde{x}_2)(u_2(x_1, \tilde{x}_2)) \\ e_2 &= \tilde{x}_2 - x_2\end{aligned}\quad (3.9)$$

In this setup, faults in  $u_1$  will be captured in both  $e_1$  and  $e_2$ , while faults in  $u_2$  will only effect  $e_2$ . The task of fault detection and isolation can therefore be carried out via a simple process of elimination.

**Remark 3.3:** Even in cases where the structure of the process dynamic model does not allow for complete isolation of a fault (i.e., more than one manipulated input has a relative degree one with respect to a given state), the proposed method can still isolate the failure to a subset of the entire group of active manipulated inputs. This would be especially useful in the case of high-dimensional process systems with a large number of states and inputs where several redundant inputs are used simultaneously. However, once a subset of control actuators including the failed ones has been identified by the filter, nothing can be said about which actuator(s) of the ones in this subset has actually failed. Therefore, in order to guarantee stability in the controller reconfiguration phase, the worst case scenario, where all the actuators in this subset have failed, must be assumed and the supervisor must then switch to a fallback configuration that does not implement any of the control actuators included in this subset.

### 3.3.2 State-feedback fault-tolerant controller

Given that a fault is detected and isolated using the filters designed in the previous section, the problem that we address in this section is that of determining an appropriate backup configuration. The first requirement for an appropriate backup control configuration is that it does not use the faulty control actuator. Secondly, the limitations imposed by the presence of input constraints must be accounted for, and in particular, a backup configuration should be implemented for which the state of the closed-loop system resides in its stability region. This idea is formalized in Theorem 3.3 below.

**Theorem 3.3:** *Consider the closed-loop system of Eqs.3.1-3.2 under state-feedback and let  $x(0) := x_0 \in \Omega_{k_0}$  for some  $k_0 \in \mathcal{K}$ . Let  $T_{j,k_0}$  be the earliest time such that  $e_{i,k_0} \neq 0$  for some  $i$  corresponding to a manipulated input  $u_{j,k_0}$  in Eq.3.5. Then the following switching rule:*

$$k(t) = \left\{ \begin{array}{ll} k_0, & 0 \leq t < T_{j,k_0} \\ q \neq k_0, & t \geq T_{j,k_0}, x(T_{j,k_0}) \in \Omega_q, \\ & u_{j,k_0} \notin u_q \end{array} \right\} \quad (3.10)$$

*guarantees asymptotic stability of the origin of the closed-loop system.*

**Proof of Theorem 3.3:** We consider the two cases, 1)  $e_{i,k_0}(t) = 0$  for all  $t \geq 0$  for all  $i \in \{1, \dots, n\}$  and 2)  $e_{i,k_0}(t) \neq 0$  for some  $T_{j,k_0}$  for some  $j \in \{1, \dots, m\}$ .

*Case 1:*  $e_{i,k_0}(t) = 0 \forall t \geq 0$  for all  $j \in \{1, \dots, m\}$  implies (using Theorem 3.2) that  $\tilde{u}_{j,k}(t) = 0$  for all  $t \geq 0$  and for all  $j \in \{1, \dots, m\}$ . The switching rule of Eq.3.10 then dictates that  $k(t) = k_0 \forall t \geq 0$ . Since  $x(0) \in \Omega_{k_0}$ , asymptotic stability of the origin of the closed-loop system follows from Theorem 3.1.

*Case 2:* If  $e_{i,k_0}(t) \neq 0$  for some  $T_{j,k_0}$  for some  $j \in \{1, \dots, m\}$ , the switching rule dictates switching to configuration  $q$  such that  $x(T_{j,k_0}) \in \Omega_q$ . Closed-loop stability of

the origin of the closed-loop system again follows from Theorem 3.1. This completes the proof of Theorem 3.3.

**Remark 3.4:** Early detection of a fault enhances the chances that corrective action can be taken in time to achieve fault-tolerant control. Specifically, it may happen that a fault occurs when the closed-loop state resides in the stability region of one of the backup configurations, but the destabilizing effect of the fault may drive the state outside the stability region of the backup configuration by the time the fault is detected. Theorem 3.2 guarantees that a fault is detected as soon as it occurs. Note also that in the presence of plant model mismatch or unknown disturbances, the value of  $e_{i,k}(t)$  will be nonzero even in the absence of faults. The presence of time varying disturbances  $\theta(t)$  with known bounds  $\theta_b$  on the disturbances can be accounted for in the filter design as well as reconfiguration. Specifically, the filter can be redesigned to declare a fault only if the value of  $e_{i,k}(t)$  increases beyond some threshold,  $\delta(\theta_b)$ , where  $\delta(\theta_b)$  accounts for the deviation of the plant dynamics from the nominal dynamics in the absence of faults. Further robust controllers can be utilized and the robust stability regions can be used as criteria for deciding which backup configuration should be implemented in the closed-loop system.

**Remark 3.5:** In the event that the process state at the time of the failure of the primary control configuration lies in the stability region of more than one backup control configurations, additional performance considerations such as ease and/or cost of implementing one control configuration over another can be used in choosing the backup control configuration to be implemented [66]. Note that the set of initial conditions starting from where a given control configuration can stabilize a steady state – the so-called null-controllable region – is fundamentally limited by the constraints on the available control action, and that different control laws typically provide estimates of



the stability region which are subsets of the null-controllable region. If the state at the time of a failure lies outside the stability region of all the backup configurations, then this indicates that the backup configurations do not have enough control action available and calls for increasing the allowable control action.

### 3.4 Output-feedback fault-tolerant control

In the previous section, we assumed the availability of all the state measurements to illustrate the design of the fault detection and isolation filters and the controller reconfiguration strategy. In this section, we consider the case where only some of the process states are available for measurement. The unavailability of some states as measurements necessitates estimating the states from the measurements for the purposes of fault detection and isolation, feedback control and controller reconfiguration. To this end, we next review an output–feedback controller design [25] that provides estimates of the states (for other examples of nonlinear observer and output-feedback controller designs, see [47, 50]) along with an explicit characterization of the output feedback stability region.

#### 3.4.1 Output feedback controller

To design the output feedback controllers for the individual configurations, we will use the following assumption:

**Assumption 3.2 :** *Consider the system of Eq.3.1 in configuration  $k$  with  $\tilde{u}_k \equiv 0$ . There exists a set of integers  $r_{1,k}, r_{2,k}, \dots, r_{m,k}$  (with  $r_{1,k} + r_{2,k} + \dots + r_{m,k} = n$  for each  $k$ ) and a coordinate transformation  $\zeta_k = \chi_k(x)$  such that the representation of the system of Eq.3.1, in the  $\zeta_k$  coordinates, takes the form:*

$$\begin{aligned}
\dot{\zeta}_{1,k}^{(i)} &= \zeta_{2,k}^{(i)} \\
&\vdots \\
\dot{\zeta}_{r_{i,k}-1}^{(i)} &= \zeta_{r_{i,k}}^{(i)} \\
\dot{\zeta}_{r_{i,k}}^{(i)} &= L_f^{r_{i,k}} h_i(x) + L_{g_{i,k}} L_f^{r_{i,k}-1} h_i(x) u_{i,k}
\end{aligned} \tag{3.11}$$

where  $x = \chi_k^{-1}(\zeta_k)$  and  $\zeta_k = [\zeta_k^{(1)T} \cdots \zeta_k^{(m)T}]^T$ .

**Theorem 3.4 [25]:** Consider the constrained nonlinear process of Eq.3.1 with  $\tilde{u}_k(t) \equiv 0$  for which Assumption 3.2 holds, under the output feedback controller using the  $k$ th control configuration:

$$\begin{aligned}
\dot{\tilde{y}}_{i,k} &= \begin{bmatrix} -L_{i,k} a_{i,k}^{(1)} & 1 & 0 & \cdots & 0 \\ -L_{i,k}^2 a_{i,k}^{(2)} & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ -L_{i,k}^{r_i} a_{i,k}^{(r_i)} & 0 & 0 & \cdots & 0 \end{bmatrix} \tilde{y}_{i,k} \\
&+ \begin{bmatrix} L_{i,k} a_{i,k}^{(1)} \\ L_{i,k}^2 a_{i,k}^{(2)} \\ \vdots \\ L_{i,k}^{r_i} a_{i,k}^{(r_i)} \end{bmatrix} y_{i,k}
\end{aligned} \tag{3.12}$$

$$u_k = -w_k(\hat{x}, u_k^{max})(L_{G_k} V_k(\hat{x}))^T$$

where  $\hat{x} = \chi_k^{-1}(\text{sat}(\tilde{y}_k))$ ,  $\tilde{y}_k = [\tilde{y}_{(1,k)}^T \cdots \tilde{y}_{(m,k)}^T]^T$ ,  $i = 1, \dots, m$  where the parameters,  $a_{i,k}^{(1)}, \dots, a_{i,k}^{(r_i)}$  are chosen such that the polynomial  $s^{r_i} + a_{i,k}^{(1)} s^{r_i-1} + a_{i,k}^{(2)} s^{r_i-2} + \cdots + a_{i,k}^{(r_i)} = 0$  is Hurwitz,  $\hat{x} = \chi_k^{-1}(\text{sat}(\tilde{y}))$ ,  $\text{sat}(\cdot) = \min\{1, \zeta_{max,k} / \|\cdot\|\}(\cdot)$ , with  $\zeta_{max,k} = \beta_\zeta(\delta_{\zeta,k}, 0)$  where  $\beta_\zeta$  is a class  $\mathcal{KL}$  function and  $\delta_{\zeta,k}$  is the maximum value of the vector  $[l_1^T(x) \ l_2^T(x) \ \cdots \ l_m^T(x)]^T$  for  $V_k(x) \leq \delta_{b,k}$ , where  $l_i(x) = [h_i(x) \ L_f h_i(x) \ \cdots \ L_f^{r_i-1} h_i(x)]^T$ , and let  $\epsilon_k = \max_i 1/L_{i,k}$ . Then, given  $\Omega_{b,k} := \{x \in \mathbb{R}^n | V_k(x) \leq \delta_{b,k}\}$  and positive real numbers  $e_{m,k}$ ,  $\tilde{u}_k^*$  and  $d_k$  there exists  $\epsilon_k^* > 0$ ,  $T_k^b > 0$  such that if  $\epsilon_k \in (0, \epsilon_k^*]$ ,  $x(0) \in \Omega_{b,k}$ , and  $\|\tilde{y}(0)\| \leq \delta_{\zeta,k}$ , the origin of the closed-loop system is asymptotically (and locally exponentially) stable, and if  $\|\tilde{u}_k(t)\| \leq \tilde{u}_k^*$  then  $\|x(t) - \hat{x}(t)\| \leq e_{m,k}$  for

all  $t \geq T_k^b$  and  $\limsup_{t \rightarrow \infty} x(t) = d_k$ .

**Remark 3.6:** Theorem 3.4 above provides the estimation and controller design that guarantees asymptotic stability in the case of fault-free system as well as practical stability in the presence of ‘small’ faults (that preserve stability). The result relies on closeness of the state estimates to the true states over the infinite time interval. In fault detection and isolation, the closeness of solution would be required to hold even in the presence of large, possibly destabilizing faults, at least up-to some finite time to be able to detect and isolate the faults. This requirement is formalized in assumption 3.3 below.

**Assumption 3.3:** Consider the system of Eq.3.1 in configuration  $k$  under the output feedback controller of Theorem 3.4. There exist positive real numbers  $T_{close} > T_k^b$  and  $\delta_k$  such that if  $\|\tilde{u}_k(t)\| > \tilde{u}_k^*$  for some  $T_{fault} > T_k^b$  where  $\tilde{u}_k^*$  was defined in Theorem 3.4, then  $\|x(t) - \hat{x}(t)\| \leq e_{m,k}$  for all  $t \in [T_k^b, T_{fault} + T_k^{close}]$  and  $\|\int_{T_k^b}^t g_{j,k}[i](x(\tau))\tilde{u}_{j,k}(\tau)d\tau\| > \delta_k$  for some  $t \in [T_{fault}, T_{fault} + T_k^{close}]$ .

Due to the lack of full state measurements, the reconfiguration decision needs to be done based only on the available state estimates. It is therefore necessary to be able to make reliable inferences regarding the states using the state estimates. Proposition 3.1 below establishes the existence of a set,  $\Omega_{s,k} := \{x \in \mathbb{R}^n : V_k(x) \leq \delta_{s,k}\}$ , such that once the state estimation error has fallen below a certain value (note that the decay rate can be controlled by adjusting  $L_k$ ), the presence of the state within the output feedback stability region,  $\Omega_{b,k}$ , can be guaranteed by verifying the presence of the state estimates in the set  $\Omega_{s,k}$ . A similar approach was employed in the construction of the output feedback stability regions  $\Omega_{b,k}$  and the regions for the state estimates  $\Omega_{s,k}$  in the context of output feedback control of linear systems in [62], and for nonlinear systems in [29]. For a proof of the proposition, see [29].

**Proposition 3.1:** *Given any positive real number  $\delta_{b,k}$ , there exist positive real numbers  $e_{m,k}^*$  and  $\delta_{s,k}$  such that if  $\|x - \hat{x}\| \leq e_{m,k}$ , where  $e_{m,k} \in (0, e_{m,k}^*]$ , and  $V_k(\hat{x}) \leq \delta_{s,k}$ , then  $V_k(x) \leq \delta_{b,k}$ .*

### 3.4.2 Output-feedback fault detection and isolation filter

The output feedback fault detection and isolation filter uses the same principle as the state feedback fault detection and isolation filter while using the state estimates to implement the filter. For the system of Eq.3.1, the fault detection and isolation filter for the  $j$ th manipulated input in the  $k$ th configuration is designed as:

$$\begin{aligned}\dot{\tilde{x}}_{i,k} &= f_i(\hat{x}_{1,k}, \dots, \tilde{x}_{i,k}, \dots, \hat{x}_{n,k}) \\ &\quad + g_{j,k}[i](\hat{x}_1^k, \dots, \tilde{x}_{i,k}, \dots, \hat{x}_{n,k})u_{j,k}(\hat{x}_{1,k}, \dots, \tilde{x}_{i,k}, \dots, \hat{x}_{n,k}) \\ e_{i,k} &= \hat{x}_{i,k} - \tilde{x}_{i,k}\end{aligned}\quad (3.13)$$

where  $g_{j,k}[i]$  denotes the  $i$ th element of the vector  $g_{j,k}$ , and  $\tilde{x}_{i,k}(T_k^b) = \hat{x}_{i,k}(T_k^b)$ , where  $T_k^b$  was defined in Theorem 3.4.

**Proposition 3.2:** *Consider the nonlinear system of Eq.3.1, for a fixed mode under the output feedback controller of Eq.3.12 and the filter of Eq.3.13. Given  $\tilde{u}_{j,k}^*$ ,  $\delta_k$  and  $T_k^{close}$  there exist positive real numbers  $\delta_{j,k}$  and  $\epsilon_k^{**}$  such that if  $|\tilde{u}_{j,k}(t)| \geq \tilde{u}_{j,k}^*$  for some  $T_k^{fault} \geq T_{b,k}$  and  $\epsilon_k \leq \min\{\epsilon_k^*, \epsilon_k^{**}\}$  then  $e_{i,k}(t) > \delta_{j,k}$  for some  $t \in [T_k^{fault}, T_k^{fault} + T_k^{close}]$ .*

**Proof of Proposition 3.2:** Consider, the filter of Eq.3.13 and the evolution of  $x_i$  for  $t \in [T_k^b, T_k^{fault} + T_k^{close}]$ , i.e., consider the systems

$$\begin{aligned}\dot{\tilde{x}}_{i,k} &= f_i(x) + g_{j,k}[i](x)(u_{j,k}(x)) \\ &\quad + (f_i(\hat{x}_{1,k}, \dots, \tilde{x}_{i,k}, \dots, \hat{x}_{n,k}) - f_i(x)) \\ &\quad + (g_{j,k}[i](\hat{x}_{1,k}, \dots, \tilde{x}_{i,k}, \dots, \hat{x}_{n,k})u_{j,k}(\hat{x}_{1,k}, \dots, \tilde{x}_{i,k}, \dots, \hat{x}_{n,k}) \\ &\quad - g_{j,k}[i](x)u_{j,k}(x))\end{aligned}\quad (3.14)$$

and

$$\dot{x}_{i,k} = f_i(x) + g_{j,k}[i](x)(u_{j,k}(x) + \tilde{u}_{j,k}(t))\quad (3.15)$$

Therefore,

$$\begin{aligned} \dot{x}_{i,k} - \dot{\tilde{x}}_{i,k} &= g_{j,k}[i](x)\tilde{u}_{j,k}(t) + (f_i(x) - f_i(\hat{x}_{1,k}, \dots, \tilde{x}_{i,k}, \dots, \hat{x}_{n,k})) \\ &\quad + (g_{j,k}[i](x)u_{j,k}(x) \\ &\quad - g_{j,k}[i](\hat{x}_{1,k}, \dots, \tilde{x}_{i,k}, \dots, \hat{x}_{n,k})u_{j,k}(\hat{x}_{1,k}, \dots, \tilde{x}_{i,k}, \dots, x_{n,k} + \hat{x}_{n,k})) \end{aligned} \quad (3.16)$$

Note that  $\hat{x}(T_b) - x(T_b)$  can be made as small as desired by choosing a sufficiently small  $\epsilon$ . From the continuity of  $f_i(\cdot)$  and  $g_{j,k}[i](\cdot)$ , this implies that the last two terms in Eq.3.16 can be made as small as desired. The difference between  $\dot{x}_{i,k}$  and  $\dot{\tilde{x}}_{i,k}$  can therefore be made as close as desired to  $g_{j,k}[i](x)(\tilde{u}_{j,k}(t))$ . Using assumption 3.3, therefore, given a time  $T^{close} > T_k^b$ , there exists a positive real number  $\delta_{j,k}^* = \delta_k^*$  such that if  $|\tilde{u}_{j,k}(t)| > \tilde{u}_{j,k}^*$  for some  $T_k^{fault} \geq T_k^b$  then  $\|x_{i,k}(t) - \tilde{x}_{i,k}(t)\| \geq \delta_{j,k}^*$  for some  $t \in [T_k^{fault}, T_k^{fault} + T_k^{close}]$ . Finally, once again since  $\hat{x}(t) - x(t)$  can be made as close as desired (up until  $T_k^{close}$ ), then given that  $\|x_{i,k}(t) - \tilde{x}_{i,k}(t)\| \geq \delta_{j,k}^*$ , there exists a positive real number  $\delta_{j,k}$  such that  $e_{i,k} = \|\hat{x}_{i,k}(t) - \tilde{x}_{i,k}(t)\| \geq \delta_{j,k}$  for some  $t \in [T_k^{fault}, T_k^{fault} + T_k^{close}]$ . In summary, there exists a positive real number  $\epsilon_k^{**}$  such that if  $\epsilon_k \leq \min\{\epsilon_k^*, \epsilon_k^{**}\}$  and  $|\tilde{u}_{j,k}(t)| \geq \tilde{u}_{j,k}^*$  for some  $T_k^{fault} \geq T_{b,k}$  then  $e_{i,k}(t) > \delta_{j,k}$  for some  $t \in [T_k^{fault}, T_k^{fault} + T_k^{close}]$ .

**Remark 3.7:** Note that unlike the case of full state-feedback, the fault detection filter is initialized only after the passage of some short period of time,  $T_k^b$  (which can be chosen arbitrarily small by increasing the observer gain), to ensure that the closed-loop state estimates have converged sufficiently close to the true closed-loop states and thus – by setting the filter state  $\tilde{x}_{i,k}$  at this time equal to the value of the state estimate – ensure that the filter state is initialized sufficiently close to the true values of the state. Note also, that unlike the case of full state availability, where the filter is able to immediately detect and isolate the occurrence of fault, the lack of measurements which induces the error in the initialization of the filter states allows detection of only such faults that impact the states of the closed-loop system above a

certain threshold. The key is to ensure that only such faults go undetected which do not impact undesirably on the stability of the closed-loop system. In the subsequent section, we design an output-feedback fault detection and isolation and fault-tolerant control structure that ensures detection and isolation of destabilizing faults.

### 3.4.3 Output-feedback fault detection and isolation and fault tolerant control

Having designed the state estimators and controllers and output feedback fault detection and isolation filters, in this section we present an integrated output-feedback fault detection and isolation and fault-tolerant controller structure. To this end, consider the nonlinear system of Eq.3.1, for which the output feedback controller of Eq.3.12 and the filters of Eq.3.13 have been designed for each manipulated input under the primary configuration,  $k(0) = k_0$  under possible faults in only one control actuator. The theorem below formalizes the integrated output-feedback fault detection and isolation and fault-tolerant control structure.

**Theorem 3.5:** *Let  $k(0) = k_0$  for some  $k_0 \in \mathcal{K}$ ,  $x(0) \in \Omega_{b,k_0}$ ,  $\tilde{x}_{i,k}(T_{i,k}^b) = \hat{x}(T_{i,k}^b)$ . Given a positive real number  $d_{k_0}$  there exist positive real numbers  $\delta_{i,k}$  and  $\epsilon_k^{***}$  such that if  $\epsilon_k \in (0, \epsilon_k^{***}]$  then under the switching rule*

$$k(t) = \left\{ \begin{array}{ll} k_0, & 0 \leq t < T_{detect} \\ q \neq k_0, & t \geq T_{detect}, \hat{x}(T_{detect}) \in \Omega_{s,q}, \\ & u_{j,k_0} \notin u_q \end{array} \right\} \quad (3.17)$$

where  $T_{detect}$  is the earliest time for which  $e_{i,k} > \delta_{i,k}$  for some  $i \in [0, \dots, n]$ , we have that  $\limsup_{t \rightarrow \infty} x(t) \leq d_{k_0}$ .

**Proof of Theorem 3.5:** We consider the two cases, 1)  $e_{i,k}(t) \leq \delta_{i,k} \forall t$  and 2)  $e_{i,k}(t) > \delta_{i,k}$  for some  $t = T_{detect}$ .

*Case 1:* From theorem 3.4, we have that given a positive real number  $d_k$ , there exist positive real numbers  $\epsilon_k^{**}$  and  $\tilde{u}_k^*$  such that if  $\|\tilde{u}_{j,k}(t)\| \leq \tilde{u}_k^*$ , then  $\limsup_{t \rightarrow \infty} x(t) = d_{k_0}$ . For such choices of  $\epsilon_k^{**}$  and  $\tilde{u}_k^*$ , we have from Proposition 3.2 that there exists a positive real number  $\delta_{i,k}$  such that if  $\epsilon_k \in (0, \min\{\epsilon_k^*, \epsilon_k^{**}\} = \epsilon_k^{***}]$  then  $e_{i,k} \leq \delta_{i,k} \Rightarrow \|\tilde{u}_{j,k}(t)\| \leq \tilde{u}_k^*$ . Therefore, for the above choices of  $\tilde{u}_k^*$ ,  $\epsilon_k^{***}$  and  $\delta_{j,k}$ , we have that  $e_{i,k}(t) \leq \delta_{i,k}$  implies  $\|\tilde{u}_{i,k_0}(t)\| \leq \tilde{u}_{i,k_0}^*$  yielding  $\limsup_{t \rightarrow \infty} x(t) = d_{k_0}$ .

*Case 2:* The switching rule of Eq.3.17 ensures that at  $t = T_{detect}$ ,  $\hat{x}(t) \in \Omega_{s,q}$ , which in turn implies that  $x(t) \in \Omega_{b,q}$  (Proposition 3.1). This, together with the switching to the  $q$ th control configuration ensures asymptotic stability of the origin of the closed-loop system (Theorem 3.4). In either cases we get that  $\limsup_{t \rightarrow \infty} x(t) \leq d_{k_0}$ . This completes the proof of the theorem.

The design of the output feedback fault detection and isolation filter and controller reconfiguration is best understood through the following algorithm

1. Given the system of the form of Eq.3.1, design the output feedback controller of Eq.3.12, that also yields estimates of the states, and estimate the output feedback stability regions of the control configurations,  $\Omega_{b,k}$  and the sets  $\Omega_{s,k}$ , defined in Proposition 3.1, and compute the values of  $T_k^b$ . For an initial condition in the stability region of the  $k_0$ th control configuration, initialize the state estimator and the output feedback controller as described in Theorem 3.4.
2. After a time  $T_{k_0}^b$ , initialize the fault detection and isolation filters of the form of Eq.3.13 using the values of the state estimates at time  $T_{k_0}^b$ .
3. Monitor the evolution of the residuals ( $e_{i,k_0}$ ). If any of the residuals go above the threshold, it implies that a possibly destabilizing fault has occurred.
4. Switch to a configuration  $q$  for which the closed-loop state estimates at the

time of fault detection lie in  $\Omega_{s,q}$ , where  $\Omega_{s,q}$  was defined in Proposition 3.1 (this ensures that the states are in the output feedback stability region of the  $q$ th configuration) and one which does not involve the failed control actuator.

5. Implement this control configuration to achieve closed-loop stability and fault-tolerant control.

**Remark 3.8:** Note that while the above switching rule provides a sufficient condition for practical stability, it is not a necessary condition. In other words, the value of the residual going above the threshold does not imply that a destabilizing fault has occurred. However, the value of the residual being less than the threshold does ensure that no destabilizing fault has occurred. So while the above switching logic may trigger a switching where simply continuing with the primary control configuration could have preserved stability (i.e., it allows for false alarms), it is designed to preclude the possibility that a destabilizing fault takes place and reconfiguration is not executed. This, however, is not a limitation of the proposed filter, but stems simply from the fundamental problem of differentiating between the error introduced in the filtering system due to the presence of estimation errors and those due to the faults.

**Remark 3.9:** Note that while the algorithm above is written for the case of a single fault, generalization to multiple faults, whether simultaneous or otherwise, is straightforward: the current fault detection filter design can detect and isolate multiple faults, while the reconfiguration rule can be ‘re-initialized’ after the first backup control configuration is activated to handle subsequent faults (see the simulation section for a demonstration).



### 3.5 Simulation examples

We demonstrate the application of the proposed fault detection and isolation and reconfiguration strategy to two chemical reactors configured to operate in series. To this end, consider two well mixed, non-isothermal continuous stirred tank reactors (see Fig.3.1), 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.  $A$  is the reactant species,  $B$  is the desired product and  $U$  and  $R$  are undesired byproducts. The feed to the first reactor consists of pure A at a flow rate  $F_0$ , molar concentration  $C_{A0}$  and temperature  $T_0$ . The output from the first reactor is fed to the second reactor along with a fresh feed that consists of pure A at a flow rate  $F_3$ , molar concentration  $C_{A03}$ , and temperature  $T_{03}$ . Due to the non-isothermal nature of the reactions, jackets are used to remove or provide heat to the reactors. Under standard modeling assumptions, a mathematical model of the process can be derived from material and energy balances and takes the following form:

$$\begin{aligned}
 \frac{dT_1}{dt} &= \frac{F_0}{V_1}(T_0 - T_1) + \sum_{i=1}^3 \frac{(-\Delta H_i)}{\rho c_p} R_i(C_{A1}, T_1) + \frac{Q_1}{\rho c_p V_1} \\
 \frac{dC_{A1}}{dt} &= \frac{F_0}{V_1}(C_{A0} - C_{A1}) - \sum_{i=1}^3 R_i(C_{A1}, T_1) \\
 \frac{dT_2}{dt} &= \frac{F_0}{V_2}(T_1 - T_2) + \frac{F_3}{V_2}(T_{03} - T_2) + \sum_{i=1}^3 \frac{(-\Delta H_i)}{\rho c_p} R_i(C_{A2}, T_2) + \frac{Q_2}{\rho c_p V_2} \\
 \frac{dC_{A2}}{dt} &= \frac{F_0}{V_2}(C_{A1} - C_{A2}) + \frac{F_3}{V_2}(C_{A03} - C_{A2}) - \sum_{i=1}^3 R_i(C_{A2}, T_2)
 \end{aligned} \tag{3.18}$$

where  $R_i(C_{Aj}, T_j) = k_{i0} \exp\left(\frac{-E_i}{RT_j}\right) C_{Aj}$ , 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.  $\Delta H_i$ ,  $k_i$ ,  $E_i$ ,  $i = 1, 2, 3$ , denote the enthalpies, pre-exponential constants and activation energies of the three re-

actions, respectively,  $c_p$  and  $\rho$  denote the heat capacity and density of the fluid. The values of the process parameters can be found in Table 3.1. CSTR 1, with  $Q_1 = 0$ , has three steady-states: two locally asymptotically stable and one unstable at  $(T_1^s, C_{A1}^s) = (388.57 \text{ K}, 3.59 \text{ kmol/m}^3)$ . 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  $(T_2^s, C_{A2}^s) = (429.24 \text{ K}, 2.55 \text{ kmol/m}^3)$ .

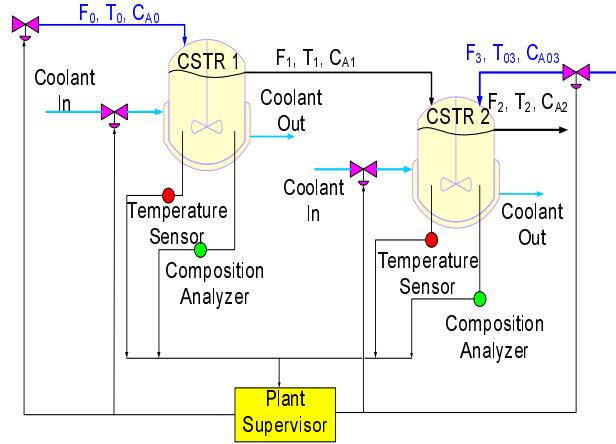


Figure 3.1: A schematic of two CSTRs operating in series.

The control objective is to stabilize the reactors 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 actuator failures, we consider the following manipulated input candidates:

1. Rate of heat input into reactor one,  $Q_1$ , subject to the constraint  $|Q_1| \leq 1.4 (10^7) \text{ kJ/hr}$ .
2. Reactor one inlet stream temperature,  $T_0 - T_0^s$ , subject to the constraint  $|T_0 - T_0^s| \leq 60 \text{ K}$ .
3. Reactor one inlet reactant concentration,  $C_{A0} - C_{A0}^s$ , subject to the constraint

Table 3.1:

$F_0$	=	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	$kmol/m^3$
$C_{A03}^s$	=	3.0	$kmol/m^3$
$\Delta H_1$	=	$-5.0 \times 10^4$	$KJ/kmol$
$\Delta H_2$	=	$-5.2 \times 10^4$	$KJ/kmol$
$\Delta H_3$	=	$-5.4 \times 10^4$	$KJ/kmol$
$k_{10}$	=	$3.0 \times 10^6$	$hr^{-1}$
$k_{20}$	=	$3.0 \times 10^5$	$hr^{-1}$
$k_{30}$	=	$3.0 \times 10^5$	$hr^{-1}$
$E_1$	=	$5.0 \times 10^4$	$KJ/kmol$
$E_2$	=	$7.53 \times 10^4$	$KJ/kmol$
$E_3$	=	$7.53 \times 10^4$	$KJ/kmol$
$\rho$	=	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	$kmol/m^3$

$$|C_{A0} - C_{A0}^s| \leq 4.0 \text{ kmol}/m^3.$$

4. Rate of heat input into reactor two,  $Q_2$ , subject to the constraint  $|Q_2| \leq 4.2 (10^7) \text{ kJ}/hr$ .
5. Reactor two inlet stream temperature,  $T_{03} - T_{03}^s$ , subject to the constraint  $|T_{03} - T_{03}^s| \leq 60 \text{ K}$ .
6. Reactor two inlet reactant concentration,  $C_{A03} - C_{A03}^s$ , subject to the constraint  $|C_{A03} - C_{A03}^s| \leq 3.0 \text{ kmol}/m^3$ .

The above manipulated inputs can be used in various combinations to stabilize the reactors using measurements of the reactor temperatures and reactant concentrations provided by the sensors (full state-feedback) and to employ reconfiguration. The primary control configuration ( $k = 1$ ) involves four inputs consisting of the two heating jackets and the two inlet stream concentrations ( $Q_1$ ,  $Q_2$ ,  $C_{A0}$ , and  $C_{A03}$ ). In the event of a partial failure in this configuration the supervisor needs to detect and isolate the fault and activate a fall-back configuration in order to maintain closed-loop stability.

We first illustrate the application of the fault detection and isolation and fault-tolerant control under state-feedback control. A quadratic Lyapunov function of the form  $V_k = x^T P_k x$ , where  $P_k$  is a positive-definite symmetric matrix that satisfies the Riccati inequality  $A^T P_k + P_k A - P_k b_k b_k^T P_k < 0$ , is used in controller design with  $A$  and  $b$  based on the linearized system around the desired steady-state.

1. For the primary control configuration, the manipulated inputs are scaled to give

$$b_1^* = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0.0198 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0.0297 \end{bmatrix} \text{ and}$$

$$P_1 = \begin{bmatrix} 1.2290 & 2.2195 & 0.0203 & 0.1733 \\ 2.2195 & 28.4462 & 0.1396 & 8.8183 \\ 0.0203 & 0.1396 & 1.6150 & 9.8728 \\ 0.1733 & 8.8183 & 9.8728 & 145.7245 \end{bmatrix}.$$

2. The fall-back control configuration involves four manipulated inputs given by

$$u_2 = [T_0 - T_0^s \quad C_{A0} - C_{A0}^s \quad T_{03} - T_{03}^s \quad C_{A03} - C_{A03}^s]'. \text{ Scaling the manipulated}$$

$$\text{input yields } b_2^* = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0.1333 & 0 & 0 \\ 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 0.2 \end{bmatrix} \text{ and}$$

$$P_2 = \begin{bmatrix} 1.1991 & 1.8730 & .00051 & 0.0236 \\ 1.8730 & 12.6725 & 0.0093 & 0.4141 \\ 0.0051 & 0.0093 & 0.6150 & 1.9055 \\ 0.0236 & 0.4141 & 1.9055 & 17.9826 \end{bmatrix}.$$

The state-feedback controller of Eq.3.2 is subsequently designed for both the control configurations, and their stability region characterized, yielding  $c_1^{max}$  and  $c_2^{max}$  equal to 7.2 and 1.9 respectively. The fault detection filters are designed using Eq.3.5 and the reactors as well as the filter states for the first control configuration are initialized at  $T_1(0) = 386.8 \text{ K}$ ,  $C_{A1}(0) = 3.6 \text{ kmol/m}^3$ ,  $T_2(0) = 430.5 \text{ K}$ ,  $C_{A2}(0) = 2.56 \text{ kmol/m}^3$ . This initial condition is within the stability region of the primary control configuration ( $V_1(x) = 6.64 \leq c_1^{max} = 7.2$ ). As shown by the solid lines in Figs.3.2-3.5 the controller proceeds to drive the closed-loop trajectory toward the desired steady-state until the heating jackets fail simultaneously 0.1 minutes after reactor startup. As can be seen in Fig.3.6 and Fig.3.7 the values of only the residuals  $e_{1,1}(t)$  and  $e_{3,1}(t)$  become non-zero, thereby detecting as well as isolating the faults in the control actuators. If the supervisor does not perform any switching at this point closed-loop stability is not achieved (dashed lines in Figs.3.2-3.5). Note that this occurs because the actuators heating/cooling the jackets have failed, but the controller still tries to use the heat supplied to/removed from the reactors as manipulated inputs. Having identified that the faults occurred in the actuators changing  $Q_1$  and

$Q_2$ , the supervisor can implement the fall-back configuration (using  $T_0$ ,  $C_{A0}$ ,  $T_{03}$ , and  $C_{A03}$  as the manipulated inputs,  $k = 2$ ) since the fall-back configuration does not use the failed actuators. Furthermore, at the time when the fault is detected, the state of the closed loop system is within the stability region of the backup control configuration ( $V_2(x(t = 0.162)) = 0.221 < c_2^{max} = 1.9$ ). The supervisor therefore activates the fall-back configuration (solid lines in Figs.3.2-3.5) which stabilizes the closed-loop system and achieves fault-tolerant control.

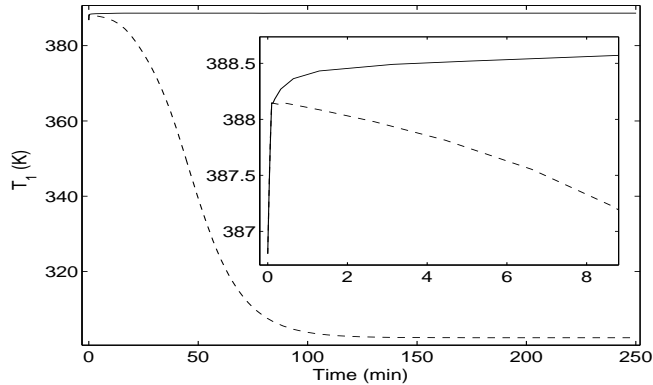


Figure 3.2: Evolution of reactor one closed-loop temperature profile under the switching rule of Theorem 3.3 (solid line) and in the absence of fault-tolerant control (dashed line) subject to simultaneous failures in both the heating jackets.

The next simulation illustrates the application of fault detection and isolation and fault-tolerant control when not all of the process states are available for measurement. In this case the output-feedback methodology is implemented on the same two-reactor system used for the previous simulation study with changes to the parameters  $F_3 = 4.998 \text{ m}^3/\text{hr}$  and  $V_2 = 0.5 \text{ m}^3$ . This changes the unstable steady state of the second reactor to  $T_2^s = 433.96 \text{ K}$  and  $C_{A2}^s = 2.88 \text{ kmol/m}^3$ . The dynamics for the controller are designed using the same state-feedback methodologies as in the previous simulation study. However, the controller utilizes the state estimates to compute a control action. The fault detection and isolation filter is designed based

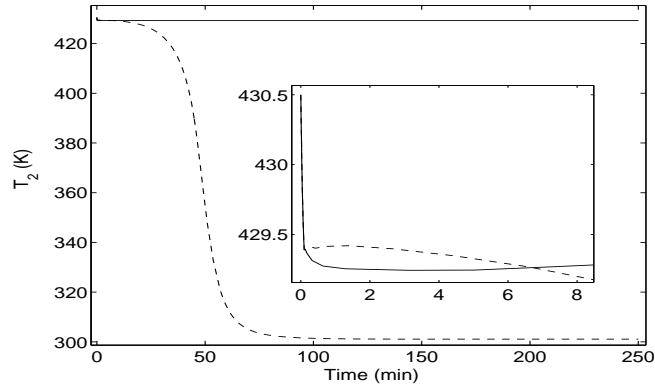


Figure 3.3: Evolution of reactor two closed-loop temperature profile under the switching rule of Theorem 3.3 (solid line) and in the absence of fault-tolerant control (dashed line) subject to simultaneous failures in both the heating jackets.

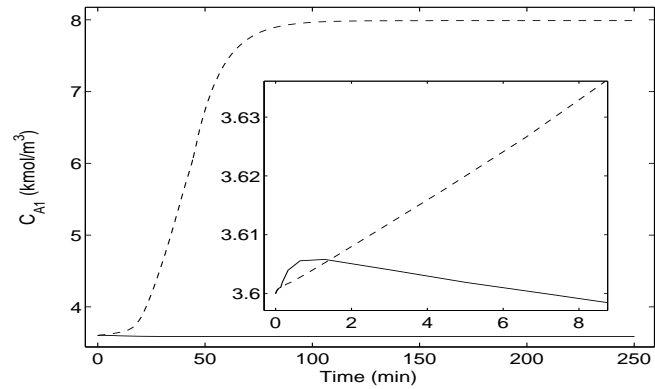


Figure 3.4: Evolution of reactor one closed-loop reactant concentration profile under the switching rule of Theorem 3.3 (solid line) and in the absence of fault-tolerant control (dashed line) subject to simultaneous failures in both the heating jackets.

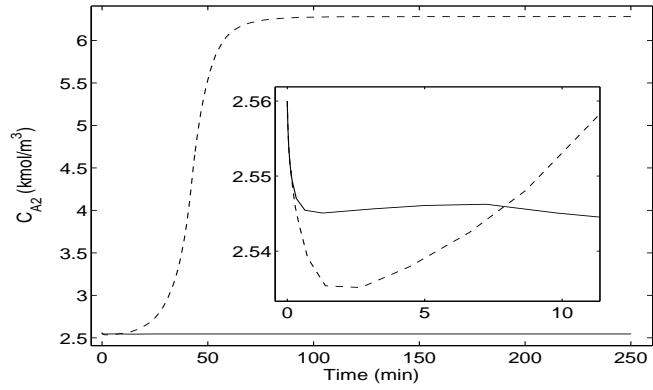


Figure 3.5: Evolution of reactor two closed-loop reactant concentration profile under the switching rule of Theorem 3.3 (solid line) and in the absence of fault-tolerant control (dashed line) subject to simultaneous failures in both the heating jackets.

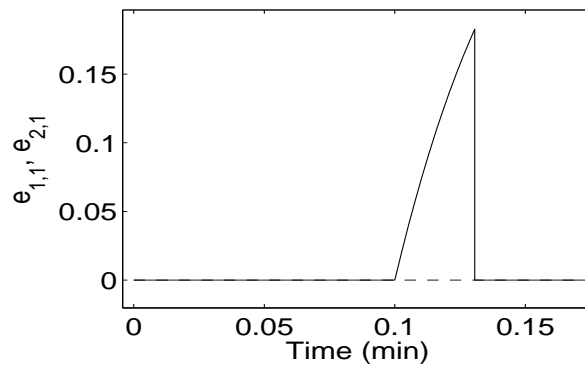


Figure 3.6: Evolution of residuals  $e_{1,1}$  (solid line) and  $e_{2,1}$  (dashed line) corresponding to the manipulated inputs in the first reactor.



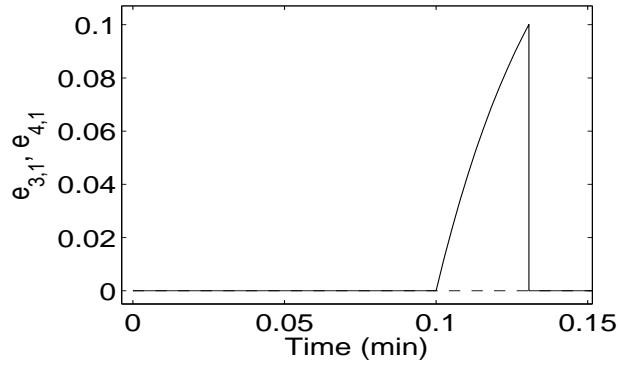


Figure 3.7: Evolution of residuals  $e_{3,1}$  (solid line) and  $e_{4,1}$  (dashed line) corresponding to the manipulated inputs in the second reactor.

on Eq. 3.13.

The control objective is to stabilize the reactor at the open-loop unstable steady-state using measurements of  $C_{A1}$  and  $C_{A2}$ . The available manipulated inputs include the rate of heat input into reactor one,  $Q_1$ , subject to the constraint  $|Q_1| \leq 2.333 (10^6) \text{ kJ/hr}$ , the rate of heat input into reactor two,  $Q_2$ , subject to the constraint  $|Q_2| \leq 1.167 (10^6) \text{ kJ/hr}$ , and a duplicate backup heating configuration for reactor one,  $Q_3$ , subject to the constraint  $|Q_3| \leq 2.333 (10^6) \text{ kJ/hr}$ .

The primary control configuration ( $k = 1$ ) consists of the manipulated inputs  $Q_1$  and  $Q_2$ , while the backup configuration ( $k = 2$ ) consists of manipulated inputs  $Q_2$  and  $Q_3$ . In order to implement the state-feedback Lyapunov-based controllers, estimates of  $T_1$  and  $T_2$  are generated using a state estimator of the form of Eq. 3.12 with  $L_{i,k} = 10000$ ,  $a_{i,k}^{(1)} = 5$ , and  $a_{i,k}^{(2)} = 1$  for  $i = 1, 2$  and  $k = 1, 2$ . The reactors are initialized at  $T_1(0) = 386.97 \text{ K}$ ,  $C_{A1}(0) = 3.59 \text{ kmol/m}^3$ ,  $T_2(0) = 432.36 \text{ K}$ , and  $C_{A2}(0) = 2.88 \text{ kmol/m}^3$ . The state estimator is initialized at the steady-state values for this system ( $\tilde{T}_1(0) = 388.57 \text{ K}$ ,  $\tilde{C}_{A1}(0) = 3.59 \text{ kmol/m}^3$ ,  $\tilde{T}_2(0) = 433.96 \text{ K}$ , and  $\tilde{C}_{A2}(0) = 2.88 \text{ kmol/m}^3$ ). The fault detection filter states are initialized with the

value of the state estimates at  $t = 0.0022 \text{ min} \equiv T_1^b$ . Note that by this time the estimates have converged sufficiently close to the true values as can be seen as the dash-dotted lines in Fig.3.8.

As shown by the solid line in Fig.3.8, the controller drives the closed-loop system to the desired steady-state (for the sake of brevity, only  $T_1$  is shown). A complete failure occurs in  $Q_1$  early on at  $T_f = 0.01 \text{ min}$  while the system is still moving toward the desired steady-state. If the fault is not detected and no switching takes place the value of  $T_1$  moves away from the desired operating temperature shown as the dotted line in Fig.3.8. However, when the fault detection and isolation filter is utilized we can see the filter value  $\hat{T}_1$ , dashed line in Fig.3.8, diverges from the estimated value  $\tilde{T}_1$ . This discrepancy causes the residual  $e_{1,1}(t)$  corresponding to  $Q_1$  to rise to the threshold value of  $0.01 \text{ K}$  (chosen to ensure that all destabilizing faults are detected) at time  $t = 0.0116 \text{ min}$ , as shown in Fig.3.9. A fault in  $Q_1$  is declared at this time, and the supervisor checks the value of the Lyapunov function for  $k = 2$ . Since  $V_2(0.0116) = 0.38 < c_2^{max} = 9.4$  the supervisor activates the fall-back configuration to achieve closed-loop stability despite actuator failure in  $Q_1$ . The fault detection and isolation filter is restarted 0.0022 minutes later at  $T_2^b = 0.0138 \text{ min}$ . As expected, no fault is declared at any time in  $Q_2$  as can be seen in Fig.3.10. In summary, the output-feedback fault detection and isolation and fault-tolerant control system is able to detect and isolate the fault to allow reconfiguration and drive the system to the desired steady state (solid line in Fig.3.8).

The application and effectiveness of the proposed fault-detection and isolation and fault-tolerant control method has been illustrated in the case of both state and output feedback. Next, this method is applied in the presence of uncertainty and measurement noise. To this end consider the two reactor system used in the previous

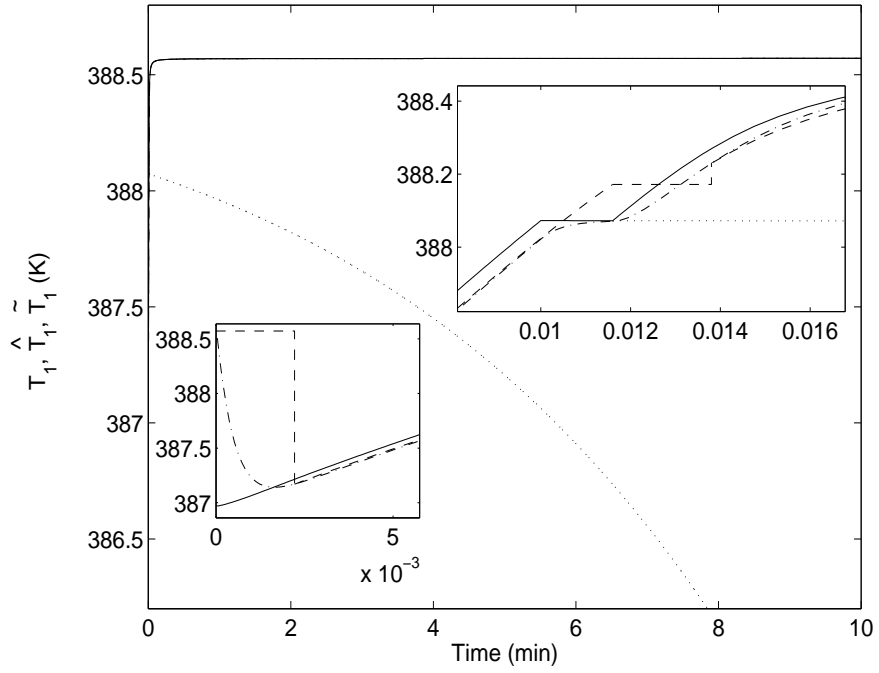


Figure 3.8: Evolution of the closed-loop temperature (solid line), estimate of temperature (dash-dotted line), and the temperature profile generated by the FDI filter (dashed line) with fault-tolerant control in place. Evolution of the temperature (dotted line) without fault-tolerant control in place.

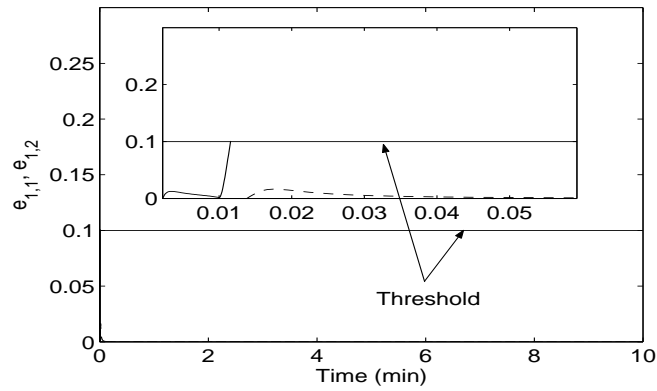


Figure 3.9: Evolution of the residual corresponding to  $Q_1$  for before switching ( $k = 1$ , solid line), and  $Q_3$  after switching ( $k = 2$ , dashed line). A fault is declared when  $e_{1,1}$  reaches the threshold at 0.1.

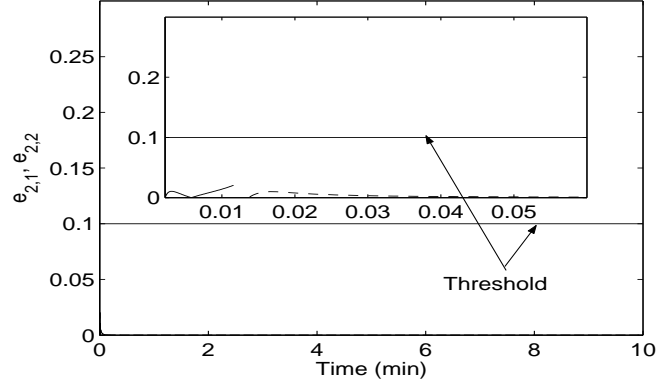


Figure 3.10: Evolution of the residual corresponding to  $Q_2$  for before switching ( $k = 1$ , solid line), and after switching ( $k = 2$ , dashed line). No fault is declared.

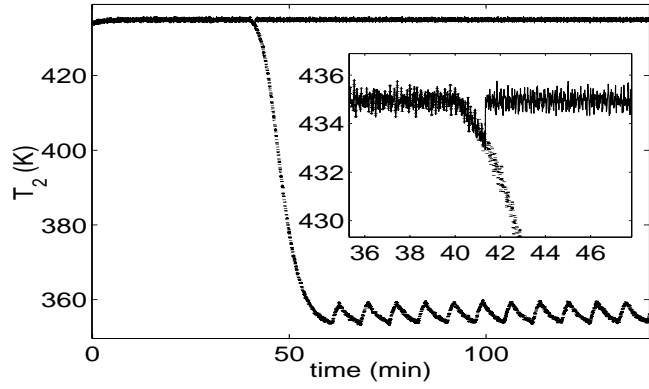
example with full-state feedback.

The control objective is to stabilize the reactor at the open-loop unstable steady-state where  $(T_1^s, C_{A1}^s) = (388.57 \text{ K}, 3.59 \text{ kmol}/\text{m}^3)$  and  $(T_2^s, C_{A2}^s) = (433.96 \text{ K}, 2.88 \text{ kmol}/\text{m}^3)$ . The measurements of temperature and concentration are assumed to contain a noise of magnitude  $1 \text{ K}$  and  $0.1 \text{ kmol}/\text{m}^3$ , respectively. Also, the concentration of A in the inlet streams  $C_{A0}$  and  $C_{A03}$  used in the process model are 10% smaller than the values used in the filter equations and the controller. The available manipulated inputs include the rate of heat input into reactor one,  $Q_1$ , subject to the constraint  $|Q_1| \leq 2.333 (10^6) \text{ kJ}/\text{hr}$ , the rate of heat input into reactor two,  $Q_2$ , subject to the constraint  $|Q_2| \leq 1.167 (10^6) \text{ kJ}/\text{hr}$ , and a duplicate backup heating configuration for reactor two,  $Q_3$ , subject to the constraint  $|Q_3| \leq 1.167 (10^6) \text{ kJ}/\text{hr}$ .

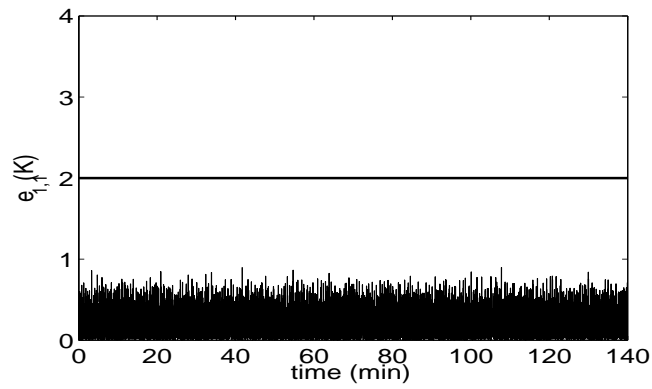
The primary control configuration consists of the manipulated inputs  $Q_1$  and  $Q_2$ , while the backup configuration is comprised of manipulated inputs  $Q_1$  and  $Q_3$ . As before, quadratic Lyapunov functions of the form  $V_k = x^T P_k x$  are used for controller design. the controller design yields a stability region estimate with  $c_1^{max}$  and  $c_2^{max}$  both approximately equal to 9.4. Note that all the information about the stability region

is completely contained in the values of  $c_1^{max}$  and  $c_2^{max}$ . Specifically, the presence of the closed-loop state in the stability region can be ascertained by simply evaluating the value of the Lyapunov-function and checking against the value of  $c^{max}$ .

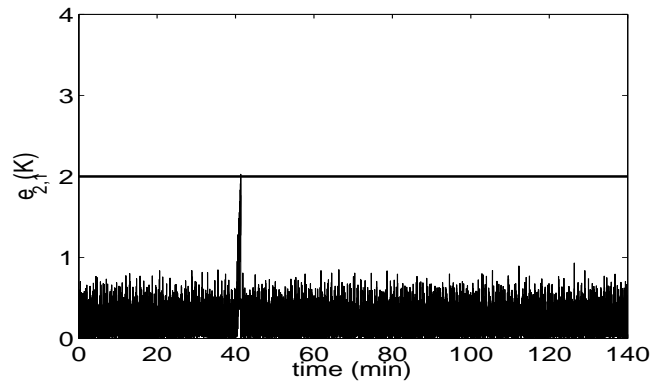
In the first scenario the ability to detect a fault in the presence of multiple disturbances and noise is demonstrated. The reactors, as well as the fault detection filter for the first control configuration are initialized at the desired unstable steady-state. For the sake of brevity, only the evolution of  $T_2$  and the residuals are shown. As can be seen in Fig.3.11a, the controller maintains the closed-loop trajectory near the desired steady-state until heating jacket two ( $Q_2$ ) fails 40 min after reactor startup. If a fault-detection and isolation filter is not in place, and the fault is not detected, close-loop stability is not achieved (dotted lines in Fig.3.11a). The fault-detection and isolation filter designed using the proposed methodology, however, detects this fault when the value of residual  $e_{2,1}(t)$  becomes greater than the threshold value of 2.0 K at  $t = 40.79 \text{ min}$  (see Fig. Fig.3.11c) while  $e_{1,1}(t)$  (Fig.3.11b) remains below the threshold of 2.0, allowing the detection and isolation of the fault. While at the time of the failure ( $t = 40 \text{ min}$ ), the state of the closed-loop system is within the stability region of the backup-configuration, but the time that the failure is detected at  $t = 40.79 \text{ min}$ , operation of reactor two in an open-loop fashion for 0.79 min results in the state moving out of the stability region of the backup configuration ( $V_2(40.79) = 73.17 > c_2^{max} = 9.4$ ) and stability is not guaranteed after switching. However, it is possible that stability may still be achieved by using the fallback configuration. In particular, having been alerted by the fault-detection and isolation filter of the occurrence of the fault, the supervisor activates the fallback configuration (with  $Q_1$  and  $Q_3$  as the manipulated inputs, solid lines in Fig.3.11a) and is able to drive the system to the desired steady-state and enforce closed-loop stability.



(a)



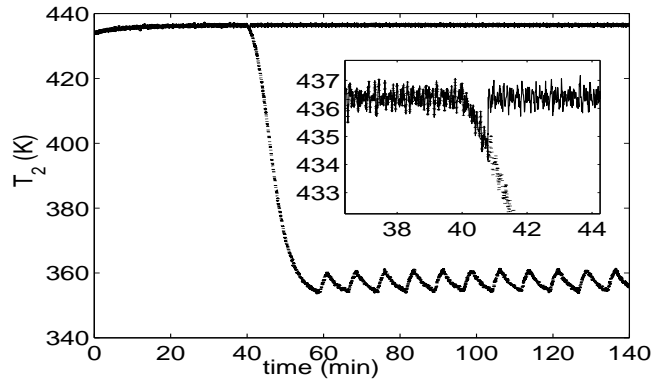
(b)



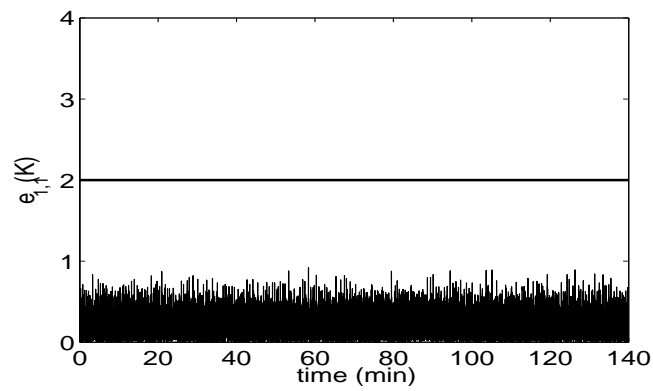
(c)

Figure 3.11: (a) Temperature profile of reactor two with reconfiguration (solid line) and without reconfiguration (dotted line), (b)  $Q_1$  residual profile, and (c)  $Q_2$  residual profile (note fault detection at time  $t = 40.79 \text{ min}$ ).

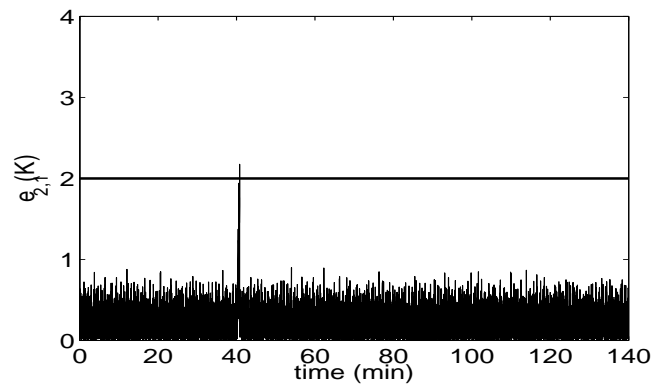
Detection of faults in the presence of process disturbances and noise is clearly possible using the methodology above. In order to guarantee stability after switching, however, the disturbances acting on the system should be reduced or the constraints on the control action should be relaxed to enlarge the estimate of the closed-loop stability region. In the second scenario, the ability to detect a fault in the presence of noise and single disturbance (in contrast to two disturbances in the first scenario), then switch to a fallback configuration with guaranteed stability is demonstrated. In this case, the measurements of temperature and concentration are again assumed to contain noise of magnitude  $1\text{ K}$  and  $0.1\text{ kmol}/\text{m}^3$ , respectively. Also, the concentration of A in the inlet stream  $C_{A03}$  used in the process model is 10% smaller than the values used in the filter equations and the controller. The reactors, as well as the fault detection filter for the first control configuration are initialized at the desired steady state. As can be seen in Fig.3.11a, the controller maintains the closed-loop trajectory near the desired steady-state until heating jacket two ( $Q_2$ ) fails 40 *min* after reactor startup. If a fault-detection filter is not in place and the fault is not detected, closed loop stability is not achieved (dotted lines in fig. Fig.3.11a). The implemented fault-detection and isolation filter detects this fault when the value of the residual  $e_{2,1}(t)$  becomes greater than the threshold value of 2.0 at 41.33 min (see Fig.3.11c) while  $e_{1,1}(t)$  (Fig.3.11b) remains below the threshold of 2.0, allowing the detection and isolation of the fault. In this scenario, by the time that the fault is detected, the state of the closed-loop system resides within the stability region of configuration two ( $V_2 = 8.03 < c_2^{max} = 9.4$ ). therefore, the supervisor activates the fallback configuration with  $Q_1$  and  $Q_3$  as the manipulated inputs (solid lines in Fig.3.11a) and the control system is able to drive the process to the desired steady-state and enforce closed-loop stability.



(a)



(b)



(c)

Figure 3.12: (a) Temperature profile of reactor two with reconfiguration (solid line) and without reconfiguration (dotted line), (b)  $Q_1$  residual profile, and (c)  $Q_2$  residual profile (note fault detection at time  $t = 41.33 \text{ min}$ ).



**Remark 3.10:** In order to implement the fault detection and isolation filter on process systems accounting for noise, disturbances, and/or output feedback considerations one needs to decide on a value for the detection threshold for each individual residual. Given the complexity of the closed-loop system, there is not a simple and explicit way (formula) to directly calculate this threshold; a trial-and-error procedure needs to be followed. However, there are several things to consider when choosing an appropriate threshold value. The threshold should be chosen large enough so that noisy data, system disturbances, or discrepancies due to estimation error do not cause frequent false alarms. The threshold must also be chosen small enough so that at the time of detection the state of the system is within the stability region of a fallback configuration. These two considerations will give a reasonable range of threshold values to implement on the fault detection and isolation filter.

## Chapter 4

# Fault-tolerant control of nonlinear process systems subject to sensor faults

### 4.1 Introduction

In this chapter we consider the problem of fault-tolerant control of nonlinear process systems subject to input constraints and sensor faults (both complete failures and asynchronous measurements). We employ a reconfiguration-based approach, wherein, for a given process, a set of candidate control configurations are first identified, and in the event of a fault an appropriate backup configuration is activated to maintain stability. To illustrate the importance of accounting for the presence of constraints, we first consider sensor faults manifested as complete loss of measurements (faults that necessitate taking corrective action to repair the sensors). We address the problem of determining which candidate control configuration should be implemented in the closed-loop system to achieve stability after the sensor is recovered (this analysis is carried out under the assumption of continuous availability of measurements

when the sensor is functioning). We then consider the problem in the presence of intermittent sensor data losses. We define the sensor data loss rate to account for the presence of constraints (specifically, we define the data loss rate over a finite time interval) and analyze the stability properties in the presence of input constraints and sensor data losses. We characterize the stability region (that is, the set of initial conditions starting from where closed-loop stabilization under continuous availability of measurements is guaranteed) and the maximum allowable data loss rate that a given control configuration can tolerate. If the data loss rate goes above the allowable data loss rate, reconfiguration is triggered and a candidate backup configuration is activated for which the state of the closed-loop system resides in the stability region of the candidate configuration and the data loss rate is less than the allowable data loss rate for the candidate control configuration. We use a chemical reactor to illustrate our method and then demonstrate an application to a polyethylene reactor.

## 4.2 Preliminaries

We consider nonlinear processes with input constraints, described by:

$$\begin{aligned} \dot{x} &= f(x) + G_{k(t)}(x)u_{k(t)}(y(t)) \\ y(t) &= \begin{cases} x(t) & t \in [t_{2i}, t_{2i+1}) \\ x(t_{2i+1}) & t \in [t_{2i+1}, t_{2i+2}) \end{cases} \\ u_k &\in \mathbf{U}_k, k(t) \in \mathcal{K} = \{1, \dots, N\}, N < \infty \end{aligned} \tag{4.1}$$

where  $x \in \mathbb{R}^n$  denotes the vector of state variables,  $y \in \mathbb{R}^n$  denotes the vector of measured variables,  $[t_{2i}, t_{2i+1})$  and  $[t_{2i+1}, t_{2i+2})$  denote the time intervals during which measurements of the state variables are available, and are lost, respectively, with  $t_0 = 0$  (that is, measurement being initially available),  $u_{k(t)}(x) \in \mathbb{R}^m$  denotes the manipulated inputs under the  $k$ th configuration taking values in a nonempty convex subset  $\mathbf{U}_k$  of  $\mathbb{R}^m$ , where  $\mathbf{U}_k = \{u \in \mathbb{R}^m : \|u\| \leq u_k^{max}\}$ ,  $\|\cdot\|$  is the Euclidean

norm of a vector,  $u_k^{max} > 0$  is the magnitude of input constraints and  $f(0) = 0$ . The vector function  $f(x)$  and the matrix  $G_k(x) = [g_{1,k}(x) \cdots g_{m,k}(x)]$  are assumed to be sufficiently smooth on their domains of definition.  $k(t)$ , which takes values in the finite index set  $\mathcal{K}$ , represents a discrete state that indexes the matrix  $G_k(\cdot)$  as well as the manipulated input  $u_k(\cdot)$ . For each value that  $k$  assumes in  $\mathcal{K}$ , the system is controlled via a different set of manipulated inputs which defines a given control configuration. The notation  $L_f h$  denotes the standard Lie derivative of a scalar function  $h(\cdot)$  with respect to the vector function  $f(\cdot)$  and the notation  $x(T^-)$  denotes the limit of the trajectory  $x(t)$  as  $T$  is approached from the left, that is,  $x(T^-) = \lim_{t \rightarrow T^-} x(t)$ . Throughout the manuscript, we assume that for any  $u_k \in \mathbf{U}_k$  the solution of the system of Eq.4.1 exists and is continuous for all  $t$ .

We next review one example of a state feedback controller [24, 25] (inspired by the results on bounded control in [54]) that, under the assumption of continuous availability of measurements, provides an explicit estimate of the stability region for the closed-loop system subject to constraints (for more details on the controller design, and the proof, see [24, 25]).

**Theorem 4.1 [25]:** *Consider the nonlinear system of Eq.4.1 under state feedback (that is,  $x(t)$  is available for all  $t \geq 0$ ) for a configuration  $k$ , for which a Control Lyapunov Function  $V_k$  exists, under the following bounded nonlinear feedback controller:*

$$u_k = -w_k(x, u_k^{max})(L_{G_k} V_k(x))^T \quad (4.2)$$

where  $w_k(x, u_k^{max}) =$

$$\begin{cases} \frac{\alpha_k(x) + \sqrt{\alpha_k^2(x) + (u_k^{max} \|b_k^T(x)\|)^4}}{\|b_k^T(x)\|^2 \left[1 + \sqrt{1 + (u_k^{max} \|b_k^T(x)\|)^2}\right]}, & b_k^T(x) \neq 0 \\ 0, & b_k^T(x) = 0 \end{cases} \quad (4.3)$$

with  $\alpha_k(x) = L_{f_k}V_k(x) + \rho_k V_k(x)$ ,  $\rho_k > 0$  and  $b_k(x) = L_{G_k}V_k(x)$ . Assume that the set  $\Phi_k(u_k^{max})$  of  $x$  satisfying

$$L_{f_k}V_k(x) + \rho_k V_k(x) \leq u_k^{max} \|(L_{G_k}V_k(x))^T\| \quad (4.4)$$

contains the origin and a neighborhood of the origin. Also, let  $\Omega_k(u_k^{max}) := \{x \in \mathbb{R}^n : V_k(x) \leq c_k^{max}\}$  be a level set of  $V_k$ , completely contained in  $\Phi_k$ , for some  $c_k^{max} > 0$ . Then for all  $x(0) \in \Omega_k(u_k^{max})$  the control law of Eqs.4.2-4.4 guarantees that the origin of the closed-loop system is asymptotically stable.

**Remark 4.1:** The problems caused by input constraints have motivated numerous studies on the dynamics and control of systems subject to input constraints. Important contributions include results on optimization-based methods such as model predictive control (for example, [36, 92, 59]) and Lyapunov-based control (for example, [54, 91, 46, 51]). Stabilizing control laws that provide explicitly-defined regions of attraction for the closed-loop system have been developed using Lyapunov techniques; the reader may refer to [51] for a survey of results in this area. Recently, we developed a hybrid predictive control structure that employs switching between bounded control and MPC for stabilization of nonlinear systems [28], and nonlinear systems with uncertainty [64], subject to input constraints via using Lyapunov-based controllers [24, 25] as fall-back controllers. More recently Lyapunov-based model predictive controllers were designed that guarantee stabilization from an explicitly characterized set of initial conditions in the presence of input [63] and input and state [65] constraints. The controller of Eq.4.3 is one example of a controller design that provides an explicit characterization of the stability region in the presence of input constraints, and is only used to illustrate the main ideas behind the proposed approach. The results in this work are not limited to this particular controller design, and any other controller design that provides an explicit characterization of the stability region can be used

instead (for example, the hybrid predictive controller [28, 64] or the Lyapunov-based predictive controller [63, 65]; for further details and references, see [13]).

#### 4.2.1 A chemical reactor example

In this section, we describe a chemical reactor that we will use to illustrate the key features of our proposed method. To this end, consider a well-mixed, non-isothermal 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 non-isothermal nature of the reactions, a jacket is used to remove/provide heat to the reactor. Under standard modeling assumptions, a mathematical model of the process can be derived from material and energy balances and takes the following form:

$$\begin{aligned} \frac{dT}{dt} &= \frac{F}{V}(T_{A0} - T) + \sum_{i=1}^3 R_i(C_A, T) + \frac{Q}{\rho c_p V} \\ \frac{dC_A}{dt} &= \frac{F}{V}(C_{A0} - C_A) - \sum_{i=1}^3 k_{i0} e^{\frac{-E_i}{RT}} C_A \end{aligned} \quad (4.5)$$

where  $R_i(C_A, T) = \frac{(-\Delta H_i)}{\rho c_p} k_{i0} e^{\frac{-E_i}{RT}} C_A$ ,  $C_A$  denotes the concentrations of the species  $A$ ,  $T$  denotes the temperature of the reactor,  $Q$  denotes rate of heat input/removal from the reactor,  $V$  denotes the volume of the reactor,  $\Delta H_i$ ,  $k_i$ ,  $E_i$ ,  $i = 1, 2, 3$ , denote the enthalpies, pre-exponential constants and activation energies of the three reactions, respectively,  $c_p$  and  $\rho$  denote the heat capacity and density of fluid in the reactor. The values of the process parameters and the corresponding steady-state values can be found in [67]. It was verified that under these conditions, the system of Eq.4.5 has three steady-states (two locally asymptotically stable and one unstable

at  $(T_s, C_{As}) = (388 \text{ K}, 3.59 \text{ mol/L})$ .

The control objective considered here is that of stabilizing the reactor at the (open-loop) unstable steady-state using the measurements of concentration and temperature. The following manipulated input candidates are assumed to be available (see Fig.4.1):

1. Configuration 1: Rate of heat input,  $u_1 = Q$ , subject to the constraints  $|Q| \leq u_{max}^1 = 748 \text{ KJ/s}$ .
2. Configuration 2: Inlet stream temperature,  $u_2 = T_{A0} - T_{A0s}$ , subject to the constraints  $|u_2| \leq u_{max}^2 = 100 \text{ K}$ .
3. Configuration 3: Inlet reactant concentration,  $u_3 = C_{A0} - C_{A0s}$ , subject to the constraints  $|u_3| \leq u_{max}^3 = 4 \text{ mol/L}$ .

where configuration 2 will be used as the primary manipulated input.

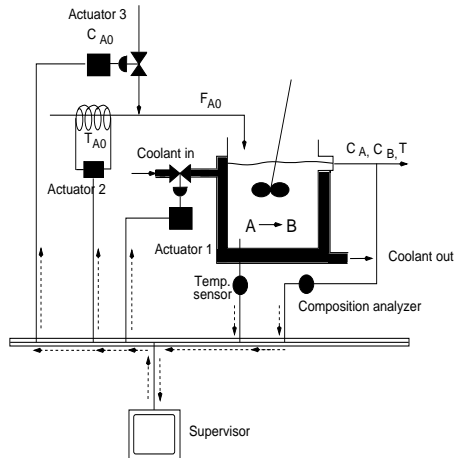


Figure 4.1: A schematic of the CSTR showing the three candidate control configurations.

We will use this chemical reactor to motivate our results. To this end, we consider the chemical reactor operating under a given control configuration. At a certain time one of the sensors fails in a way that it is imperative to recover the sensor to

implement feedback control. The problem that we analyze is whether reactivating the original control configuration (after sensor recovery) guarantees closed-loop stability. We will next consider the problem where the sensors do not fail, however, the process experiences intermittent loss of measurements (and this rate increases at a certain time due to sampling/measurement/communication errors. In this case, we want to know how much measurement data loss can be tolerated by the currently active control configuration, before it becomes necessary to reconfigure, and, if necessary, which backup configuration should be activated in the closed-loop system. Note that while we use the simple chemical reactor example only to motivate our results, the scenarios that we describe are relevant to all process operations. We also include an application to a more realistic process example, a polyethylene reactor, on the second example.

### 4.3 Stabilization subject to sensor failures

In this section, we consider the problem arising out of sensor failures that lead to the failure of the control loop and necessitate recovery. In analyzing this problem and in devising the fault-tolerant control strategy, we account for the presence of nonlinearity and constraints and show how they impact the reconfiguration logic.

#### 4.3.1 Reconfiguration law

Consider the closed-loop system of Eqs.4.1-4.4 for which candidate control configurations have been identified and the stability region under each candidate configuration has been explicitly characterized. Let the closed-loop system of Eqs.4.1-4.4 be initialized under a configuration  $k$  with  $x_0 \in \Omega_k$ . Let  $T^f$  be the time at which the sensor fails and  $T^r$  be the time at which the sensor recovers. In the absence of measurements,



the process runs open loop from the time  $T^f$  to  $T^r$ . Consequently, during this time the process state may drift further away from the desired operating condition. When the measurements become available again, switching to the original control configuration may not achieve closed-loop stability. The key consideration in devising the reconfiguration logic is the limitation imposed on the stability region under a given control configuration by the presence of input constraints and is formalized below:

**Theorem 4.2:** *Let  $k(0) = i$  for some  $i \in \mathcal{K}$  and  $x(0) := x_0 \in \Omega_i$ . Let  $T^f$  be the time that the sensor measurements become unavailable and let  $T^r$  be the earliest time that they become available again. Then, the following switching rule:*

$$k(t) = \left\{ \begin{array}{ll} i, & 0 \leq t < T^f \\ l, & t \geq T^r, x(T^r) \in \Omega_l \end{array} \right\} \quad (4.6)$$

*guarantees asymptotically stabilization of the origin of the closed-loop system.*

**Proof of Theorem 4.2:** We consider the two possible cases; first if no sensor failure occurs ( $T^f = \infty$ ), and second if a failure occurs at some finite time  $T^f$  and the sensors are recovered at time  $T^r$ .

*Case 1:* The absence of a failure implies  $k(t) = i \forall t \geq 0$ . Furthermore, since  $x(0) \in \Omega_i$ , and control configuration  $i$  is implemented for all times in this case, asymptotic stability follows from Theorem 4.1.

*Case 2:* At time  $T^r$ , the supervisor switches to a control configuration  $l$  for which  $x(T^r) \in \Omega_l$ . From this time onwards, since configuration  $l$  is implemented in the closed-loop system for all times, and since  $x(T^f) \in \Omega_l$ , once again, asymptotic stability follows from Theorem 4.1. This completes the proof of Theorem 4.2.

**Remark 4.2:** Theorem 4.2 accounts for the presence of constraints in the reconfiguration logic via the consideration of the stability region of candidate control configurations. Note that the problem that we consider here are sensor failures that result

in loss of controllability. For the sake of illustration, consider a linear system of the form  $\dot{x} = Ax + Bu; y = Cx$ , where  $x$  is the state vector,  $y$  is the vector of measured variables and  $u$  is the vector of manipulated variables, with  $A$ ,  $B$  and  $C$  being matrices of appropriate dimensions. Consider the case when all state variables are being measured ( $C = I$ ), and a state feedback law of the form  $u = Ky = Kx$  is used to stabilize the system. Further let some of the sensors fail at some time, resulting in a new  $C$  matrix denoted by  $\bar{C}$ . The same feedback gain matrix  $K$  may no longer be stabilizing. If  $\bar{C}$  is such that it can be used to reconstruct (estimate) the unstable states of the system (that is, all the unstable states remain observable) then feedback control (with an observer, and with a different feedback gain matrix) can still be used to stabilize the system. However if  $\bar{C}$  is such that some of the unstable states of the system become unobservable, then the system simply cannot be stabilized using feedback control, and fixing the sensors becomes imperative. In other words, it is when measurements become unavailable (due to individual sensor malfunction, or loss of communication lines) that result in loss of controllability, that it becomes imperative to detect, isolate and correct the problem. Due to the open-loop behavior of the process during this intermediate time, the process states may drift and go out of the stability region of the currently active control configuration. Reactivating the original control configuration may therefore not stabilize the closed-loop system making it necessary to ascertain the suitability of a candidate control configuration by using Theorem 4.2 (see the simulation example for a demonstration).

**Remark 4.3:** While in this work we do not focus on the problem of fault-detection and isolation (considering instead the problem of determining the corrective action that needs to be taken once the fault information is available), this problem has been approached using a data-based or a model-based strategy. Statistical and pattern

recognition techniques for data analysis and interpretation (for example, [85, 41, 4, 3, 61]), use past plant data to construct indicators that identify deviations from normal operation, and help in isolating faults. The problem of using fundamental process models for the purpose of detecting faults has been studied extensively in the context of linear systems [58, 31, 37]; and recently, some existential results in the context of nonlinear systems have been derived [76, 86, 19].

In [67] we proposed an integrated fault-detection and fault-tolerant control structure that handles faults in the control actuators under the assumption of continuous availability of state or output measurements. The fault-detection and isolation filter in [67] relies on the measurements to observe deviations of the process behavior from the expected closed-loop behavior to detect faults, and needs to be redesigned if required to detect and isolate faults in the sensors. While the problem of designing sensor fault-detection and isolation filter remains outside the scope of the present work, we note that the proposed fault-tolerant controller allows the use of any data- or model-based fault-detection and isolation filter to provide information about the occurrence of the fault (leading to its recovery). In this work we focus instead on determining what corrective action needs to be taken after a fault has been reported and how the time that it takes to recover the fault impacts on the reconfiguration logic. Specifically, the reconfiguration logic points to the necessity of recovering the sensor sufficiently fast to avoid the situation where the process state, by the time of recovery, has escaped the stability region of the backup configurations. Alternatively, the proposed method can also be used for the purpose of designing the control configurations in a way that maximizes the region in state space covered by the backup configurations to increase the chances that the process state at the time of recovery lies in the stability region of at least one backup configuration.

### 4.3.2 Application to the chemical reactor

In this section, we illustrate the utility of the reconfiguration law of Eq.4.6. To this end, consider the chemical reactor of Eq.4.5 with the three candidate control configurations available. The first step in implementing the reconfiguration law of Eq.4.6 is that of determining the stability regions of the individual control configurations under the control law of Eqs.4.2-4.4. An explicit characterization of the stability regions is obtained and is shown in Fig.4.2. The area indicated by I, II and III indicates the set of initial conditions starting from where all three configurations can stabilize the closed-loop system, I, II starting from where only configurations 1 and 2 can achieve stability and I, III indicate the set of initial conditions starting from where only configurations 1 and 3 can stabilize the closed-loop system.

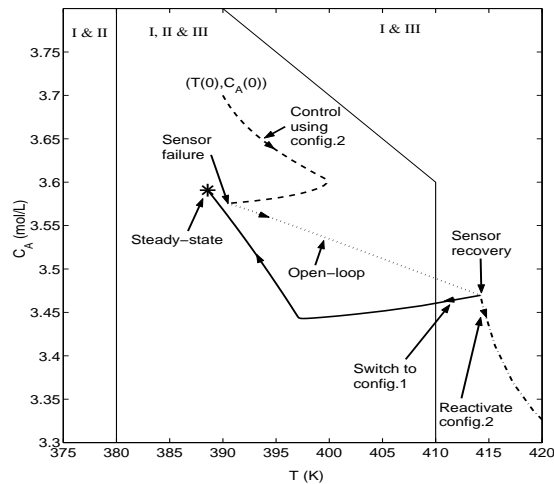


Figure 4.2: Evolution of the state profile under configuration 2 (dashed line) followed by loss of measurements (dotted line) and upon recovery reactivating configuration 2 (dash-dotted line), closed-loop stability is not preserved; however, switching to configuration 1 (solid line) preserves closed-loop stability.

The closed-loop system is initialized under configuration 2 from an initial condition belonging to the stability region of configuration 2. At  $t = 200$  min, however, a sensor

failure occurs resulting in open-loop operation, and the process state begins to drift away from the desired equilibrium point (see dotted line in Fig.4.2). Recognizing that it is imperative to rectify this fault, the sensors are recovered (alternatively, redundant sensors are activated) at  $t = 220$  min. With the state information again available, if the original control configuration (configuration 2) is reactivated, closed-loop stability is not achieved (see dash-dotted lines in Fig.4.2). This happens because during the time that the process was running open-loop, the states of the closed-loop system moved away from the desired equilibrium point and out of the stability region of configuration 2. In contrast, if the reconfiguration law of Eq.4.6 is used, the law dictates activation of configuration 1 (since the process state, when state information becomes available again, lies in the stability region of configuration 1). Closed-loop stability is subsequently achieved (solid line in Fig.4.2). Note that at the time the state information became available again, the state was also in the stability region of configuration 3, and switching to either configuration 1 or 3 would guarantee closed-loop stability. In such cases (when more than one control configurations satisfy the stability criteria), additional performance criteria, such as ease/cost of use can be used to decide which control configuration should be implemented in the closed-loop system [66].

#### **4.4 Stabilization subject to sensor data losses**

In the previous section, we considered the problem of devising the reconfiguration law in a way that accounts for the presence of constraints on the manipulated inputs under the available control configurations. We now consider the problem of intermittent sensor data losses (not complete failures) and develop a reconfiguration law that achieves fault-tolerant in the presence of sensor data-losses. As evidenced

in the previous section, a prerequisite to implementing fault-tolerant control is the characterization of the stability properties under the available control configurations, which we undertake in this section, and in the next section present the reconfiguration law. We consider the closed-loop system of Eqs.4.1-4.4 under a configuration  $k$  and drop the subscript  $k$  in the remaining of this section with the understanding that the robustness of the closed-loop system under control configuration  $k$  is being analyzed.

#### 4.4.1 Modeling sensor data loss

Preparatory to the analysis of the stability properties of the closed-loop system under sensor data losses, we describe how we model the occurrence of sensor data losses. Specifically, sensor data availability is modeled as a random Poisson process. At a given time  $t$  an ‘event’ takes place that determines whether the system will be closed-loop or open-loop (see Fig.4.3). For a given rate of data loss  $0 \leq r \leq 1$ , a random variable  $P$  is chosen from a uniform probability distribution between 0 and 1. If  $P \leq r$ , the event is deemed to be ‘measurement loss’, while if  $P > r$ , the event is understood to be ‘measurement available’. Furthermore, with  $W$  defined as the number of events per unit time, another random variable  $\chi$  with uniform probability distribution between 0 and 1 determines the time for which the current event will last, given by  $\Delta = \frac{-\ln\chi}{W}$ . At  $t + \Delta$  another event takes place and whether it represents a measurement or loss of measurement, as well as its duration, is similarly determined. Note that in the presence of constraints, prolonged duration of measurement loss may land the system states at a point starting from where stabilization may not be achievable (even with continuous measurement); in characterizing the stability properties of constrained systems, we therefore need to define data loss rates over a finite time interval as stated in assumption 1 below.

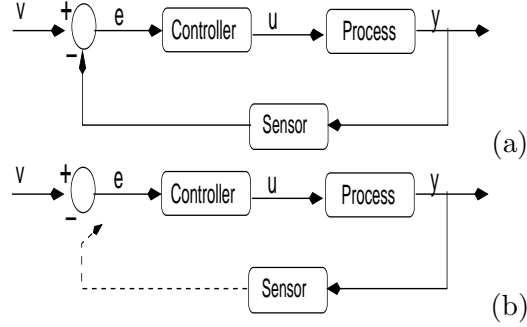


Figure 4.3: Closed-loop system in the (a) absence, and (b) presence of sensor data losses.

**Assumption 1:** For a positive real number  $T^*$ , defining  $r \in [0, 1]$  as the sensor data loss rate implies that over every successive finite time interval  $T^*$ , the measurements are available for a total time of  $T^* \times (1 - r)$ .

Note that assumption 1 does not impose any restrictions on the distribution of sequences of measurement loss and availability over the time interval  $T^*$ . Furthermore, the assumption does not need to hold for *any* finite interval  $T^*$  but only successive time intervals  $T^*$ . To illustrate the difference, consider the case where the assumption requires the data loss rate to hold over any finite time interval  $T^*$ , and that one such interval is  $\tau, \tau + T^*$ . Requiring the data loss rate to hold over any interval  $T^*$  would mean that the same data loss rate should also hold over the interval  $\tau + \epsilon_t, \tau + T^* + \epsilon_t$ , for *any* positive real number  $\epsilon_t$ , which can only be true if the data loss and measurement events are periodic with a period  $T^*$ . The requirement that the data loss rate hold over successive intervals  $T^*$  only says that over the time interval  $T^*$ , if the duration of all the measurement loss events is summed up, then that sum is equal to  $T^* \times r$ , and the data loss events could be distributed arbitrarily during this time interval. In simulating data losses, assumption 1 can be practically realized by picking  $W$  to be sufficiently large; the reasoning behind this is as follows: a larger value of  $W$  increases the number of events per unit time, and when  $W$  is sufficiently large, we

can get a sufficiently large number of events over every finite time interval  $T^*$  such that the rate of data loss is sufficiently close to  $r$ .

#### 4.4.2 Analyzing closed-loop stability

In this section, we consider the closed-loop system subject to sensor data losses as defined in previous section, and analyze the stability properties (robustness) with respect to sensor data losses. Specifically, the objective is to establish, for convergence to a desired neighborhood of the origin, a data loss rate  $r^*$ , defined over a finite time interval  $T$ , such that if  $r \leq r^*$  then convergence to a desired neighborhood is achieved in the presence of data losses. Note that implicit in this analysis (also in the formulation of Eq.4.1) is the understanding that during the time that sensor measurements are unavailable, the values of the measured variables (in computing the control action) are ‘frozen’ at the last available measurement. This results in the value of the manipulated variable being frozen at the last computed value. The implications of this intuitive assumption on the stabilizing properties under a given control configuration is discussed in Remark 4.5.

We first consider the closed-loop system under the controller of Eq.4.3, where the control action is computed in an implement and hold fashion with a hold time  $\Delta$ . We establish that for convergence to a desired neighborhood of the origin, there exists a bound on the implement and hold time  $\Delta^*$ , such that if the hold time is less than  $\Delta^*$ , then during the entire hold time, we get (outside of the desired neighborhood of the origin) that  $\dot{V} < 0$  (by virtue of the fact that the control action is ‘held’ at the value computed using the last available measurement) and eventual convergence to the desired neighborhood can be achieved. This analysis reveals that anytime the control action is ‘updated’ by using the current state value, the closed-loop Lyapunov-



function decreases during the next  $\Delta$  (for  $\Delta \leq \Delta^*$ ) time. In essence, it reveals that the worst distribution of the measurement loss events, or the most destabilizing that they can be, would be if they were to occur consecutively. The sum of the duration of all the measurement loss events not being greater than  $r \times T^*$  over a finite time interval  $T^*$  can be exploited to yield the desired result which is formalized in Theorem 4.3 below.

**Theorem 4.3:** *Consider the constrained system of Eq.4.1 under the bounded control law of Eqs.4.2-4.4 designed using the Lyapunov function  $V$  and  $\rho > 0$ , and the stability region estimate  $\Omega$  under continuous implementation. Then, given any positive real number  $d$  such that  $\|x\| \leq d$  implies  $x \in \Omega$  and  $T^*$  over which a data loss rate  $r$  is defined, there exists a positive real number  $r^*$  such that if  $x(0) := x_0 \in \Omega$  and is known, and  $r \in (0, r^*]$ , then  $x(t) \in \Omega \forall t \geq 0$  and  $\limsup_{t \rightarrow \infty} \|x(t)\| \leq d$ .*

**Proof of Theorem 4.3:** The proof consists of two parts. In the first part, we assume that the measurement loss events occur consecutively, and show the existence of a bound on the data loss rate  $r^*$  below which convergence to the desired neighborhood is achieved. In part 2, we show that this result also holds for any distribution of the open loop events over the time interval  $T^*$ .

*Part 1:* Substituting the control law of Eqs.4.2-4.4 into the system of Eq.4.1 it can be shown that:

$$\dot{V}(x) = -\rho^*V(x) \quad (4.7)$$

for all  $x \in \Omega$ , where  $\Omega$  was defined in Eq.4.4. Note that since  $V(\cdot)$  is a continuous function of the state, one can find a finite, positive real number,  $\delta'$ , such that  $V(x) \leq \delta'$  implies  $\|x\| \leq d$ . Consider now evolution of the states between the time 0 to  $T^*$ , where  $T^*$  is the time interval over which the data loss rate is defined, and for a given data loss rate  $r$ , denote the duration of open-loop operation as  $\Delta$ . In the rest of the proof, we show the existence of a positive real number  $\Delta^*$  such that all state trajectories

originating in  $\Omega$  converge to the level set of  $V$  ( $V(x) \leq \delta'$ ) for any value of  $\Delta \in (0, \Delta^*]$ . Hence we have that  $\limsup_{t \rightarrow \infty} \|x(t)\| \leq d$ . We then use the definition of the data loss rate to come up with an  $r^*$  to show that the result holds for any  $r \leq r^*$ .

To this end, consider a “ring” close to the boundary of the stability region, described by  $\mathcal{M} := \{x \in \mathbb{R}^n : (c^{max} - \delta) \leq V(x) \leq c^{max}\}$ , for a  $0 \leq \delta < c^{max}$ . Let the control action be computed for some  $x(0) := x_0 \in \mathcal{M}$  and, upon unavailability of subsequent measurements, held constant until a time  $\Delta^{**}$ , where  $\Delta^{**}$  is a positive real number ( $u(t) = u(x_0) := u_0 \forall t \in [0, \Delta^{**}]$ ) to be determined. Then,  $\forall t \in [0, \Delta^{**}]$ ,

$$\begin{aligned} \dot{V}(x(t)) &= L_f V(x(t)) + L_G V(x(t))u_0 \\ &= L_f V(x_0) + L_G V(x_0)u_0 \\ &\quad + (L_f V(x(t)) - L_f V(x_0)) \\ &\quad + (L_G V(x(t))u_0 - L_G V(x_0)u_0) \end{aligned} \tag{4.8}$$

Since the control action is computed based on the states in  $\mathcal{M} \subseteq \Omega$ ,  $L_f V(x_0) + L_G V(x_0)u_0 \leq -\rho^* V(x_0)$ . By definition, for all  $x_0 \in \mathcal{M}$ ,  $V(x_0) \geq c^{max} - \delta$ , therefore  $L_f V(x_0) + L_G V(x_0)u_0 \leq -\rho^*(c^{max} - \delta)$ .

Since the function  $f(\cdot)$  and the elements of the matrix  $G(\cdot)$  are continuous,  $\|u\| \leq u^{max}$ ,  $\mathcal{M}$  is bounded and  $L_f V(\cdot)$ ,  $L_G V(\cdot)$  are Lipschitz, then one can find, for all  $x_0 \in \mathcal{M}$ , positive real numbers  $\Delta^{**}$ ,  $K^1$ ,  $K^2$  and  $K^3$  such that  $\|x(\tau) - x_0\| \leq K^1 \Delta^{**}$  for all  $\tau \leq \Delta^{**}$ ,  $\|L_f V(x(\tau)) - L_f V(x_0)\| \leq K^3 K^1 \Delta^{**}$ ,  $\|L_G V(x(\tau))u_0 - L_G V(x_0)u_0\| \leq K^2 K^1 \Delta^{**}$  for all  $\tau \leq \Delta^{**}$ , and  $\Delta^{**} < \frac{\rho^*(c^{max} - \delta) - \epsilon}{(K^1 K^2 + K^1 K^3)}$  where  $\epsilon$  is a positive real number such that

$$\epsilon < \rho^*(c^{max} - \delta) \tag{4.9}$$

Using these inequalities in Eq.4.8, we get

$$\dot{V}(x(\tau)) \leq -\epsilon < 0 \forall 0 \leq \tau \leq \Delta^{**} \tag{4.10}$$

This implies that, given  $\delta'$ , if we pick  $\delta$  such that  $c^{max} - \delta < \delta'$  then if the control action is computed for any  $x \in \mathcal{M}$ , and the measurement loss time is less than  $\Delta^{**}$ ,

we get that  $\dot{V}$  remains negative during this time, and therefore the state of the closed-loop system cannot escape  $\Omega$  (since  $\Omega$  is a level set of  $V$ ). We now show the existence of  $\Delta'$  such that for all  $x_0 \in \Omega^f := \{x \in \mathbb{R}^n : V(x_0) \leq c^{max} - \delta\}$ , we have that  $x(\Delta) \in \Omega^u := \{x_0 \in \mathbb{R}^n : V(x_0) \leq \delta'\}$ , where  $\delta' < c^{max}$ , for any  $\Delta \in (0, \Delta']$ .

Consider  $\Delta'$  such that

$$\delta' = \max_{V(x_0) \leq c^{max} - \delta, u \in \mathcal{U}, t \in [0, \Delta']} V(x(t)) \quad (4.11)$$

Since  $V$  is a continuous function of  $x$ , and  $x$  evolves continuously in time, then for any value of  $\delta < c^{max}$ , one can choose a sufficiently small  $\Delta'$  such that Eq.4.11 holds. Let  $\Delta^* = \min\{\Delta^{**}, \Delta'\}$ . We now show that for all  $x_0 \in \Omega^u$  and  $\Delta \in (0, \Delta^*]$ ,  $x(t) \in \Omega^u$  for all  $t \geq 0$ .

For all  $x_0 \in \Omega^u \cap \Omega^f$ , by definition  $x(t) \in \Omega^u$  for  $0 \leq t \leq \Delta$  (since  $\Delta \leq \Delta'$ ). For all  $x_0 \in \Omega^u \setminus \Omega^f$  (and therefore  $x_0 \in \mathcal{M}$ ),  $\dot{V} < 0$  for  $0 \leq t \leq \Delta$  (since  $\Delta \leq \Delta^{**}$ ). Since  $\Omega^u$  is a level set of  $V$ , then  $x(t) \in \Omega^u$  for  $0 \leq t \leq \Delta$ .

We note that for  $x$  such that  $x \in \Omega \setminus \Omega^u$ , negative definiteness of  $\dot{V}$  is guaranteed for  $\Delta \leq \Delta^* \leq \Delta^{**}$ . Finally, for all  $\Delta^* \leq t \leq T^*$ , negative definiteness of  $\dot{V}$  is guaranteed by the control law of Eq.4.3. Now for a given value of  $T^*$ , the worst case scenario (that is, the maximum time over which the system may run open-loop) involves loss of measurements for the last  $\Delta$  time for a given interval, followed by consecutive loss of measurements for the first  $\Delta$  time of the next interval. Therefore, continued negative definiteness of  $V$  (and convergence to the desired neighborhood) can be guaranteed if the measurement loss time in each interval  $\Delta \leq \frac{\Delta^*}{2}$ . An  $r^* = \frac{\Delta^*}{2T^*}$  will ensure that the maximum duration of measurement loss over the interval  $T^*$  is less than  $\Delta^*/2$ , and also maximum loss of measurement between two successive intervals is less than  $\Delta^*$  ( If  $\frac{\Delta^*}{2} > T^*$ , then we have to restrict  $r^*$  to 1 to ensure that  $r < 1$  and that we get at least one measurement over the entire interval  $T^*$ ). Therefore, for all  $x(0) \in \Omega$ ,

there exists an  $r^*$  such that if  $r \leq r^*$ ,  $\limsup_{t \rightarrow \infty} V(x(t)) \leq \delta'$ . Finally, since  $V(x) \leq \delta'$  implies  $\|x\| \leq d$ , therefore we have that  $\limsup_{t \rightarrow \infty} \|x(t)\| \leq d$ .

*Part 2:* Consider now the finite time interval  $T^*$ , such that for convergence to a desired neighborhood of the origin, the bound on the data loss rate  $r^*$ , under the assumption that the data-loss events all occur consecutively, has been computed. Consider now that the data-loss events do not occur continuously, but occur in  $N$  intervals, each of duration  $\Delta_i$  with  $\sum_{i=1}^N \Delta_i = T^* \times r^*$ . From part 1 above, for each of these durations  $\Delta_i$ , negative definiteness of  $\dot{V}$  can be established. For the duration during which the measurements are available,  $\dot{V} < 0$  is achieved by virtue of the control law. In summary, having established the bound  $r^*$  under consecutive loss of measurement, the same bound  $r^*$  continues to guarantee practical stability irrespective of the distribution of the measurement loss events. This completes the proof of Theorem 4.3.

**Remark 4.4:** Note that one can easily remove the assumption that  $x_0$  is known by ‘stepping back’ from the boundary of the stability region enough to ensure that during the time  $r^*T^*$ , the state trajectory cannot escape the boundary of the stability region. By the definition of rate of data loss, the first measurement is guaranteed to be available by  $(r^*T^*)^+$ . Any time during the interval  $T^*$  that a measurement is received with the state still residing in the stability region (due to the ‘stepping back’) Theorem 4.3 can be used to establish practical stability. Note also, that the value of  $r^*$  depends on the interval  $T^*$  over which it is defined (see the simulation example in section 4.4.3 for a demonstration). To understand this more clearly, let us revisit the proof of Theorem 4.3. It can be seen that for convergence to a desired neighborhood of the origin, one can come up with a value  $\Delta^*$  such that if only one measurement was received every  $\Delta^*$ , then convergence to the desired neighborhood would be achieved. Theorem 4.3 exploits this fact together with the definition of the

data loss rate, to ensure that over a  $\Delta^*$  duration within  $T^*$  (and across two time intervals), at least one measurement is received. In summary,  $\Delta^*$  is fixed by the given size of the neighborhood to the origin where convergence is desired ( $\delta'$ ); given a  $T^*$  over which the data loss rate is defined,  $r^*$  can then in turn be picked such that the maximum duration of open-loop behavior across intervals stays less than  $\Delta^*$ .

**Remark 4.5:** In our results, no bound on the open-loop instability is assumed to be known, leading to practical (and not asymptotic) stability to the desired equilibrium point. If additional assumptions are made on the open-loop growth of the Lyapunov-function (locally) around the desired equilibrium point, asymptotic stability can be shown using the same line of reasoning as in [42]. Specifically, during the time that the measurements are not available, the value of  $V$  is allowed to increase during  $T^*$ , so long as the increase in  $V$  can be ‘countered’ by the decrease in  $V$  during the rest of the time (which relies on assuming a known measure of open-loop instability). The limitations imposed by the presence of constraints, however, would still need to be accounted for, with the data loss rate having to be defined over a finite interval. Furthermore, the set of stabilizable initial conditions will only be a subset of  $\Omega$  such that starting from this subset, the closed-loop state can not escape  $\Omega$  during the time of open-loop evolution  $r^*T^*$ . In our results, with  $x_0$  known,  $r^*$  is picked so that  $\dot{V}$  stays negative during the entire duration of  $T^*$  (until convergence to the desired neighborhood is achieved), thereby obviating the need to restrict the set of initial conditions to a subset of  $\Omega$ . Note also that  $V$  being allowed to increase during  $T^*$  (as long as it decays by the end of  $T^*$ ) could possibly lead to a larger allowable  $r^*$ . The tradeoff would be that the Lyapunov function would not be guaranteed to decay all the time but only to decay in value at steps of  $T^*$ , and it could take longer to reach the desired neighborhood of the origin. Note that the problem considered in this work is not that of ascertaining

finite-time stability (ensuring convergence to the desired equilibrium point in finite time, see, for example, [8]) under continuous availability of measurement but rather that of analyzing preservation of stability under asynchronous measurements. Note that for the case when sensor measurements are lost but it is possible to change the value of the manipulated input, statistical (e.g., [75]) or first principles model based methods designed to ‘fill-in’ the unavailable state measurement can very well be included within the proposed framework, and can serve to improve the data-loss handling capabilities of the control designs (depending upon the accuracy of the data prediction). The proposed fault-tolerant control structure, however, addresses a more general problem, that of intermittent loss of communication between the controller and the process, including asynchronous measurements as well as the inability to change the manipulated input value during the communication lapses.

**Remark 4.6:** The proof of theorem 4.3 relies on the stabilizing properties of the controller during the time that measurements are not available to ensure that even during that time,  $\dot{V} < 0$ . Note that the rate of decay of the Lyapunov function that is achieved under continuous measurements is closely related to how much data loss can be tolerated in the system in the sense that for a given process and constraints on the manipulated inputs, if one control law achieves greater decay of the Lyapunov function over the other, then it can tolerate greater sensor data loss compared to the other (note that the tradeoff could be a smaller stability region estimate). The continued decay of the Lyapunov function, however, can only be achieved over a finite time, and in turn, requires the data loss rate to be defined over a finite time. Even if one were to use the approach discussed in Remark 4.4 to come up with an alternate bound, the limitations imposed by the constraints on the definition of the rate of data loss (specifically, the need to define it over a finite time interval) would be

present and can be understood as follows: If there were no constraints,  $\dot{V} < 0$  under continuous measurement could possibly be achieved over the entire state space. No matter how ‘far’ the states go during the unavailability of measurements, when (over the infinite time duration) the measurements do become available, one could require them to be available for a large enough time (compared to the time during which they were not available) to achieve an overall reduction in the value of the Lyapunov function. Constraints, however, limit the set of initial conditions (estimated using the stability region  $\Omega$ ) starting from where  $\dot{V} < 0$  is achievable. If the measurements are not available for a large duration, the states may go too ‘far’ (that is, out of the stability region) and then even if measurements were available for all time after that,  $\dot{V} < 0$  could not be achieved simply due to limited available control action (see the simulation example for a demonstration). In contrast, defining the data loss rate over a finite time interval enables restricting the states to stay within the region from where  $\dot{V} < 0$  and hence closed-loop stability is achievable.

**Remark 4.7:** Note that the specific problem that this work considers yields a solution that is essentially different from, and cannot be handled by simply using adaptive or other robust control approaches. These approaches, however, can very well be integrated within the proposed framework. The key requirement being that the controller design (whether it be an adaptive control design or another robust controller design) for the individual control configuration allow for an explicit characterization of its stability properties in the presence of input constraints and asynchronous data losses. It is this characterization that can be subsequently used in fault-tolerant reconfiguration strategies. Note also that multi-rate data loss problems, where data is available at predetermined (but different) times for the different measurements can be analyzed as special cases for the problem considered in the present work which does not assume

data availability at predetermined rates.

#### 4.4.3 Control of a chemical reactor subject to sensor data loss

Consider the chemical reactor of Eq.4.5 again with the inlet stream temperature, as the manipulated input  $u_2 = T_{A0} - T_{A0s}$ , subject to the constraints  $|u_2| \leq u_{max}^2 = 100$  K, and subject to measurement data losses. We first design the bounded controller and estimate the stability region (see Fig.4.4). For a given value of  $T^* = 10$  minutes, we pick a value of  $W = 10$  events per minute (the simulations are run as discussed in section 4.4.1); which yields an overall event rate of  $1/W$  that is, about one event every six seconds (or about 100 events in 10 minutes). It was verified that with this value of  $W$ , the rate of data loss, as defined, was approximately achieved over the duration of every ten minutes, in other words, that  $W = 10$  is a sufficiently large value of  $W$ . Starting from an initial condition within the stability region of the first configuration, the closed-loop system is unstable with a data loss rate  $r = 0.4$  (dashed lines in Fig.4.4; the corresponding manipulate input profile can be seen in Fig.4.5). However, if the data loss rate is kept at 0.1, closed-loop stability is achieved (see solid lines in Figs.4.4-4.5), demonstrating the need for the data loss to be sufficiently small.

The next simulation run demonstrates the dependence of  $r^*$  on the time interval over which it is defined (as discussed in Remark 4.6). Specifically, we now run the same simulation with an even smaller data loss rate ( $r = 0.05$ ), however, with the data rate defined over the duration of the simulation of 68 minutes. A scenario where measurements are received continuously for the first five minutes, lost consecutively for the next 3.6 minutes, and received thereafter results in an overall rate of data loss of only 0.05. We see however, that closed-loop stability is not achieved (dash-dotted lines in Figs.4.4-4.5). This is so because with this larger value of  $T^*$ , the acceptable



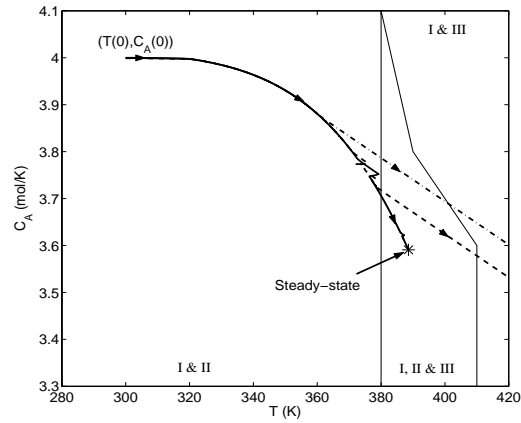


Figure 4.4: Evolution of the state trajectory under control configuration 2 in the presence of sensor data loss (defined over a finite interval) at a rate of 0.4 (dashed line), sensor data loss (defined over an infinite interval) at a rate of 0.05 (dash-dotted line) and sensor data loss (defined over a finite interval) at a rate of 0.1 (solid line).

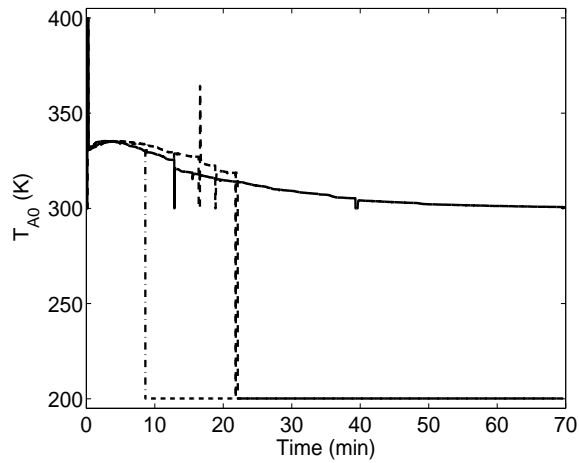


Figure 4.5: Manipulated input profile under control configuration 2 in the presence of sensor data loss (defined over a finite interval) at a rate of 0.4 (dashed line), sensor data loss (defined over an infinite interval) at a rate of 0.05 (dash-dotted line) and sensor data loss (defined over a finite interval) at a rate of 0.1 (solid line).

bound on the rate of data loss decreases, and illustrates the interconnection between the maximum allowable data loss rate and the interval over which it is defined. In summary, the above simulations demonstrate the need for the data loss rate to be less than what the system can tolerate (that is, for  $r \leq r^*$ ), with  $r^*$  appropriately computed for a given time interval  $T^*$  over which the rate is defined.

## 4.5 Fault-tolerant control subject to sensor data losses

Having analyzed the stability properties of the individual control configurations subject to sensor data losses, in this section we present a fault-tolerant controller that maintains closed-loop stability in the presence of sensor data losses.

### 4.5.1 Reconfiguration law

Fault-tolerance is achieved via switching to a backup configuration for which the state of the closed-loop system is within the stability region, and the sensor data loss rate is less than the bound on the data loss rate required for closed-loop stability. To formalize this idea, consider the constrained nonlinear system of Eq.4.1 for which the bounded controllers of the form of Eq.4.3 have been designed and the stability regions  $\Omega_j$ ,  $j = 1, \dots, N$  have been explicitly characterized under each control configuration, and the bounds on the data loss rate  $r_j^*$ ,  $j = 1, \dots, N$  have been computed. Let  $d_{max} = \max_{j=1, \dots, N} d_j$ , where  $d_j$  was defined in Theorem 4.3 and let  $\Omega_U = \bigcup_{j=1}^N \Omega_j$ . We consider the problem where the process starts operating under configuration  $i$  with a data loss rate of  $r_i(0)$ , and at some point in time the data loss rate  $r(t)$  possibly becomes greater than  $r_i^*$ .

**Theorem 4.4:** *Let  $k(0) = i$  for some  $i \in \mathcal{K}$  and  $x(0) := x_0 \in \Omega_i$ . Let  $T^f$  be the earliest time such that  $r(t) > r_i^*$  with  $x(T^f)$  measured. Then, the following switching*

rule:

$$k(t) = \left\{ \begin{array}{ll} i, & 0 \leq t < T^f \\ l, & t \geq T^f, x(T^f) \in \Omega_l, r(T^f) \leq r_l^* \end{array} \right\} \quad (4.12)$$

and  $r(t) \leq r_l^* \forall t \geq T^f$  guarantees that  $x(t) \in \Omega_U \forall t \geq 0$  and  $\limsup_{t \rightarrow \infty} \|x(t)\| \leq d_{max}$ .

**Proof of Theorem 4.4:** We consider the two possible cases; first if the data loss rate  $r$  stays less than or equal to  $r_i^*$  for all times, and second if  $r > r_i^*$  at some time  $T^f$ .

*Case 1:* The absence of a switch implies  $k(t) = i \forall t \geq 0$ . Furthermore, since  $x(0) \in \Omega_i$ ,  $r(t) \leq r_i^*$  and control configuration  $i$  is implemented for all times in this case, we have that  $x(t) \in \Omega_i \forall t \geq 0$  and  $\limsup_{t \rightarrow \infty} \|x(t)\| \leq d_i$ . Finally, since  $\Omega_i \subseteq \Omega_U$  and  $d_i \leq d_{max}$ , we have that  $x(t) \in \Omega_U \forall t \geq 0$  and  $\limsup_{t \rightarrow \infty} \|x(t)\| \leq d_{max}$ .

*Case 2:* At time  $T^f$ , the supervisor switches to a control configuration  $l$  for which  $x(T^f) \in \Omega_l$  and  $r \leq r_l^*$ . From this time onwards, since configuration  $l$  is implemented in the closed-loop system for all times, and since  $x(T^f) \in \Omega_l$  and  $r(t) \leq r_l^*$ , we have that  $x(t) \in \Omega_l \forall t \geq 0$  and  $\limsup_{t \rightarrow \infty} \|x(t)\| \leq d_l$ . As in case 1, since  $\Omega_l \subseteq \Omega_U$  and  $d_l \leq d_{max}$ , we have that  $x(t) \in \Omega_U \forall t \geq 0$  and  $\limsup_{t \rightarrow \infty} \|x(t)\| \leq d_{max}$ . This completes the proof of Theorem 4.4.

**Remark 4.8:** Theorem 4.4 explicitly takes into consideration the constraints in the manipulated inputs and the measurement losses in deciding which backup configuration to implement in the closed-loop system, and therefore requires that a backup configuration is implemented for which the state resides in its stability region *and* the data loss rate is less than the data loss rate that the backup configuration can tolerate. Disregarding either of these factors could lead to instability (see the simulation example for a demonstration).

**Remark 4.9:** Note that the result of Theorem 4.4 assumes explicit knowledge of

the current data loss rate to not only identify the appropriate backup configuration but also to trigger reconfiguration. In this sense, the reconfiguration logic has an in-built fault detection mechanism, with faults being defined as data loss rate exceeding the allowable data loss rate. In practice, the data loss rate can only be estimated over finite intervals of time, and this estimate can be used in deciding which backup configuration should be activated according the reconfiguration rule of Theorem 4.4. Note also, that other than the data loss rate (estimate) going over the allowable bound, other means of detecting instability like behavior (such as the state trajectory going close to the boundary of the stability region under the currently-active control configuration) can be used to trigger the reconfiguration. It is worth pointing out, however, that this fault-detection capability is only limited to the rate of data loss exceeding the tolerable value. As discussed in Remark 4.3, explicit fault detection mechanisms which detect faults in the sensors (such as sensors reporting incorrect values) can be used within the proposed approach to tackle sensor faults manifested as erroneous measurements.

**Remark 4.10:** While we assume the availability of measurements of all the state variables, the same approach can be used to analyze the case where each control configuration is comprised of a set of sensors and actuators with the sensors (measurements) different in different control configurations. Specifically, under each control configuration, an estimation scheme, coupled with the feedback controller, will have to be implemented and the output feedback stability region, subject to constraints and sensor data losses characterized. Subsequently, the reconfiguration rule will have to be modified to account for the fact that the reconfiguration decision is made on the basis of state estimates (which may contain errors); for a switching scheme that addresses these issues in the context of switched nonlinear systems under continuous

output feedback control, see [29].

#### 4.5.2 Fault-tolerant control of a chemical reactor

Consider, once again the chemical reactor of section 4.4.3 in the presence of sensor data losses. As seen in section 4.4.3, the closed-loop system using configuration 2 experiences instability when the data loss rate becomes 0.4. In the event of such data losses, one of the backup control configurations need to be activated and this choice cannot be made only by looking at the states with respect to the stability region. In this section we demonstrate the application of the switching rule of Theorem 4.4 that achieves fault-tolerance. To this end, we first characterize the stability region under each backup configuration. Fig.4.6 depicts the stability region, in the  $(T, C_A)$  space, for each configuration. The desired steady-state is depicted with an asterisk that lies in the intersection of the three stability regions. For configurations 1, 2 and 3, the bound on the data loss rate is estimated at  $r_1^* = 0.35$ ,  $r_2^* = 0.3$  and  $r_3^* = 0.15$ , respectively.

We consider an initial condition,  $T(0) = 300 \text{ K}$ ,  $C_A(0) = 4.0 \text{ mol/L}$ ,  $C_B(0) = 0.0 \text{ mol/L}$ , using the  $T_{A0}$ -control configuration within the stability region of configuration 2, and consider a case where the rate of sensor data loss increases from an initial value of 0.1 to 0.35. As shown by the solid line in Fig.4.6, the controller proceeds to drive the closed-loop trajectory towards the desired steady-state, up until time 13.5 minutes of reactor startup when the sensor data loss rate increases to 0.35. If the supervisor does not use the result of Theorem 4.4 to trigger reconfiguration, but persists with using configuration 2, stability is not achieved (see dotted lines in Figs.4.6-4.7). Note that at this time, the state of the closed-loop system resides in the stability region of both backup configurations 1 and 3. If the supervisor does

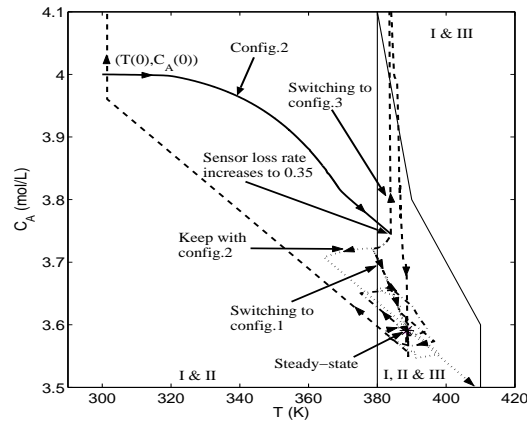


Figure 4.6: Evolution of the state trajectory: At  $t = 13.5$  minutes the data loss rate goes up to 0.35 under configuration 2 (solid line). Keeping with configuration 2 (dotted line) or switching to configuration 3 (dashed line) does not preserve stability, while switching to configuration 1 (dash-dotted line) preserves stability.

implement reconfiguration, but in a way that does not account for the presence of sensor data loss and activates configuration 3, the state trajectory does not converge to the desired steady-state (see dashed line in Fig.4.6) even though the state at the switching time is within stability region of control configuration 3. This happens because the rate of data loss is not within the tolerable bound for configuration 3. In contrast, if the reconfiguration rule of Eq.4.12 is implemented, and the supervisor activates configuration 1, the state trajectory converges to the desired steady-state (see dashed-dotted line in Fig.4.6). The corresponding manipulated input profiles are shown in Fig.4.7.

### 4.5.3 Fault-tolerant control of a polyethylene reactor subject to sensor data loss

Having demonstrated the application of the proposed fault-tolerant controller on the illustrative example, we next consider a more complex process, specifically, an in-

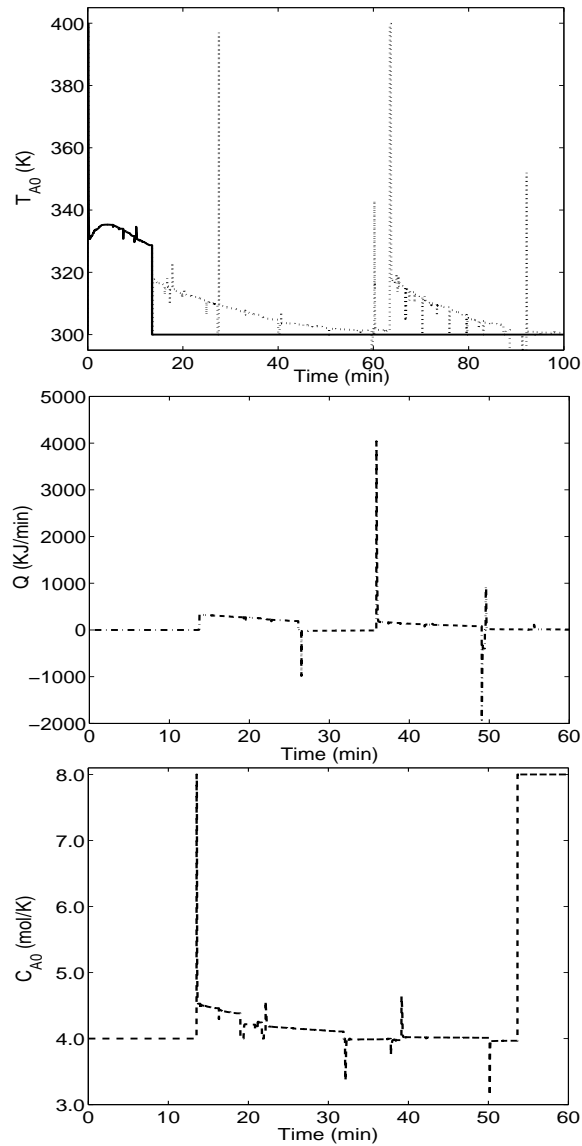


Figure 4.7: Manipulate input profiles: At  $t = 13.5$  minutes the data loss rate goes up to 0.35 under configuration 2 (solid line), switching to configuration 3 does not preserve stability (dashed line), while switching to configuration 1 (dash-dotted line) preserves stability.

dustrial gas phase polyethylene reactor system (see Fig.4.8 for a schematic). This reactor was also studied in [35] in the context of faults in the control actuator (under assumption of continuous availability of process measurements).

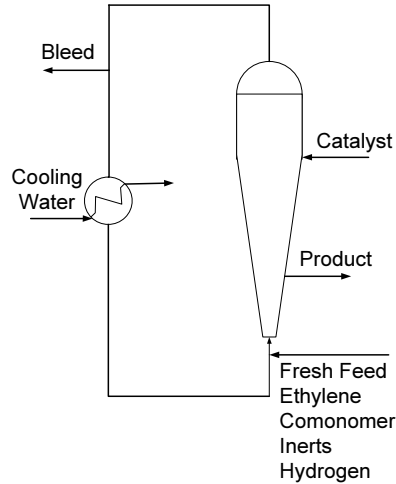


Figure 4.8: Industrial gas phase polyethylene reactor system.

The feed to the reactor consists of ethylene, comonomer, hydrogen, inerts, and catalyst. A stream of unreacted gases flows from the top of the reactor and is cooled by passing through a heat exchanger in counter-current flow with cooling water. Cooling rates in the heat exchanger are adjusted by instantaneously blending cold and warm water streams while maintaining a constant total cooling water flowrate through the heat exchanger. Mass balance on hydrogen and comonomer have not been considered in this study because hydrogen and comonomer have only mild effects on



the reactor dynamics [60]. A mathematical model for this reactor has the form [15]:

$$\begin{aligned}
\frac{d[In]}{dt} &= \frac{F_{In} - \frac{[In]}{[M_1] + [In]} b_t}{V_g} \\
\frac{d[M_1]}{dt} &= \frac{F_{M_1} - \frac{[M_1]}{[M_1] + [In]} b_t - R_{M_1}}{V_g} \\
\frac{dY_1}{dt} &= F_c a_c - k_{d_1} Y_1 - \frac{R_{M_1} M_{W_1} Y_1}{B_w} \\
\frac{dY_2}{dt} &= F_c a_c - k_{d_2} Y_2 - \frac{R_{M_1} M_{W_1} Y_2}{B_w} \\
\frac{dT}{dt} &= \frac{H_f + H_{g_1} - H_{g_0} - H_r - H_{pol}}{M_r C_{pr} + B_w C_{ppol}} \\
\frac{dT_{w_1}}{dt} &= \frac{F_w}{M_w} (T_{wi} - T_{w_1}) - \frac{UA}{M_w C_{pw}} (T_{w_1} - T_{g_1}) \\
\frac{dT_{g_1}}{dt} &= \frac{F_g}{M_g} (T - T_{g_1}) + \frac{UA}{M_g C_{pg}} (T_{w_1} - T_{g_1})
\end{aligned} \tag{4.13}$$

where

$$\begin{aligned}
b_t &= V_p C_v \sqrt{([M_1] + [In]) \cdot RR \cdot T - P_v} \\
R_{M_1} &= [M_1] \cdot k_{p0} \cdot \exp\left[\frac{-E_a}{R} \left(\frac{1}{T} - \frac{1}{T_f}\right)\right] \cdot (Y_1 + Y_2) \\
C_{pg} &= \frac{[M_1]}{[M_1] + [In]} C_{pm1} + \frac{[In]}{[M_1] + [In]} C_{pIn} \\
H_f &= F_{M_1} C_{pm1} (T_{feed} - T_f) + F_{In} C_{pIn} (T_{feed} - T_f) \\
H_{g_1} &= F_g (T_{g_1} - T_f) C_{pg} \\
H_{g_0} &= (F_g + b_t) (T - T_f) C_{pg} \\
H_r &= H_{reac} M_{W_1} R_{M_1} \\
H_{pol} &= C_{ppol} (T - T_f) R_{M_1} M_{W_1}
\end{aligned} \tag{4.14}$$

For the definition of all the variables used in Eqs.4.13-4.14 and the values of the process parameters see [35]. The open-loop system at the nominal operating condition exhibits an unstable equilibrium point surrounded by a limit cycle. The control objective is to stabilize the reactor using measurements of the state variables. To accomplish this objective we consider the following manipulated input candidates:

1. Catalyst flowrate,  $u_1 = (F_c - F_c^s)a_c$ , subject to the constraint  $|u_1| \leq u_{max}^1 = (\frac{2}{3600})a_c \frac{mol}{s}$ .
2. Feed temperature,  $u_2 = \frac{F_{M_1}C_{pm1}+F_{In}C_{pIn}}{M_rC_{pr}+B_wC_{ppol}}(T_{feed} - T_{feed}^s)$ , subject to the constraint  $|u_2| \leq u_{max}^2 = \frac{F_{M_1}C_{pm1}+F_{In}C_{pIn}}{M_rC_{pr}+B_wC_{ppol}}(20) \frac{K}{s}$ .

First, process operation under primary control configuration was considered (that is, the catalyst flowrate,  $F_c$ , was the manipulated input) and a bounded nonlinear controller was designed using the formula of Eqs.4.2-4.4. Specifically, a quadratic function of the form  $V_1 = e_1^T P_1 e_1$  and  $\rho_1 = 0.01$  were used to design the controller and a composite Lyapunov function of the form  $V_{c_1} = 5 \times 10^{-3}(In - In_s)^4 + 5 \times 10^{-4}(M_1 - M_{1s})^2 + 5 \times 10^{-11}(Y_1 - Y_{1s})^2 + 5 \times 10^{-11}(Y_2 - Y_{2s})^2 + 5 \times 10^{-4}(T - T_s)^2 + 5 \times 10^{-11}(T_{w_1} - T_{w_{1s}})^2 + 5 \times 10^{-11}(T_{g_1} - T_{g_{1s}})^2$  was used to estimate the stability region of the primary control configuration yielding a  $c_1^{max} = 56.8$ . A quadratic Lyapunov function of the form  $V_2 = \frac{1}{2}(T - T_s)^2$  and  $\rho_2 = 0.01$  were used to design the controller that used the fall-back control configuration (that is, the feed temperature,  $T_{feed}$ , was the manipulated input) and a composite Lyapunov function of the form  $V_{c_2} = 5 \times 10^{-3}(In - In_s)^4 + 5 \times 10^{-4}(M_1 - M_{1s})^2 + 5 \times 10^{-11}(Y_1 - Y_{1s})^2 + 5 \times 10^{-11}(Y_2 - Y_{2s})^2 + 5 \times 10^{-4}(T - T_s)^2 + 5 \times 10^{-2}(T_{w_1} - T_{w_{1s}})^2 + 5 \times 10^{-11}(T_{g_1} - T_{g_{1s}})^2$  was used to estimate the stability region of the fall-back control configuration yielding a  $c_2^{max} = 62$ .

Fig.4.9 shows the evolution of the closed-loop state profiles under continuous measurement (solid lines) starting from the initial condition  $In(0) = 450 \frac{mol}{m^3}$ ,  $M_1(0) = 340 \frac{mol}{m^3}$ ,  $Y_1(0) = 4.6 mol$ ,  $Y_2(0) = 4.6 mol$ ,  $T(0) = 360 K$ ,  $T_{w_1}(0) = 300 K$ , and  $T_{g_1}(0) = 300 K$  for which  $V_{c_1} = 56.78$ . Since this initial state is within the stability region of the primary control configuration (that is,  $V_{c_1}(x(0)) \leq c_1^{max}$ ), the primary control configuration is able to stabilize the system at the steady-state of interest.

The corresponding manipulated inputs are shown on Figs.4.10-4.11. The dynamics of the process also reveal an important feature regarding tolerance to sensor data losses. Specifically, for this particular process, even under no control (equivalent to complete data loss), the process goes to a limit cycle which is within the stability region for the closed-loop system under continuous availability of measurements. This characteristic impacts positively on the tolerance of the closed-loop system to data losses, and a high sensor data loss rate of 0.75 ends up being tolerable (see dotted lines in Figs.4.9 & 4.11), even with the value of the manipulated input variable set to the nominal value during the time that the measurements are unavailable (equivalent to open-loop operation).

Consider now a case where the rate of sensor data loss increases from an initial value of 0.75 to 0.80 at 0.97 hour of reactor startup. As shown by the dashed lines in Fig.4.12, the controller proceeds to drive the closed-loop trajectory towards the desired steady-state up until 0.97 hours. If the supervisor does not account for the increase of sensor data loss and continues utilizing the primary control configuration to control the reactor, the state trajectory does not converge to the desired steady-state (see Fig.4.12) even though the state at the time that the data loss rate increases is within the stability region of the primary configuration ( $V_{c_1}(x(t = 0.97hour)) = 1.6380 \leq c_1^{max}$ ). This happens because the rate of data loss is not within the tolerable bound for primary control configuration ( $r > r_1^* = 0.75$ ).

In this case, the supervisor had available a fall-back control configuration with the feed temperature as the manipulated input. At time 0.97 hour when sensor data loss rate increases from 0.75 to 0.80,  $V_{c_2} = 1.6382$  implying that the state of the closed-loop system resides in the stability region of the fall-back configuration (that is,  $V_{c_2}(x(t = 0.97hour)) \leq c_2^{max}$ ) as well as  $r \leq r_2^* = 0.95$ . If the reconfiguration rule of Eq. 4.12

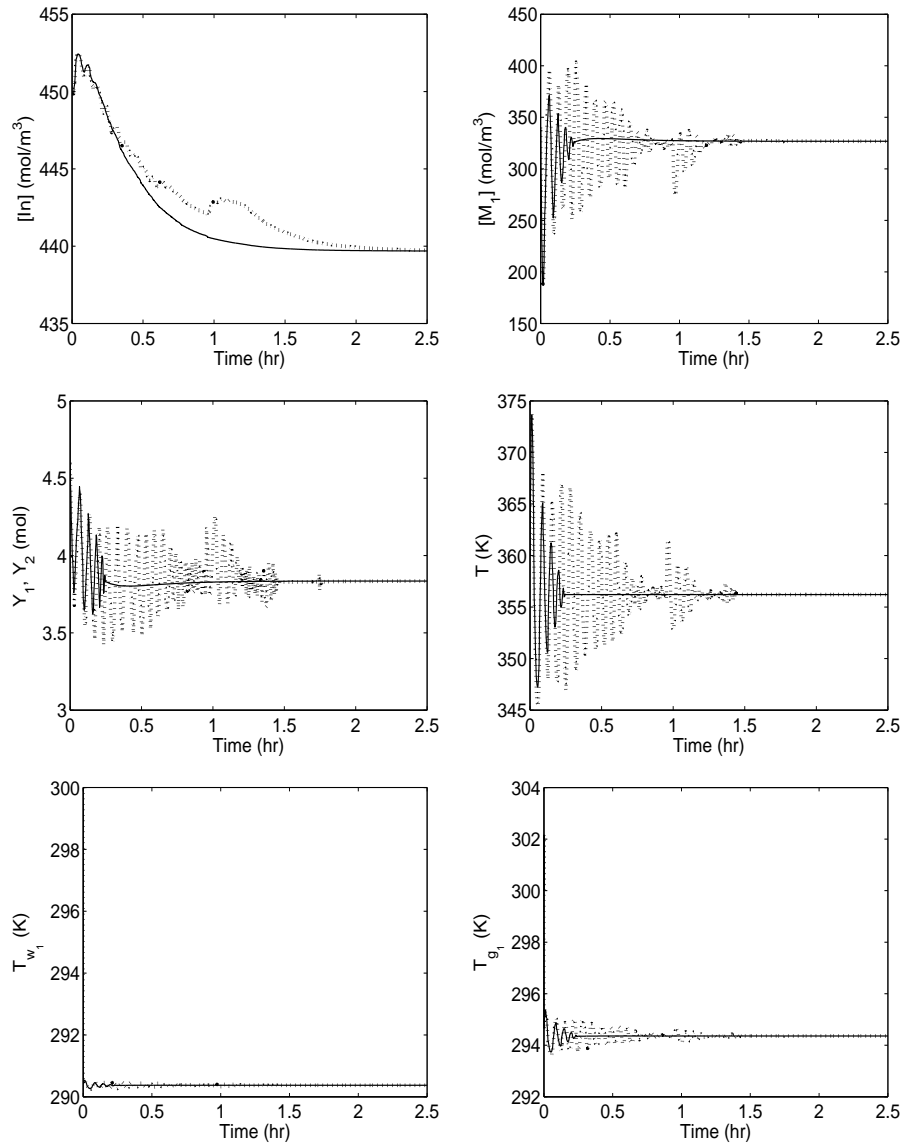


Figure 4.9: Evolution of the closed-loop state profiles under primary control configuration under continuous measurements (solid lines) and sensor data loss rate of 0.75 (dotted lines).

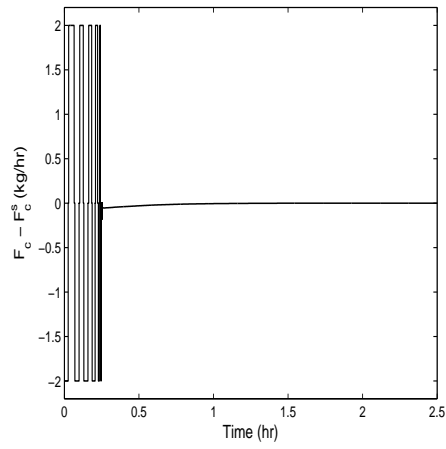


Figure 4.10: Evolution of the manipulated input profiles under primary control configuration under continuous measurements.

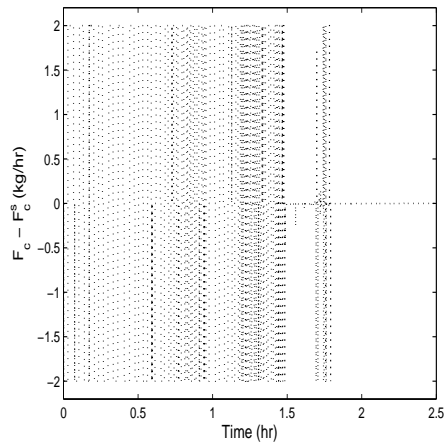


Figure 4.11: Evolution of the manipulated input profiles under primary control configuration with sensor data loss rate of 0.75.

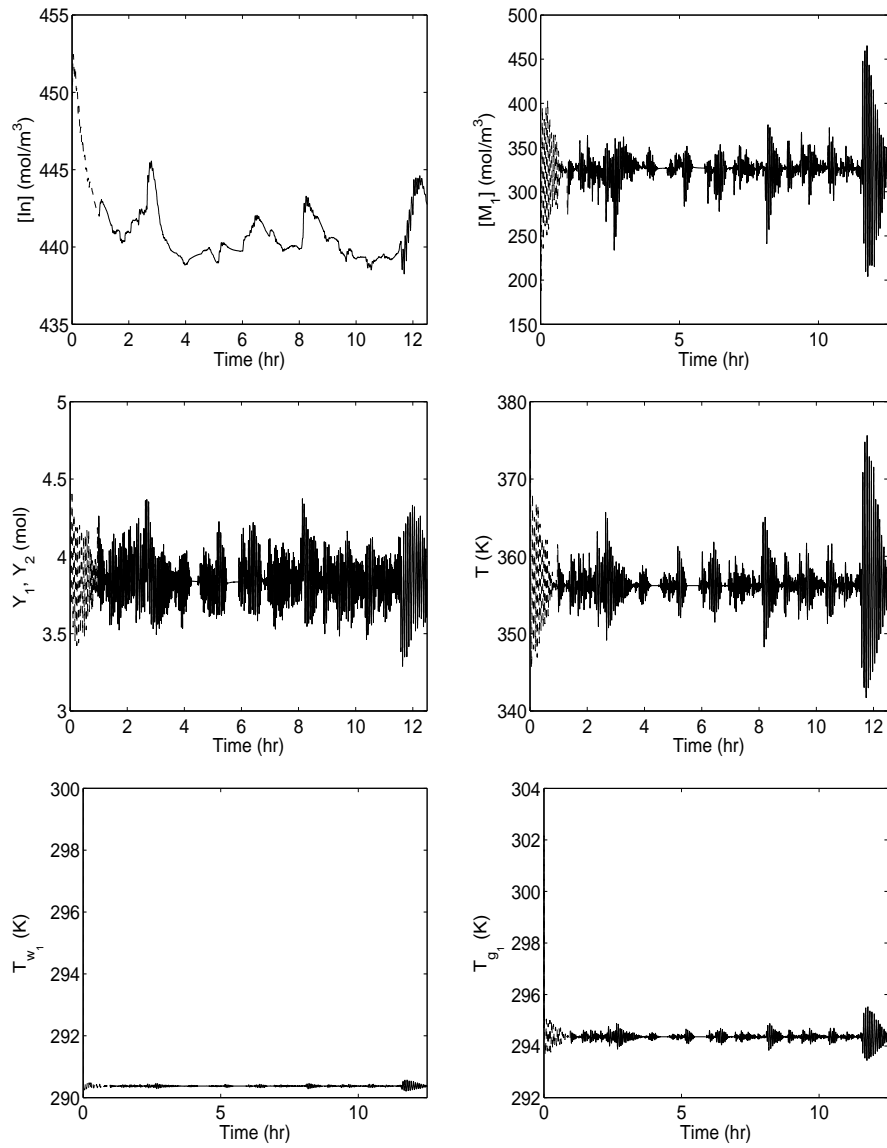


Figure 4.12: Evolution of the closed-loop state profiles under the primary configuration with the data loss rate increasing from 0.75 to 0.80 at 0.97 hours.

is implemented, and the supervisor activates the fall-back configuration, the state trajectory converges to the desired steady-state (see Fig.4.13). The corresponding manipulated input profiles are shown in Fig.4.14.

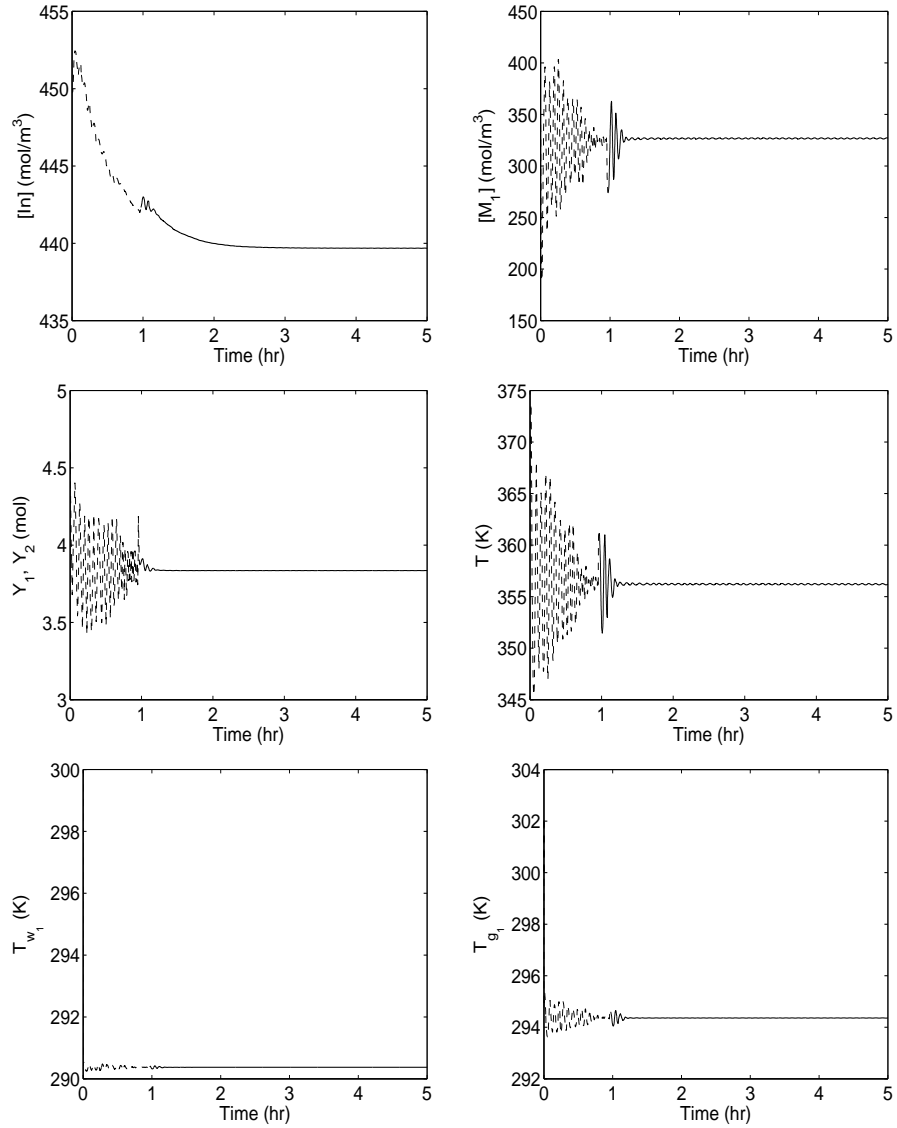


Figure 4.13: Evolution of the closed-loop state profiles under the reconfiguration law of Eq.4.12 with the data loss rate increasing from 0.75 to 0.80 at 0.97 hours.



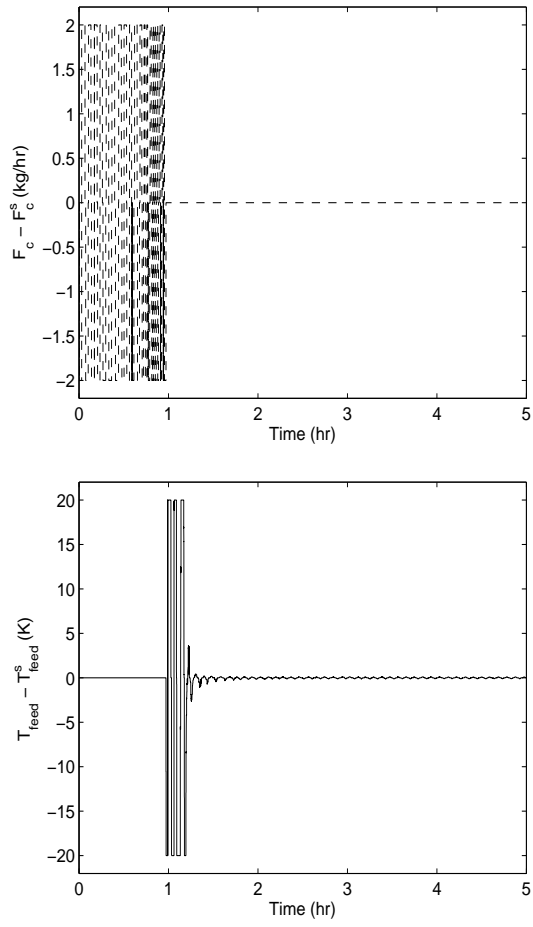


Figure 4.14: Evolution of the closed-loop input profiles under the reconfiguration law of Eq.4.12 with the data loss rate increasing from 0.75 to 0.80 at 0.97 hours.

## 4.6 Conclusions

In this work we considered the problem of designing a fault-tolerant controller for non-linear process systems subject to constraints and sensor data losses. Having identified candidate control configurations for a given system, we first explicitly characterized the stability properties that is, the set of initial conditions starting from where closed-loop stabilization under continuous availability of measurements is guaranteed as well as derived a bound on the maximum allowable data loss rate which preserves closed-loop stability. This characterization was utilized in designing a reconfiguration logic that was shown to achieve practical stability in the presence of sensor data losses. The application of the proposed method was illustrated using a chemical process example and demonstrated on a polyethylene reactor.

## Chapter 5

# Fault-tolerant control of a reverse osmosis desalination process

### 5.1 Introduction

System automation and reliability are crucial components of any modern reverse osmosis (RO) plant. The operational priorities are personnel and product water safety, while also meeting environmental and economic demands. Automated RO plants, however, can be vulnerable to faults in several process components that can effect plant operation. Examples of faults can include valve failure, membrane fouling or scaling, sensor data loss, and pump or variable frequency drive failure. Because RO plants run at high pressures, these failures may cause immediate safety risks to plant personnel. These failures can also lead to a decline in the product water quality, rendering it unsafe for public consumption. These safety issues provide strong motivation for the development of fault-detection and isolation (FDI) and fault-tolerant control (FTC) structures that can quickly identify failed actuators and make effective decisions to maintain safe plant operation.

Several contributions have been made in the literature to process control of RO systems. The first paper which proposed an effective closed-loop control strategy for RO utilized multiple SISO control-loops [2]. Step tests were used to perform system identification, resulting in a model that is a linear approximation around the operating point. The control algorithm of MPC was applied to the resulting linear model in [83] and [1]. Experimental system identification and MPC applications can also be found in [5] and [10]. [55] and [44] implemented minimal feedback control on RO desalination systems, powered by renewable energy sources, in the form of digital on/off switching. Some hybrid systems modeling and control work has been published, such as in [34]. The goal of this chapter is to extend the research on RO control systems to include model-based FDIFTC structures.

Fault-tolerant control structures are based on an underlying assumption that there are more control configurations available than required for the given process [87] and [97]. The use of the minimum number of control inputs is desirable to minimize unnecessary control action. Fault-tolerant control, in this case, can be achieved through reconfiguration of the control-loops. To implement fault-tolerant control structures on an RO system, first it is necessary to detect and isolate failure events. The results from [69] can be directly applied in order to implement FDI on an RO system. Other FTC results relevant to this project can be found in [67] and [35].

This chapter focuses on FTC of an RO process. First, a detailed mathematical model that adequately describes the process evolution is derived. A family of candidate control configurations are identified, and Lyapunov-based feedback control laws are constructed for each configuration such that closed-loop stability is guaranteed within an associated constrained stability region. Subsequently, an FDI filter that observes the deviation of the process states from the expected closed-loop behavior

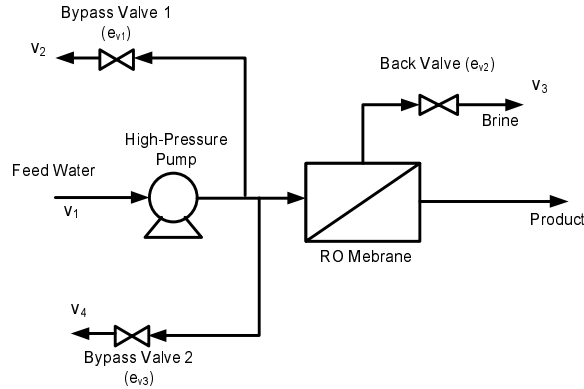


Figure 5.1: Single membrane unit reverse osmosis desalination process.

is developed to detect and isolate actuator failures. A supervisory switching logic is then derived, on the basis of stability regions and FDI filter information, to orchestrate switching between the available control configurations in a way that guarantees closed-loop stability in the event of actuator faults. The effectiveness of the proposed FDIFTC structure is demonstrated through simulation.

## 5.2 Process description and modeling

Fig.5.1 shows a schematic of an elementary RO desalination process. This is a single-unit RO system with no pre-treatment or post-treatment units. Feed brackish or seawater enter the system through the high pressure pump. This high-pressure water then flows across an RO membrane, and low salinity product water permeates. Concentrated brine then continues through a throttling valve and is discharged at atmospheric pressure. The RO plant consists of a high pressure pump, the three automated valves, membrane unit, and required plumbing and tanks. The valve settings can be manipulated in real time based on measurement information which includes the flow velocities.

The first principles model of this system is based on a macroscopic kinetic energy

balance. This model assumes an incompressible fluid and constant internal volume and mass. Skin friction through piping and the membrane system are negligible relative to hydraulic losses in the throttling valves and across the membrane. Three ordinary differential equations that can describe such a system are derived and they have the following form:

$$\begin{aligned}
\frac{dv_2}{dt} &= \frac{1}{\rho V} \left( \frac{W_p}{v_1(v_2, v_3, v_4)} - \frac{1}{2} e_{v1} v_2 \right) \\
\frac{dv_3}{dt} &= \frac{1}{\rho V} \left( \frac{W_p}{v_1(v_2, v_3, v_4)} - \frac{1}{2} e_{v2} v_3 \right) \\
\frac{dv_4}{dt} &= \frac{1}{\rho V} \left( \frac{W_p}{v_1(v_2, v_3, v_4)} - \frac{1}{2} e_{v3} v_4 \right) \\
v_1 &= -\frac{1}{2}b + \frac{1}{2}\sqrt{b^2 + 4c} \\
b &= -(v_2 + v_3 + v_4 - \frac{A_m K_m \Delta \pi}{\rho A_p}) \\
c &= \frac{A_m K_m W_p}{\rho A_p^2}
\end{aligned} \tag{5.1}$$

where  $v_1$ , feed velocity, is a nonlinear function of  $v_2$ ,  $v_3$ , and  $v_4$ .  $v_2$ ,  $v_3$ , and  $v_4$  are the velocities of bypass discharge one, brine discharge, and bypass discharge two respectively.  $\rho$  is the fluid density,  $V$  is the internal volume,  $W_p$  is the power delivered by the pump,  $A_p$  is the pipe cross sectional area.  $e_{v1}$ ,  $e_{v2}$ , and  $e_{v3}$  are the frictional valve constants.  $A_m$  is the membrane area,  $K_m$  is a membrane mass transfer coefficient, and  $\Delta\pi$  is the osmotic pressure. The potential manipulated inputs of the model are the valve constants ( $e_{v1}$ ,  $e_{v2}$ , and  $e_{v3}$ ) which can be manipulated in practice by an automated electric motor that partially opens or closes the valves. The measured outputs are the velocities of the fluid in the bypass lines, and brine velocity ( $v_2$ ,  $v_3$ , and  $v_4$ ). Internal pressure,  $P$  can be related to feed velocity by  $P = \frac{W_p}{v_1 A_p}$ . The product velocity,  $v_5$ , can be related to the system pressure by  $v_5 = \frac{A_m K_m}{\rho A_p} (P - \Delta\pi)$ .

Table 5.1 shows the parameter values used for this example.

The control objective is to stabilize the process at the desired steady-state. There are at least two unique configurations that will give simultaneous independent control of transmembrane pressure and brine flow-rate. Configuration one,  $u_1$ , uses the back valve and the first bypass valve ( $e_{v1}$ ,  $e_{v2}$ ) as manipulated inputs. The valves are subjected to input constraints of the form  $0 < e_{v1} < 200$  and  $130 < e_{v2} < 330$ . Configuration two,  $u_2$ , uses the back valve with the second bypass valve ( $e_{v2}$ ,  $e_{v3}$ ) as manipulated inputs. These valves are subjected to input constraints of the form  $130 < e_{v2} < 330$  and  $200 < e_{v3} < 400$ . The first control configuration,  $u_1$ , will be considered as the primary configuration. However, in the event of a failure the plant supervisor may need to implement the fall-back configuration,  $u_2$ , to maintain closed-loop stability. By observing the evolution of the plant the FDI filters can detect and isolate an actuator fault. If there is a fall-back control configuration available that is able to stabilize the RO plant, then the supervisor will initiate a mode transition to the fall-back configuration. These issues are addressed in detail in the next section.

### 5.3 Fault-detection and isolation and fault-tolerant control

Given the properties of the dynamic model, Eq.5.1, it can be shown that both configurations,  $u_1$  and  $u_2$ , satisfy the requirements of achieving fault-detection and isolation of actuator faults (see [69] for details). This section discusses the four steps to implement FDIFTC on the RO process. The first step is to synthesize stabilizing feed-back controllers for each configuration. The second step is to explicitly characterize the constrained stability region associated with each configuration. The third step is to design FDI filters for each manipulated input. The final step is to design the switching law that orchestrates the reconfiguration of the control system in a way that guaran-

Table 5.1: Process parameters and steady-state values

$\rho$	=	1000	$kg/m^3$
$V$	=	10	$L$
$W_p$	=	104.4	$Watts$
$A_p$	=	0.25	$in^2$
$A_m$	=	5	$m^2$
$K_m$	=	$9.218 \times 10^{-9}$	$s/m$
$\Delta\pi$	=	200	$psi$
$e_{v1}^{s1}$	=	100	
$e_{v2}^{s1}$	=	230	
$e_{v3}^{s1}$	=	$10^{-8}$	
$v_2^{s1}$	=	1.0547	$m/s$
$v_3^{s1}$	=	0.4625	$m/s$
$v_4^{s1}$	=	$1.07 \times 10^{-6}$	$m/s$
$P^{s1}$	=	243.7	$psi$
$e_{v1}^{s2}$	=	150	
$e_{v2}^{s2}$	=	230	
$e_{v3}^{s2}$	=	300	
$v_2^{s2}$	=	0.7092	$m/s$
$v_3^{s2}$	=	0.4625	$m/s$
$v_4^{s2}$	=	0.3546	$m/s$
$P^{s2}$	=	243.7	$psi$



tees closed-loop stability in the event of faults in the active control configuration.

To present results in a convenient form, the model of Eq.5.1 is written in deviation variable form around the desired steady state. This is defined as  $x = [x_1 \ x_2 \ x_3]^T$  where  $x_1 = v_2 - v_{2s}$ ,  $x_2 = v_2 - v_{2s}$ , and  $x_3 = v_4 - v_{4s}$ . The plant can then be described by the following nonlinear continuous-time system:

$$\begin{aligned} \dot{x}(t) &= f_{k(t)}(x(t)) + g_{k(t)}(x(t))u_{k(t)} \\ |u_{k(t),i}| &\leq u_{k,i}^{max} \\ k(t) \in K &= \{1, 2\} \end{aligned} \tag{5.2}$$

where  $x(t) \in \mathfrak{R}^3$  denotes the vector of process state variables and  $u_{k(t)}$  is a vector of inputs where  $u_{k,i}(t) \in [-u_{k,i}^{max}, u_{k,i}^{max}] \subset \mathfrak{R}^3$  denotes the  $i^{th}$  constrained manipulated input associated with the  $k^{th}$  control configuration.  $k(t)$ , which takes values in the finite set  $K$ , represents a discrete state that indexes the vector fields  $f_k(\cdot)$ ,  $g_k(\cdot)$  and the manipulated inputs  $u_k(\cdot)$ . The explicit form of the vector fields can be obtained by comparing Eqs.5.1 and 5.2 and is omitted for brevity. For each value that  $k$  assumes in  $K$ , the process is controlled via a different set of manipulated inputs which define a given control configuration. Switching between the two available configurations is handled by the high-level supervisor. The control objective is to stabilize the process in the presence of actuator constraints and possible faults. The state feedback problem where measurements of all process states are available for all times is considered to simplify presentation of the results.

### 5.3.1 Constrained feedback controller synthesis

In this step we synthesize for each control configuration a feedback controller that enforces asymptotic closed-loop stability in the presence of actuator constraints. To accomplish this task first a quadratic Lyapunov function of the form  $V_k = x^T P_k x$  is defined, where  $P_k$  is a positive-definite symmetric matrix that satisfies the Riccati in-

equality. This Lyapunov function is used to synthesize a bounded nonlinear feedback control law for each control-loop (see [54] and [25]) of the form:

$$u_k = -r(x, u_k^{max})L_{\bar{g}_k}V_k \quad (5.3)$$

where

$$r = \frac{L_{\bar{f}_k}^*V_k + \sqrt{(L_{\bar{f}_k}^*V_k)^2 + (u_k^{max}|L_{\bar{g}_k}V_k|)^4}}{(|L_{\bar{g}_k}V_k|)^2(1 + \sqrt{1 + (u_k^{max}|L_{\bar{g}_k}V_k|)^2})} \quad (5.4)$$

and  $L_{\bar{f}_k}^*V_k = L_{\bar{f}_k}V_k + \alpha V_k$ ,  $\alpha > 0$ . The scalar function  $r(\cdot)$  in Eqs.5.3 and 5.4 can be considered as a nonlinear controller gain. It can be shown that each control configuration asymptotically stabilizes the states in each mode. This controller gain, which depends on both the size of actuator constraints,  $u_k^{max}$ , and the particular configuration used is shaped in a way that guarantees constraint satisfaction and asymptotic stability within a well-characterized region in the state space. The characterization of this region is discussed in the next step.

### 5.3.2 Characterization of stability regions

Actuator constraints place fundamental limitations on the initial conditions from which the closed-loop system is asymptotically stable. It is important for the control system designer to explicitly characterize these limitations by identifying, for each control configuration, the set of initial conditions for which the constrained closed-loop system is asymptotically stable. This is necessary for the design of an appropriate switching policy that ensures the fault-tolerance of the closed-loop system. The feedback controller of Eq.5.3 that is synthesized for each configuration provides such a characterization. Specifically, using a Lyapunov argument, one can show that the set

$$\Theta(u_k^{max}) = \{x \in \mathbb{R}^3 : L_{\bar{f}_k}^* V_k \leq u_k^{max} |L_{\bar{g}_k} V_k|\} \quad (5.5)$$

describes a region in the state-space where the control action satisfies the constraints and the time-derivative of the corresponding Lyapunov function is negative-definite along the trajectories of the closed-loop system (see [13]). Note that the size of the set depends on the magnitude of the constraints. The set becomes smaller as the constraints become tighter (smaller  $u_{k,i}^{max}$ ). For a given control configuration, the above inequality can be used to estimate the associated stability region. This can be done by constructing the largest invariant subset of  $\Theta$ , which is denoted by  $\Omega(u_k^{max})$ . Initial conditions within the set  $\Omega(u_k^{max})$  ensure that the closed-loop trajectory stays within the region defined by  $\Theta(u_k^{max})$ , and thereby  $V_k$  continues to decay monotonically, for all times that the  $k^{th}$  control configuration is active (see [24] for further discussion on this issue). An estimate of  $\Omega(u_k^{max})$  is obtained by defining a composite Lyapunov function of the form  $V_{C_k} = x^T P_C x$ , where  $P_C$  is a positive definite matrix, and choosing a level set of  $V_{C_k}$ ,  $\Omega_{C_k}$ , for which  $\dot{V}_{C_k} < 0$  for all  $x$  in  $\Omega_{C_k}$ . The value  $c_k^{max}$  represents a level set on  $V_{C_k}$  where  $\dot{V}_{C_k} < 0$ .

### 5.3.3 Fault-detection and isolation filter design

The third step in implementing FDIFTC is that of designing appropriate fault-detection filters. The filters should detect and isolate the occurrence of a fault in an actuator by observing the behavior of the closed-loop process. The FDI filter design for the primary control configuration takes the form:

$$\begin{aligned}
\frac{d\tilde{v}_2}{dt} &= \frac{1}{\rho V} \left( \frac{W_p}{v_1(\tilde{v}_2, v_3, v_4)} - \frac{1}{2} e_{v1}(\tilde{v}_2, v_3, v_4) \tilde{v}_2 \right) \\
\frac{d\tilde{v}_3}{dt} &= \frac{1}{\rho V} \left( \frac{W_p}{v_1(v_2, \tilde{v}_3, v_4)} - \frac{1}{2} e_{v2}(v_2, \tilde{v}_3, v_4) \tilde{v}_3 \right) \\
r_{1,1} &= |v_2 - \tilde{v}_2| \\
r_{1,2} &= |v_3 - \tilde{v}_3|
\end{aligned} \tag{5.6}$$

Where  $\tilde{v}_2$  and  $\tilde{v}_3$  are the filter states for valve one and two respectively.  $r_{k,i}$  is the residual associated with the  $i^{th}$  input of the  $k^{th}$  configuration. The filter states are initialized at the same value as the process states ( $\tilde{x}(0) = x(0)$ ) and essentially predict the evolution of the process in the absence of actuator faults (This assumption can be relaxed, see [69]). The residual associated with each manipulated input captures the difference between the predicted evolution of the states in the absence of a fault on that actuator and the evolution of the measured process state. If a given residual becomes non-zero, a fault is declared on the associated input. For a detailed analysis of the FDI properties of the filter, see [69].

### 5.3.4 Fault-tolerant supervisory switching logic

The final step is to design a switching logic that the plant supervisor will use to decide what fall-back control configuration to implement given an actuator failure. The supervisor should only implement those configurations that will guarantee closed-loop stability and do not utilize a failed actuator. This requires that the supervisor only activates fall-back control configurations for which the state is within the associated stability region at the time of fault-detection. Let the initial actuator configuration be  $k(0) = 1$ ,  $T_{fault}$  be the time of an actuator failure, and  $T_{detect}$  be the earliest time at which the value of  $r_{1,i}(t) > \delta_{r_{1,i}} > 0$  (for the  $i^{th}$  input where  $\delta_{r_{1,i}}$  is the  $i^{th}$  detection threshold). The switching rule given by

$$k(t \geq T_{detect}) = 2 \text{ if } x(T_{detect}) \in \Omega_{C_2}(u_2^{max}) \quad (5.7)$$

guarantees asymptotic closed-loop stability if  $u_2$  does not include any faulty actuators. The switching law requires monitoring of FDI filters and process state location with respect to fall-back stability regions.

## 5.4 Simulation results

A simulation has been performed to demonstrate the implementation of the proposed FDIFTC strategy on the RO plant of Fig.5.1. The states in the mathematical model given in Eq.5.1 may not be the system parameters of interest for the operator because bypass flows ( $v_2$  and  $v_4$ ) do not interact with the membrane unit. Pressure and brine flow,  $P$  and  $v_3$ , are useful parameters to regulate because they directly effect the membrane unit. Hence, two steady-states have been considered, each one of them has the same system pressure and brine flow rate ( $v_3$ ), but different bypass flows ( $v_2$  and  $v_4$ ). The first steady-state corresponds to bypass valve two being closed. The parameters and steady-state values can be seen in Table 5.1. Under these operating conditions the open-loop system behaves in a stable fashion at each steady-state.

First, nonlinear feedback control under the primary configuration,  $u_1$ , was considered. The bounded nonlinear controller was synthesized using Eqs.5.3 and 5.4, with  $\alpha = 0.1$ . The stability region for the primary configuration was estimated using the Lyapunov function,  $V_1 = x^T P_1 x$ , yielding a  $c_1^{max} = 1$  (note: this value of  $c_1^{max}$  represents a sufficiently large region of the state space for this simulation, in general much higher values can be considered). Fig.5.2 shows the evolution of the closed-loop state profiles starting from the initial condition  $v_2 = v_3 = 0.1 \frac{m}{s}$  and  $v_4 = 0.001 \frac{m}{s}$  for which  $V_1(x(0)) = 0.0263$ . Evolution of the system pressure is shown in Fig.5.3. Since

the initial state was within the stability region of the primary control configuration,  $V_1(x(0)) = 0.0263 \leq c_1^{max} = 1$ , the primary control configuration was able to stabilize the system at the desired steady-state.

Next, a fault in the primary configuration (in  $e_{v1}$  specifically) at a time  $T_{fault} = 10$  s was considered. In this case the fall-back configuration,  $u_2$ , was available with valve three,  $e_{v3}$ , as one of the manipulated inputs. The quadratic Lyapunov function  $V_2 = x^T P_2 x$  and  $\alpha = 0.1$  was used to design the controller. The stability region was also estimated using  $V_2$  yielding a  $c_2^{max} = 1$ .

To demonstrate the advantage of operating under the FDIFTC structure consider the case where no control system reconfiguration takes place after  $T_{fault}$ . The system is initialized at  $v_2 = v_3 = 0.1 \frac{m}{s}$  and  $v_4 = 0.001 \frac{m}{s}$ , and the primary control configuration operates normally until the time  $T_{fault} = 10$  s. At this time valve one stops operating and is partially closed,  $e_{v1} = 150$ . As shown by the solid lines in Figs.5.2 and 5.3 the states move away from the desired values, and settle at a new, undesired, steady-state.

However, by implementing the FDIFTC structure the fault can be mitigated. The residual value associated with valve one,  $r_{1,1}$ , becomes non-zero and reaches the detection threshold,  $\delta_{r_{1,1}} = 0.01$ , at  $T_{detect} = 10.004$  s when the fault is declared. The residual value associated with valve two,  $r_{1,2}$  remains at zero, indicating that the fault is effecting only valve one. At time  $T_{detect}$  the value of the fall-back Lyapunov function is checked against the fall-back stability region to see if switching would guarantee stability. The value of  $V_2(x(T_{detect})) = 0.0119 < c_2^{max} = 1$ , so reconfiguration to the fall-back controller,  $k = 2$ , does guarantee closed-loop stability. The evolution of the system states and pressure under the proposed FDIFTC structure can be seen in Figs.5.2 and 5.3 (solid lines). This automated reconfiguration allowed the closed-loop system to maintain pressure and brine flow at the desired values.

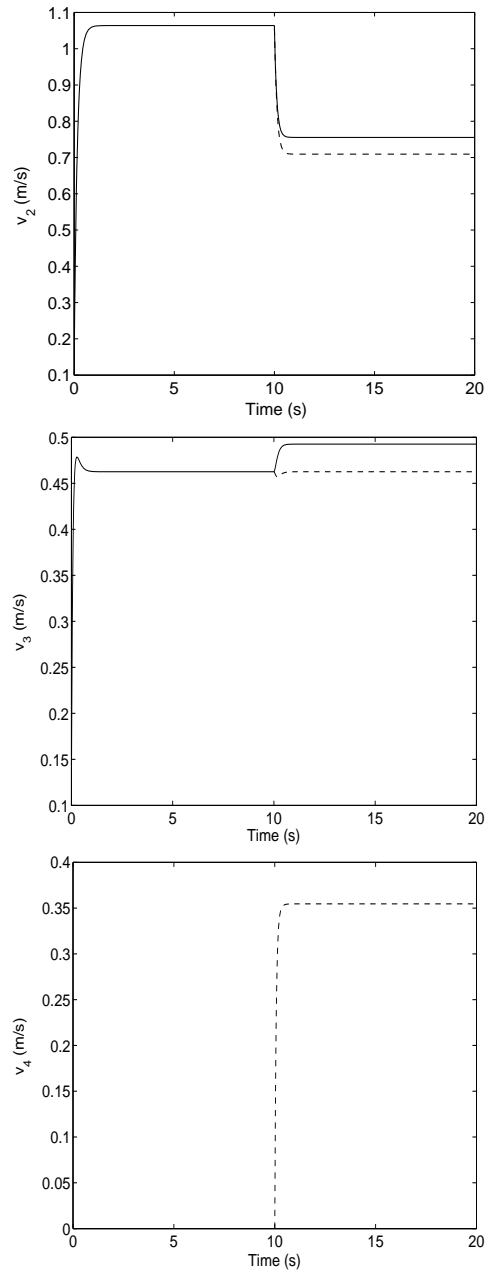


Figure 5.2: Evolution of the closed-loop state profiles under fault-tolerant control (dashed line) and without fault tolerant-control (solid line). FTC recovers the desired brine flow,  $v_3$ .

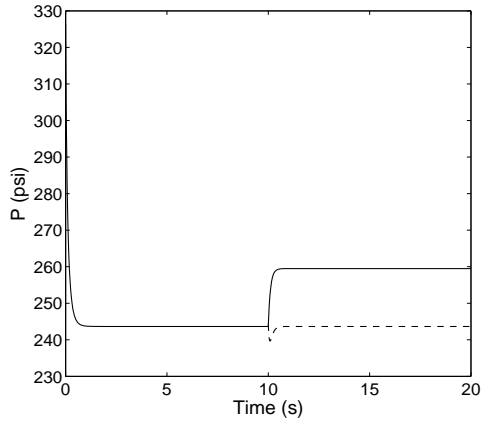


Figure 5.3: Evolution of the closed-loop pressure profile under fault tolerant control (dashed line) and without fault tolerant control (solid line). FTC recovers the desired operating pressure.

## 5.5 Conclusions

The focus of this work was to apply recently-developed FDIFTC structures to an RO desalination process model. First, a mathematical model that describes the process evolution was developed. A family of candidate control configurations was then identified, and Lyapunov-based feedback control laws were constructed for each configuration such that closed-loop stability was guaranteed within an associated constrained stability region. An FDI filter that observes the deviation of the process states from the expected closed-loop behavior was developed to detect and isolate actuator failures. A supervisory switching logic was then derived, on the basis of stability regions and FDI filter information, to orchestrate switching between the available control configurations in a way that guarantees closed-loop stability in the event of actuator faults. These ideas were then demonstrated in the context of an RO system simulation. The proposed FDIFTC methodology was able to maintain closed-loop operation at the desired steady-state in the presence of actuator failures.



## Chapter 6

# Control and monitoring of a high recovery reverse osmosis desalination process

### 6.1 Introduction

The goal of this chapter is to develop model-based nonlinear feed-forward/feedback control structures for high recovery RO desalination systems while accounting for such practical issues as sampled and noisy measurements, large time-varying disturbances, and actuator faults. In order to accomplish this goal a detailed mathematical model of a high-recovery RO plant is first developed. This model adequately describes the evolution of process states in time, and it also accounts for the spatial variation of total dissolved solids (TDS) and flow-rate inside the membrane units. Most RO models simple enough for control purposes, such as those found in [39], consider a well mixed model with a single value for concentration on the retentate side of the membrane. However, under high recovery operation the gradients along the length of the membrane unit can be quite significant. As fluid flows axially along the module

the bulk concentration increases, the flow rate decreases, and the local permeate flux decreases [20]. The model developed in the present work includes appropriate differential equations in space that account for these gradients. A Lyapunov-based nonlinear controller [13, 24] is then applied to this high recovery RO model. One of the main objectives of a controller in high recovery RO is to reject disturbances caused by feed water variation. Feed disturbances could cause undesired fluctuations in the product flow rate or the internal pressure. To accomplish disturbance rejection, the control law includes both feedback and feed-forward components (i.e., measurement of feed concentration fluctuations). The feed water stream concentration can easily be measured in practice, so the first set of simulations presented in this work explore the ability of the proposed control method to reject such disturbances. Another objective is to detect and isolate actuator faults as soon as possible. A second set of illustrative examples demonstrate how fault-detection and isolation (FDI) and fault tolerant control (FTC) can be applied to this system, and how appropriate action can be taken to maintain desired system operation when a fault occurs in the control system.

## 6.2 Process description and modeling

Fig. 6.1 shows a schematic of an elementary RO desalination process. This is a single-unit RO system with no pre-treatment or post-treatment units. Feed brackish water or seawater enters the system through a high pressure pump. This high pressure water then flows across an RO membrane, and low salinity product water permeates through the membrane. Concentrated brine then exits the membrane module and passes through a throttling valve to be discharged at atmospheric pressure. The RO plant consists of a high pressure pump, two automated valves, a spiral wound

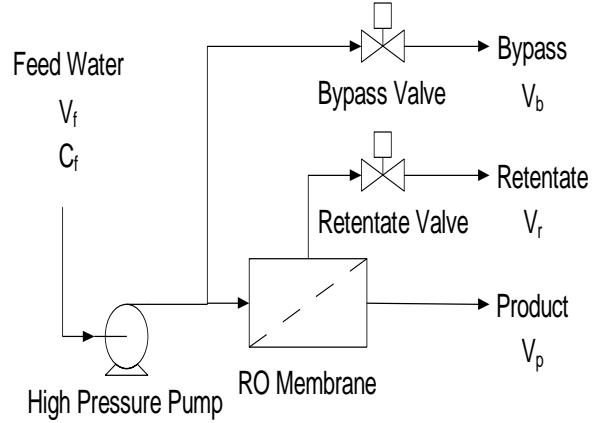


Figure 6.1: Single membrane unit high recovery reverse osmosis desalination process. The two actuated valves, retentate valve and bypass valve, act as manipulated inputs.

membrane unit, required plumbing, and tanks. The valve settings can be manipulated in real time based on measurement information which includes the flow velocities and feed concentration.

The first principles model of this system is based on a macroscopic kinetic energy balance, a local mass balance, and a microscopic mass shell balance. This model assumes an incompressible fluid and constant internal volume and mass. It is assumed that the water in the module travels in a plug flow with no back-mixing or axial diffusion. It is also assumed that the osmotic pressure can be related to the TDS at the membrane surface [56]. Skin friction through the piping and the membrane module are considered negligible relative to the hydraulic losses in the throttling valves and across the membrane.

The energy balance consists of two nonlinear ordinary differential equations (ODEs) in time where the velocities of the bypass and retentate stream are the states. Each ODE is derived from an energy balance around an actuated control valve [9]. Specifically, the two ODEs that can describe the process depicted in Fig. 6.1 take the following form:

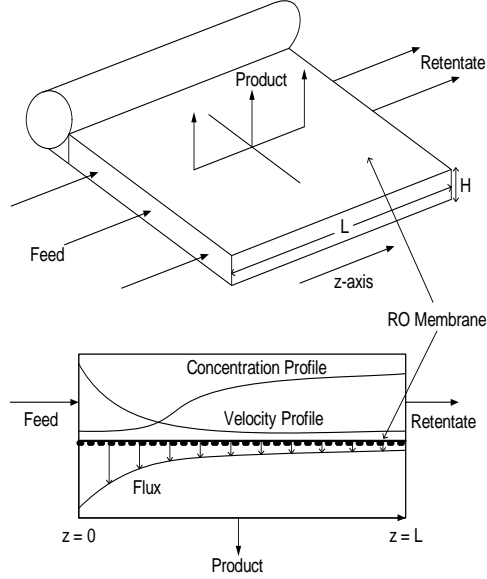


Figure 6.2: An expanded view of a spiral wound membrane module and typical concentration and velocity profiles inside the module.

$$\begin{aligned}\frac{dv_b}{dt} &= \frac{A_p}{\rho V} (P - \frac{1}{2} v_b^2 e_{v1}) \\ \frac{dv_r}{dt} &= \frac{A_p}{\rho V} (P - \frac{1}{2} v_r^2 e_{v2})\end{aligned}\tag{6.1}$$

where  $v_b$  is the bypass velocity,  $v_r$  is the retentate velocity,  $A_p$  is the pipe cross sectional area,  $V$  is the total internal volume,  $\rho$  is the fluid density, and  $P$  is the internal pressure.  $e_{v1}$  and  $e_{v2}$  are friction loss factors for the actuated valves and act as manipulated inputs. The two ODEs of (6.1) are not explicitly coupled, however, coupling does occur through the pressure term,  $P$ . The pressure,  $P$ , in this work is an algebraic variable which is an implicitly nonlinear function of  $v_b$  and  $v_r$  for which there exists no differential equation in time.  $P$  is assumed to be space independent throughout the high pressure region by neglecting skin friction. Specifically,  $P$  at each time is obtained via solving a local mass balance and a microscopic mass shell balance

along the length of the membrane module in space. The local mass balance around the bypass line and feed line junction allows the calculation of the feed velocity to the membrane module,  $v_{mf}$ , given the bypass and retentate velocities from (6.1):

$$v_f = v_b + v_{mf} \quad (6.2)$$

where  $v_f$  is the constant velocity of the feed stream.

It is critical in a high recovery system, where the concentration and velocity in the module change significantly along the axis of flow, to accurately describe the concentration and velocity profiles along the membrane module. In order to model these profiles a shell balance is performed across the length of the membrane module to generate a two state ODE system in space. To clarify how this model is developed, an expanded view of an unwound spiral-wound membrane module and a drawing depicting typical concentration and velocity profiles in a module can be seen in Fig. 6.2. The internal compartment of the membrane module is simplified to a rectangular space. A steady-state shell balance is performed on this space assuming radially well mixed plug flow. This steady-state approximation is made under the assumption that disturbances on the system will act on a much slower time scale than the residence time in the membrane unit. The shell balances are based on the conservation of TDS mass and water mass inside the membrane module. The differential volume for the shell balance has the dimensions  $W$  by  $H$  by  $\delta z$ , where  $\delta z$  is an infinitesimal length in the  $z$  direction.  $W$  is the membrane width ( $W = A_m/L$ ,  $A_m$  is the membrane area) and  $H$  is the channel height. The derivation assumes that dissolved solids are completely rejected, and that only water permeates the membrane at a flux approximated by  $J_w = K_m(P - K_{\Delta\pi}C_z)$ , where  $J_w$  is the permeate flux,  $K_m$  is the overall

mass transfer coefficient,  $K_{\Delta\pi}$  is a constant that relates  $C_z$  to osmotic pressure, and  $C_z$  is the concentration along the  $z$ -axis in the membrane. The result of the shell balance is the following two coupled ODEs in space and three boundary conditions (owing to the fact that  $P$  is an algebraic variable):

$$\begin{aligned}
\frac{dC_z}{dz} &= \frac{C_z K_m (P - K_{\Delta\pi} C_z)}{v_z \rho H} \\
\frac{dv_z}{dz} &= -\frac{K_m (P - K_{\Delta\pi} C_z)}{\rho H} \\
C_z(z=0) &= C_f \\
v_z(z=0) &= \alpha v_{mf} \\
v_z(z=L) &= \alpha v_r
\end{aligned} \tag{6.3}$$

where  $z$  is the direction of flow through the membrane,  $v_z$  is the velocity of flow in the membrane along the  $z$ -axis, and  $H$  is the height of the membrane channel. The boundary conditions arise when (6.3) is coupled with (6.1) and (6.2). Equation (6.3) is solved at each time step as we integrate (6.1) in time. The solution to the ODEs of (6.3) is complicated by the fact they must satisfy three boundary conditions, two at the inlet, and one at the outlet owing to the fact that  $P$  is an unknown algebraic variable. The feed concentration,  $C_f$ , represents a boundary condition at  $z = 0$  (at the membrane inlet) provided as a time varying parameter. The feed velocity to the module,  $v_{mf}$ , provides the velocity boundary condition at  $z = 0$ . Retentate velocity,  $v_r$ , provided from (6.1) is a boundary condition at the membrane outlet,  $x = L$ , where  $L$  is the membrane length in the  $z$  direction. The parameter  $\alpha$  is the ratio of the pipe cross sectional area to the membrane channel cross sectional area. Pressure is the unknown variable in time that must be adjusted in order to find the solution to (6.3). The solution to (6.3) is found at each time by using a type of shooting method [11] where the system pressure is adjusted until all three boundary conditions are satisfied.

This system pressure is then substituted into (6.1) for the next step of integration forward in time.

**Remark 6.1:** The model of (6.1), (6.2), and (6.3) can be expanded in several ways to improve the accuracy at the expense of greater model complexity. The pressure, for example, is taken as constant along the length of the membrane module at a specific time instant. However, in a real system there is a minor pressure loss due to skin friction and the pressure will decrease in the  $z$  direction. The model could be expanded to handle this by deriving an ODE that describes  $\frac{dP}{dz}$  and including it in Eq. 6.3. The same solution algorithm would be used, but a guess value for  $P(z = 0)$  should be used in the place of  $P$  in step 2. Another improvement to the model would be to use transient PDEs to describe the velocity and concentration profiles. This would effectively remove the steady-state approximation in (6.3), and would make the model dynamics more accurate on time scales shorter than the membrane residence time. The model could also be expanded to include the concentration gradient in the  $y$  direction (the gradient from the bulk solution to the membrane surface), thus giving a more accurate osmotic pressure and product flux. The osmotic pressure term could also be expanded algebraically to include the effects of temperature. There are also other minor modeling improvements that could be made, but the goal is to obtain a model that is computationally tractable yet accurate enough to synthesize model-based feedback control laws.

### 6.3 Reverse osmosis process model solution algorithm

A step-by-step discussion of the algorithm used to compute the solution of the open-loop model of (6.1), (6.2), and (6.3) is presented to clarify the method employed in this work. An assumption is made that the profiles of  $C_z$  and  $v_z$  change only with

respect to  $z$  within each integration step in time; this assumption can be satisfied by picking the time step of integration to be sufficiently small. It is also assumed that  $C_f$  changes slowly relative to the residence time in the module which is a reasonable assumption for any real RO process. This allows the independent solution of (6.3) at each time step. A large well-mixed holding tank placed before the feed can act as a low-pass filter to eliminate fast time-scale disturbances.

In order to solve the system of equations presented in (6.1), (6.2), and (6.3) numerically, the following algorithm is applied.

1. Initial conditions for  $v_b$  and  $v_r$  are chosen.
2.  $v_{mf}$  is computed from (6.2).
3. Given  $v_{mf}$ ,  $C_f$ , and a guess value for  $P$ , a solution to (6.3) is computed numerically.
4. The resulting  $v_z(z = L)$  is compared to  $\alpha v_r$ , and  $P$  is adjusted via shooting method until  $v_z(z = L)$  is equal to  $\alpha v_r$ .
5. The value of  $P$  resulting from step 4 is used in (6.1) to integrate numerically one step forward in time.
6. The results of step 5 provide updated values of the states,  $v_b$  and  $v_r$ , and the algorithm returns to step 1 using these values as new initial conditions. This process is repeated until the desired integration time is reached.

The open-loop simulation results can be seen as the solid lines in Figs. 6.4 through 6.7 for the parameters in Table 6.1. The simulation is run at high recovery (just over 90% recovery) for a time of 24 *hr* with a time varying disturbance on  $C_f$  as defined in Fig. 6.3. This disturbance was generated from sinusoidal functions and



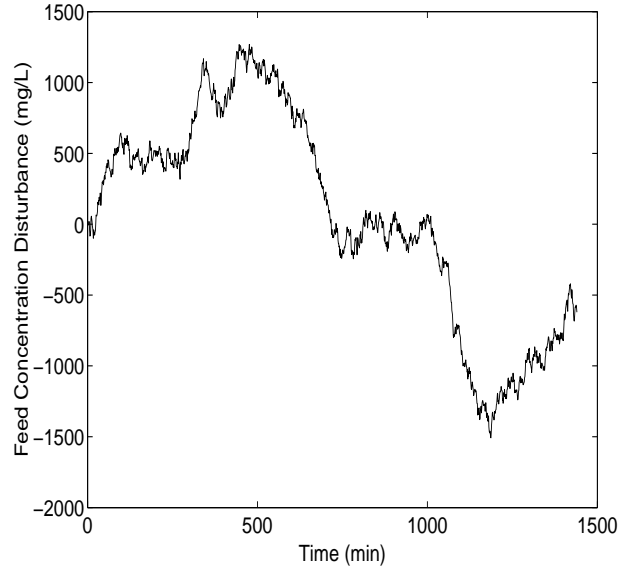


Figure 6.3: Disturbance on feed concentration versus time, this large time-varying disturbance on the RO system is added to the nominal  $C_f$  value.

autocorrelated noise to give an approximation of disturbances encountered in practice. It can be seen that  $v_b$  and  $v_p$  oscillate due to the disturbance, but the oscillations are not large relative to the steady-state values for these states. However, Fig. 6.6 shows wide swings in the internal pressure for the open-loop case. This type of behavior could lead to safety issues if the pressure exceeds the safety rating of hoses, fittings, or pressure vessels. This motivates the use of feedback control to reduce the effects of feed disturbances on the process.

## 6.4 Feedback controller synthesis

The potential manipulated inputs to the system are the friction loss factors for the valves ( $e_{v1}$  and  $e_{v2}$ ). Valves can be manipulated in practice by an automated electric motor that partially opens or closes the valves. The measured outputs are the bypass velocity ( $v_b$ ), retentate velocity ( $v_r$ ), and internal pressure ( $P$ ). The super script

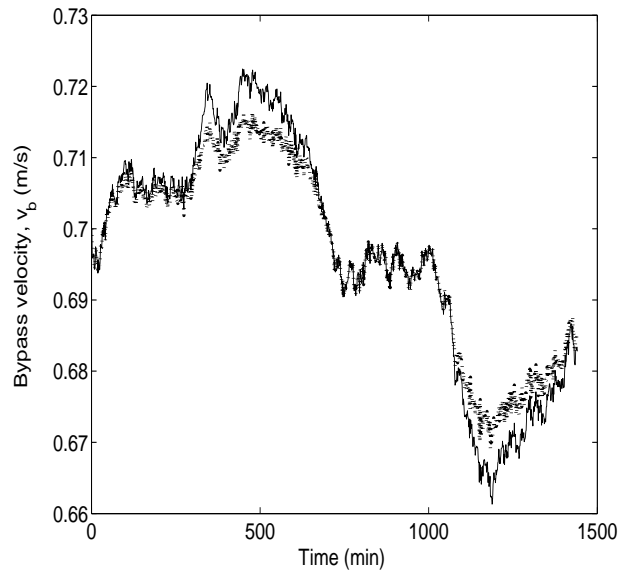


Figure 6.4: Bypass velocity,  $v_b$ , profiles versus time; Open-loop (solid line), closed-loop feedback control without disturbance measurements (dotted line).

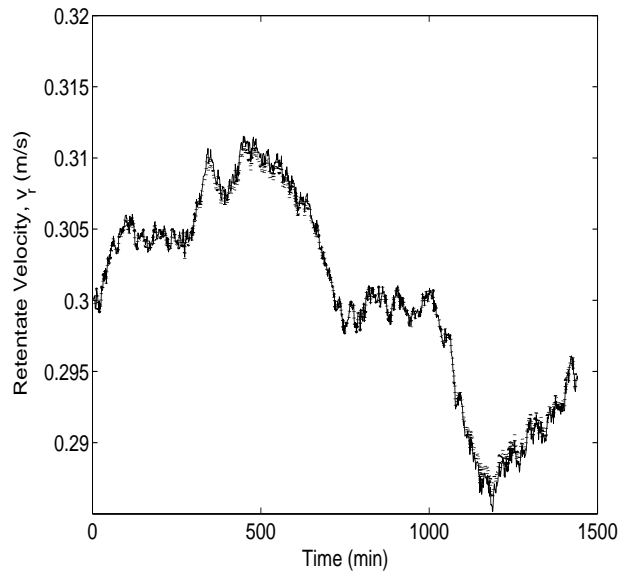


Figure 6.5: Retentate velocity,  $v_r$ , profiles versus time; Open-loop (solid line), closed-loop feedback control without disturbance measurements (dotted line). The dotted line nearly overlaps the solid line.

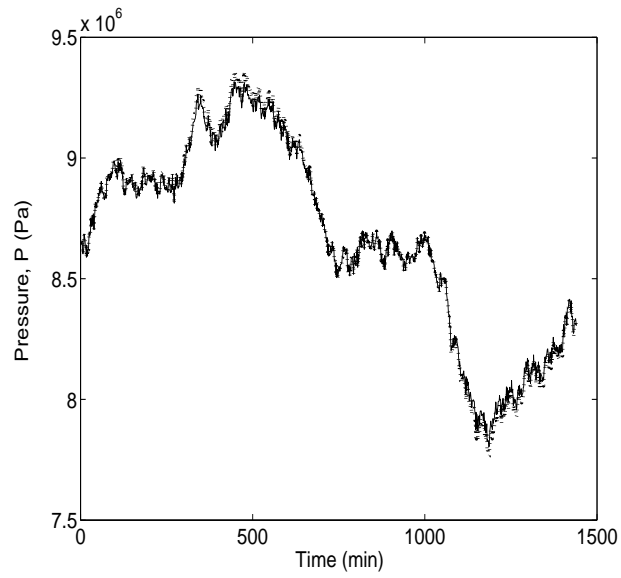


Figure 6.6: Internal pressure,  $P$ , profiles versus time; Open-loop (solid line), closed-loop feedback control without disturbance measurements (dotted line). The dotted line nearly overlaps the solid line.

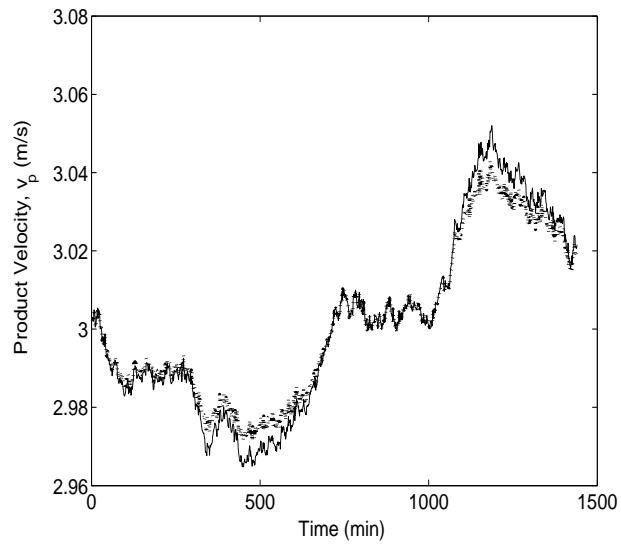


Figure 6.7: Product velocity,  $v_p$ , profiles versus time; Open-loop (solid line), closed-loop feedback control without disturbance measurements (dotted line).

Table 6.1: Process parameters and steady-state values

$\rho$	=	1000	$kg/m^3$
$V$	=	0.1	$m^3$
$v_f$	=	4.0	$m/s$
$A_p$	=	1.27	$cm^2$
$A_m$	=	13	$m^2$
$K_m$	=	$9.218 \times 10^{-9}$	$s/m$
$K_{\Delta\pi}$	=	78.7	$Pa/(mg/L)$
$C_f$	=	10000	$mg/L$
$H$	=	1.0	$mm$
$L$	=	5.0	$m$
$\alpha$	=	0.049	
$C_f^{ss}$	=	10000	$mg/L$
$e_{v1}^{ss}$	=	$3.57 \times 10^7$	
$e_{v2}^{ss}$	=	$1.92 \times 10^8$	
$v_b^{ss}$	=	0.7	$m/s$
$v_r^{ss}$	=	0.3	$m/s$
$v_p^{ss}$	=	3.0	$m/s$
$P^{ss}$	=	$8.61 \times 10^6$	$Pa$

$ss$  corresponds to the high recovery steady-state values for this system when  $C_f^{ss} = 10000 \text{ mg/L}$ , corresponding to a brackish feed water source. Operation at this point provides a recovery of 91%.

One control objective is to stabilize the process at the desired retentate velocity,  $v_r$ , and operating pressure,  $P$ , in the presence of large time varying disturbances in the feed concentration  $C_f$ . This configuration would be used on a system that operates close to the maximum allowable internal pressure. The internal pressure often needs to be below a specified value for safety reasons (safety ratings for fittings and pressure vessels), and at high recovery an RO plant may operate close to this safety threshold. Another control objective could be to stabilize the process at the desired retentate velocity,  $v_r$ , and the desired product flow rate,  $v_p$ . This type of disturbance rejection may be used on RO systems that are designed for extremely high pressures, and allows for a consistent delivery of product water. The controller will use both  $e_{v1}$  and  $e_{v2}$  as manipulated inputs.

To present the controller design method in a concise form, the model of (6.1) is written in a deviation variable form around the desired steady state. The states are defined as  $x = [x_1 \ x_2]^T$  where  $x_1 = v_b - v_b^{ss}$  and  $x_2 = v_r - v_r^{ss}$ . The plant can then be described by the following non-linear continuous-time ODE system:

$$\begin{aligned} \dot{x}(t) &= f(x(t)) + g(x(t))u(t) + w(x(t))d(t) \\ |u_i| &\leq u_i^{max} \end{aligned} \tag{6.4}$$

where  $x(t) \in \mathfrak{R}^2$  denotes the vector of process state variables,  $u(t)$  is a vector of inputs where  $u(t) \in [-u_i^{max}, u_i^{max}] \subset \mathfrak{R}^2$  denotes the  $i^{th}$  constrained manipulated input,  $u_1(t) = e_{v1} - e_{v1}^{ss}$  and  $u_2(t) = e_{v2} - e_{v2}^{ss}$ , and  $d(t)$  denotes the disturbance on the system,  $d(t) = P - P^{ss}$ . The disturbance in this system originates from the feed concentration,  $C_f$ , but  $d(t)$  is expressed in terms of  $P$  because  $C_f$  acts on  $P$  in an

algebraic fashion through (6.3). The control objective is to maintain the outputs at their desired values in the presence of large time varying disturbances on the feed concentration. The state feedback control problem where measurements of all process states are available for all times is considered because velocities,  $v_b$  and  $v_r$ , can be readily measured in practice. The disturbance,  $d(t)$  is available as a measurement of  $C_f$ , and  $C_f$  can be used to calculate  $P$ , and hence,  $d(t)$ . Since  $d(t)$  is readily available  $\hat{f}(x(t)) = f(x(t)) + w(x(t))(d(t))$  is defined.

Next, a Lyapunov-based nonlinear feedback controller that enforces asymptotic stability in the presence of actuator constraints is synthesized. First, a quadratic Lyapunov function of the form  $V_L = x^T P_L x$  is defined where  $P_L$  is a positive-definite symmetric matrix. This Lyapunov function is used to synthesize a bounded nonlinear feedback control law (see [54], [25], and [13]) of the form:

$$u_k = -r(x, u^{max})L_g V_L \quad (6.5)$$

where

$$r = \frac{L_{\hat{f}}^* V_L + \sqrt{(L_{\hat{f}}^* V_L)^2 + (u^{max}|L_g V_L|)^4}}{(|L_g V_L|)^2(1 + \sqrt{1 + (u^{max}|L_g V_L|)^2})} \quad (6.6)$$

and  $L_{\hat{f}}^* V_L = L_{\hat{f}} V_L + \alpha V_L$ ,  $\alpha > 0$ . The scalar function  $r(\cdot)$  in (6.5) and (6.6) can be considered as a nonlinear controller gain.

If the value of  $d(t)$  is available at each time the Lyapunov-based feedback controller of (6.5) and (6.6) employs a feed-forward compensation component. In this case the controller is updated with the latest disturbance information to reject the effects of the disturbance on the states,  $v_b$  and  $v_r$ . In practice it is possible to use a conductivity meter in the feed line to get real-time measurements of the disturbance. However,

if the value of  $d(t)$  is not available for measurement at each time,  $P = P^{ss}$  for the control law and the controller acts in a standard Lyapunov-based feedback manner using a nominal value for  $d(t)$ . In this case, control action is not taken until the states have moved away from the steady-state values due to the disturbance, and the control action does not completely compensate for the disturbance.

## 6.5 Fault detection and isolation and fault tolerant control

In addition to feedback control and disturbance compensation, the problem of actuator fault detection and isolation and fault tolerant control is also addressed. Given the properties of the dynamic model of (6.1), (6.2), and (6.3) it can be shown that the primary control configuration with  $e_{v1}$  and  $e_{v2}$  as manipulated inputs, satisfies the requirements of achieving fault-detection and isolation of actuator faults (see chapter 3 for details). This section presents the methods used to implement fault detection and isolation and fault tolerant control (FDIFTC) on this high recovery RO process. First, the existence of fall-back control configurations is discussed. Next, the construction and explicit forms of FDI filters for the primary configuration are presented. Finally, a switching law that orchestrates the reconfiguration of the control system in a way that provides closed-loop stability in the event of actuator failures is presented.

### 6.5.1 Fall-back control configurations

In order to carry out FTC there must be some redundant control inputs that can be used to control the system in the event of a failure. For this RO system let the initial control configuration,  $k(t = 0) = 1$  be the primary configuration with  $e_{v1}$  and  $e_{v2}$  as manipulated inputs. For the first fall-back configuration consider the system shown in

Fig. 6.1 with an identical fall-back actuator for the retentate valve,  $e_{v2}^{fb}$ . Flow can be diverted from the primary retentate valve ( $e_{v2}$ ) to the fall-back retentate valve ( $e_{v2}^{fb}$ ) through the use of simple on/off valves. Let  $k = 2$  be this fall-back configuration with  $e_{v1}$  and  $e_{v2}^{fb}$  as manipulated inputs. For the second fall-back configuration consider the RO system with an additional fall-back valve for the bypass. Let  $k = 3$  be the fall-back configuration with  $e_{v1}^{fb}$  and  $e_{v2}$  as the manipulated inputs.

### 6.5.2 Fault detection and isolation filters

The FDI filters should enable the detection and isolation of an actuator fault by observing the behavior of the closed-loop process. This is done by using real-time measurements of system states to decouple the ODEs in time. The FDI filter design for the primary control configuration takes the form:

$$\begin{aligned}
\frac{d\tilde{v}_b}{dt} &= \frac{A_p}{\rho V}(\tilde{P}_1 - \frac{1}{2}\tilde{v}_b^2 e_{v1}), & \tilde{v}_b(0) &= v_b(0) \\
\frac{d\tilde{v}_r}{dt} &= \frac{A_p}{\rho V}(\tilde{P}_2 - \frac{1}{2}\tilde{v}_r^2 e_{v2}), & \tilde{v}_r(0) &= v_r(0) \\
r_b &= |v_b - \tilde{v}_b| \\
r_r &= |v_r - \tilde{v}_r|
\end{aligned} \tag{6.7}$$

where  $\tilde{v}_b$  and  $\tilde{v}_r$  are the FDI filter states for the bypass and retentate velocity respectively.  $r_b$  is the residual associated with the bypass valve, and  $r_r$  is the residual associated with the retentate valve.  $\tilde{P}_1$  is a function of  $\tilde{v}_b$  and  $v_r$ .  $\tilde{P}_2$  is a function of  $v_b$  and  $\tilde{v}_r$ .  $\tilde{P}_1$  and  $\tilde{P}_2$  are calculated for each time using the above algorithm and (6.2) and (6.3) with the appropriate values for  $v_b$ ,  $v_r$ ,  $\tilde{v}_b$ , and/or  $\tilde{v}_r$ . The filter states are initialized at the same value as the process states ( $\tilde{v}(0) = v(0)$ ) and essentially predict the evolution of the process in the absence of actuator faults. The residual associated with each manipulated input captures the difference between the predicted



evolution of the states in the absence of a fault on that actuator and the evolution of the measured process state. If a given residual becomes non-zero, a fault is declared on the associated input. For a detailed mathematical analysis of the FDI properties of the filter, the reader may refer to chapter 3.

### 6.5.3 Fault-tolerant supervisory switching logic

The next step is to design a switching logic that the plant supervisor will use to decide what fall-back control configuration to implement given an actuator failure. The supervisor should only implement those configurations that do not utilize a failed actuator. Let  $T_{fault}$  be the time of an actuator failure, and  $T_{detect}$  be the earliest time at which the value of  $r_i(t) > \delta r_i > 0$  (for the  $i$ -th input where  $\delta r_i$  is the  $i$ -th detection threshold). The switching rule given by

$$\begin{aligned} k(t \geq T_{detect}) &= 2 \text{ if } r_r(t) > \delta r_r \\ k(t \geq T_{detect}) &= 3 \text{ if } r_b(t) > \delta r_b \end{aligned} \tag{6.8}$$

guarantees asymptotic closed-loop stability if the new configuration does not include any faulty actuators. The switching law requires monitoring of FDI filters and activation of a fall-back control configuration when a threshold is exceeded.

**Remark 6.2:** In general when considering FDI and FTC for a nonlinear system using the framework proposed in chapter 3 one needs to account for the stability region of each bounded control configuration, and switch only to a control configuration that guarantees stability. However, in the case of an RO system, stability is not the focus because the system is inherently very stable even when operated open-loop, and converges quickly to the steady-state equilibrium point. Therefore, the main goal of feedback control in the RO application is not to enforce stability on the system, but to improve performance and handle events such as actuator faults and feed disturbances.

## 6.6 Simulation results

The simulation results section is divided into two subsections where the first subsection considers large time-varying disturbance on the feed concentration, as shown in Fig. 6.3, and the second subsection considers actuator failures in addition to this disturbance. Time varying disturbances in the feed concentration tend to occur on a long time scale (hours or days), however, failures in the actuators are often sudden and propagate quickly (on the order of 1 second or less).

### 6.6.1 Large time varying disturbance

This section considers the application of three different non-linear control algorithms to handle a large time-varying disturbance in the feed concentration,  $C_f$ . Proportional/Integral (PI) control is also implemented on the system as a point of comparison.

#### **PI feedback control: $P$ and $v_r$ as the controlled outputs**

The first simulation scenario involves using two PI loops to handle the time varying disturbance on the feed concentration, as shown in Fig. 6.3. The first PI loop uses the bypass valve to control the value of the term  $P$ . For the first loop the measurement is  $P$ , and the manipulated input is  $e_{v1}$ . The second PI loop uses the retentate valve to control the state  $v_r$ . For the second loop the measurement is  $v_r$  and the manipulated input is  $e_{v2}$ . The proportional gain,  $K$ , and the integral time,  $\tau_I$ , could not be tuned using standard tuning methods as in [82, 79] because of non-linearities and the coarse grained sampling time. For this reason, the gains and integral time constants were determined through trial and error runs. The system has a sampling time of 60 seconds, and the control is sample and hold. The results can be seen in Figs. 6.8

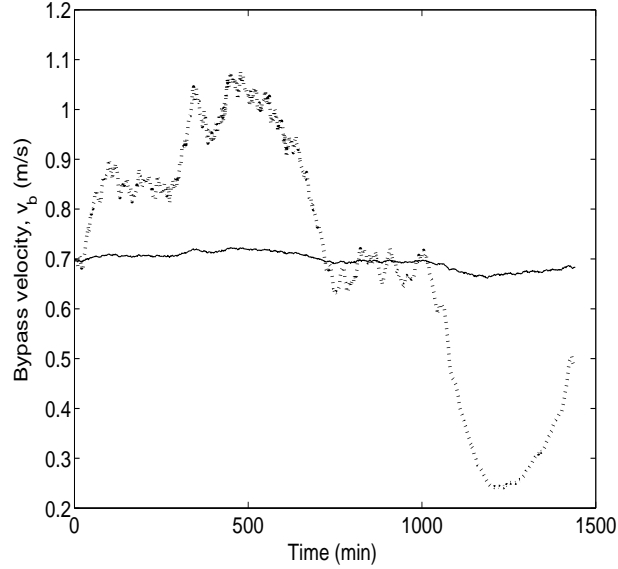


Figure 6.8: Bypass velocity,  $v_b$ , profiles versus time; Open-loop (solid line) and PI control with  $P$  and  $v_r$  as controlled outputs (dashed line).

to 6.12. While PI control is able to reject the disturbance under some conditions, it ultimately fails to keep  $P$  and  $v_r$  at the desired values due to the time varying nature of the disturbance.

### Feedback control: $v_b$ and $v_r$ are the controlled outputs

This simulation scenario involves using the Lyapunov-based control law presented in (6.5). This scenario considers the same dynamic disturbance as in the previous case, where  $C_f$  varies with time according to Fig. 6.3. The states,  $v_b$  and  $v_r$ , are sampled at a rate of one measurement per 60 seconds which is well within the capabilities of existing sensing systems. The control action for the manipulated inputs is computed once per 60 seconds based on these measurements. This control action is implemented for the duration of the sample time, which is 60 seconds, in a sample-and-hold fashion. The disturbance is not measured in this case. The value of  $P$  used in  $\hat{f}(x(t))$  is  $P^{ss}$

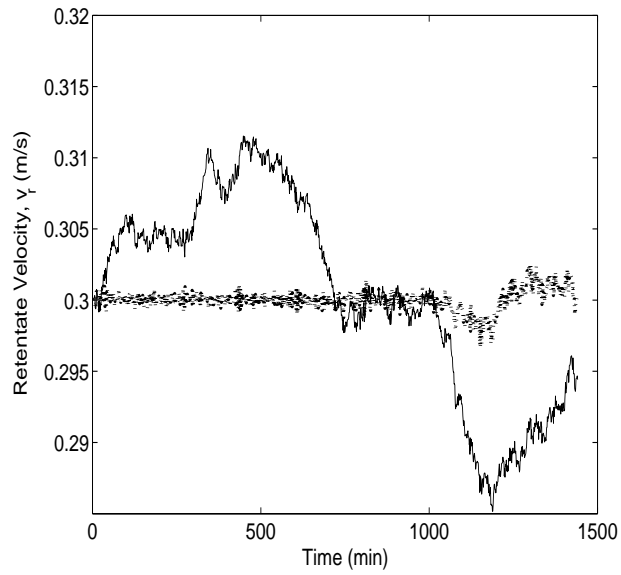


Figure 6.9: Retentate velocity,  $v_r$ , profiles versus time; Open-loop (solid line) and PI control with  $P$  and  $v_r$  as controlled outputs (dashed line).

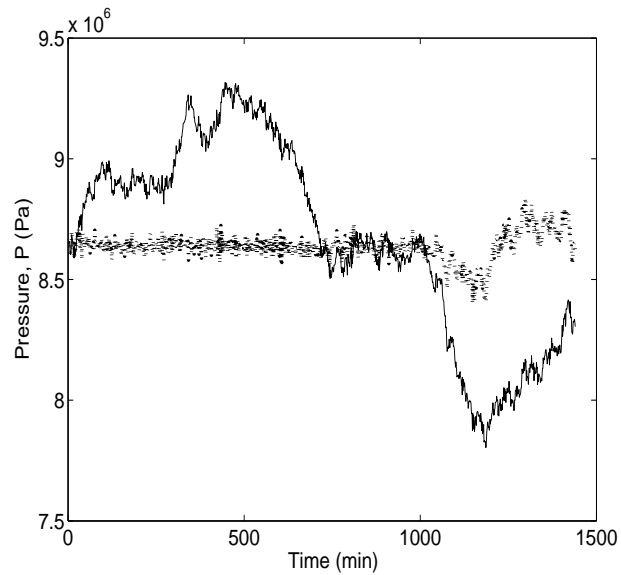


Figure 6.10: Internal pressure,  $P$ , profiles versus time; Open-loop (solid line) and PI control with  $P$  and  $v_r$  as controlled outputs (dashed line).

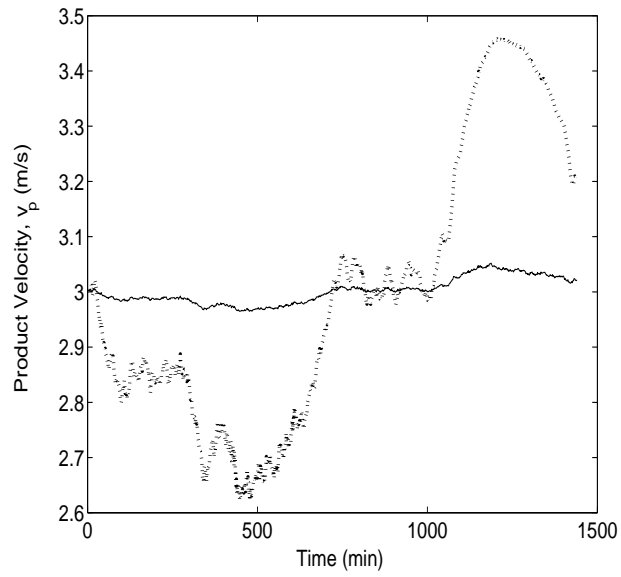


Figure 6.11: Product velocity,  $v_p$ , profiles versus time; Open-loop (solid line) and PI control with  $P$  and  $v_r$  as controlled outputs (dashed line).

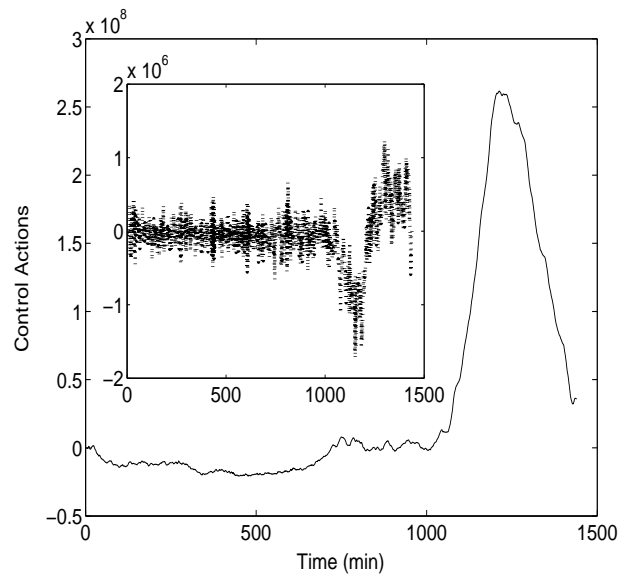


Figure 6.12: Manipulated inputs for the PI controller with  $P$  and  $v_r$  as the controlled outputs. Control action applied to  $e_{v1}$  and  $e_{v2}$  are the solid and dashed lines, respectively.

for all  $t$ , and the controller does not compensate well for the disturbance on  $C_f$ .

The closed-loop simulation results can be seen as the dotted lines in Figs. 6.4 through 6.7. The manipulated inputs can be seen in Fig. 6.13. The states,  $v_b$  and  $v_r$ , and the product flow,  $v_p$ , oscillate at a marginally lower magnitude than the corresponding profiles for the open-loop case, so the feedback control is able to slightly damp out the effects of the disturbance. If the gain on the controller is increased by changing  $P_L$ , it is possible to decrease the disturbance effect further at the expense of higher control actions and possible instability at this sampling rate. However, the pressure oscillates at a somewhat higher magnitude than in the open-loop simulation, and this may not be acceptable for safety reasons. This type of feedback control may be useful for the case where regulating the states and product flow rate is more important than regulating the internal pressure, for example, when the system is being operated at a pressure far below its rated maximum. However, the poor performance of feedback alone motivates the addition of feed-forward compensation to the controller that takes advantage of  $C_f$  measurements.

**Feed-forward/feedback control:  $v_b$  and  $v_r$  are the controlled outputs**

The second simulation scenario involves using the Lyapunov-based control law presented in (6.5), with model-based feed-forward compensation. This technique takes advantage of the dynamic model and the ability to measure  $C_f$  to produce better system performance. For this scenario the time varying nature of  $C_f$  is the same as in the open-loop case. Measurements of the states and of the disturbance,  $C_f$ , are sampled at a rate of one per 60 seconds. At each sampling time a control action is computed and implemented in a sample-and-hold fashion. At each sampling time (6.1), (6.2), and (6.3) are solved for the parameters contained in  $\hat{f}(x(t))$  correspond-

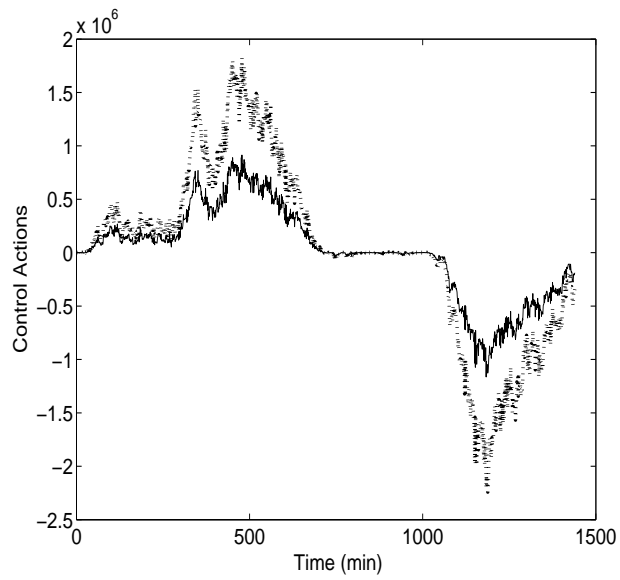


Figure 6.13: Manipulated inputs for the Lyapunov-based feedback controller with no feed-forward compensation with  $v_b$  and  $v_r$  as the controlled outputs. Control actions applied to  $e_{v1}$  and  $e_{v2}$  are the solid and dashed lines, respectively.

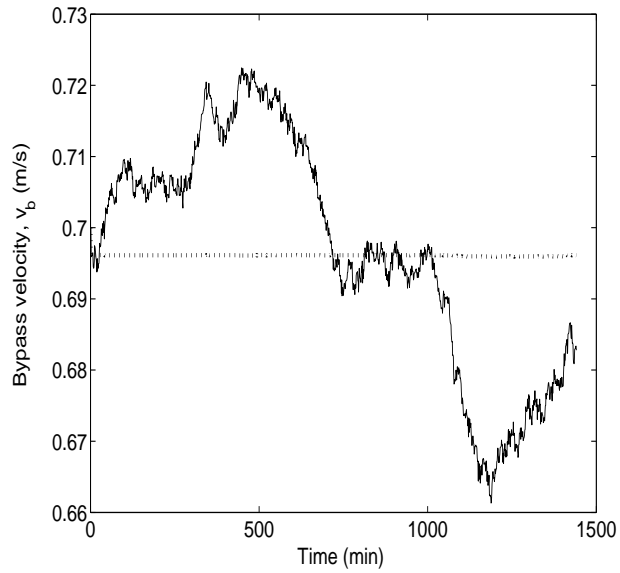


Figure 6.14: Bypass velocity,  $v_b$ , profiles versus time; Open-loop (solid line) and under feed-forward/feedback control with  $v_b$  and  $v_r$  as controlled outputs (dashed line).

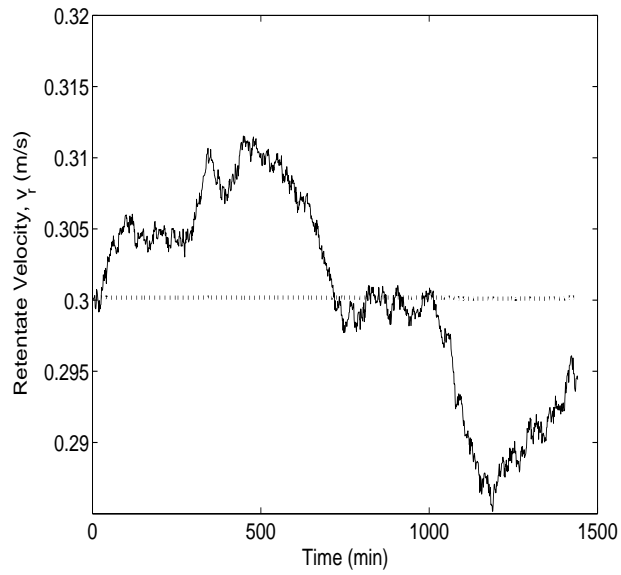


Figure 6.15: Retentate velocity,  $v_r$ , profiles versus time; Open-loop (solid line) and under feed-forward/feedback control with  $v_b$  and  $v_r$  as controlled outputs (dashed line).

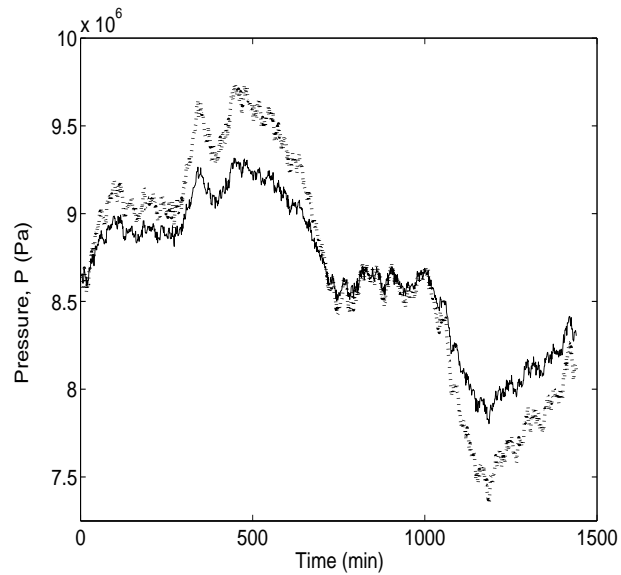


Figure 6.16: Internal pressure,  $P$ , profiles versus time; Open-loop (solid line) and under feed-forward/feedback control with  $v_b$  and  $v_r$  as controlled outputs (dashed line).



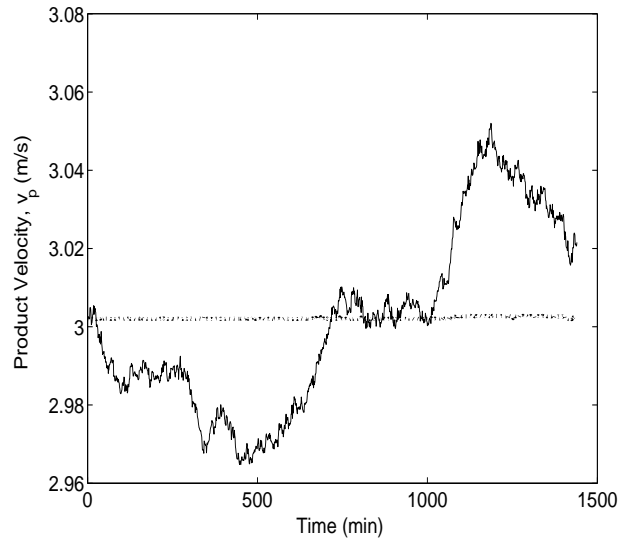


Figure 6.17: Product velocity,  $v_p$ , profiles versus time; Open-loop (solid line) and under feed-forward/feedback control with  $v_b$  and  $v_r$  as controlled outputs (dashed line).

ing to the current  $C_f$  value and the desired  $v_p$  and  $v_r$  values. This can be done with the following steps:

1. Choose set points for  $v_b$  and  $v_r$ , in this case 0.7 and 0.3  $m/s$  respectively to achieve a recovery of over 90%.
2. Solve (6.2) for  $v_{mf}$ .
3. Find the appropriate  $P$ , via shooting method, to satisfy all the boundary conditions for (6.3).
4. Set (6.1) equal to zero, and solve for  $e_{v1}^{nom}$  and  $e_{v2}^{nom}$ . These are the nominal values for the manipulated inputs that will compensate for the current disturbance,  $C_f(t)$ , and are components of  $\hat{f}(x(t))$ .

The control law in (6.5) is used to compute a feedback control action based on the current  $\hat{f}(x(t))$  obtained from the above algorithm. This control action is added to

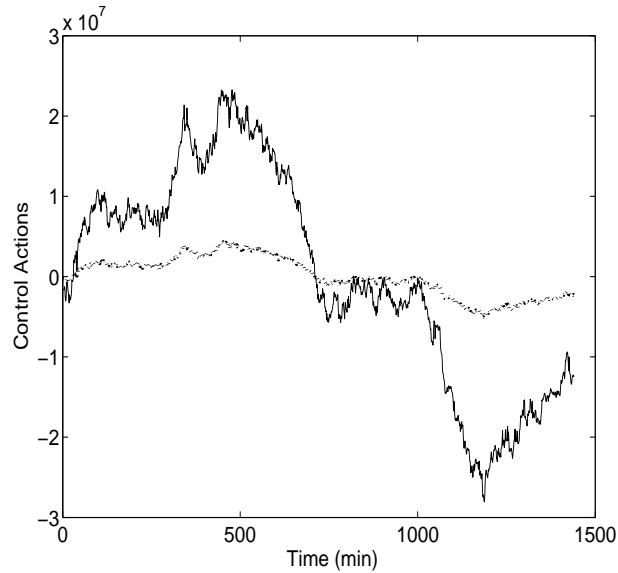


Figure 6.18: Manipulated inputs for the feed-forward/feedback controller with  $v_b$  and  $v_r$  as the controlled outputs. Control actions applied to  $e_{v1}$  and  $e_{v2}$  are the solid and dashed lines, respectively.

the nominal  $e_{v1}^{nom}$  and  $e_{v2}^{nom}$  values, and implemented on the valves. This process is repeated at each sampling time to obtain a feedback control action that includes feed-forward compensation. The manipulated inputs can be seen in Fig. 6.18 and compared to Fig. 6.13. The control actions are larger, yet they are within reasonable actuator limits.

The simulation results can be seen as the dashed lines in Figs. 6.14 through 6.17. The values of  $v_b$ ,  $v_r$ , and  $v_p$  all stay very close to the steady-state values given in Table 6.1, and the effects of the disturbance are effectively damped. A shorter sampling interval would reduce the disturbance effects even further. In this case, the value of  $P$  swings sharply in order to compensate for the changing feed conditions. To achieve the desired recovery of over 90% even when the  $C_f(t)$  is much higher than  $C_f^{ss}$  requires very high pressures. This type of control would be advantageous only when product flow rate is a critical parameter that cannot be disturbed, and the RO

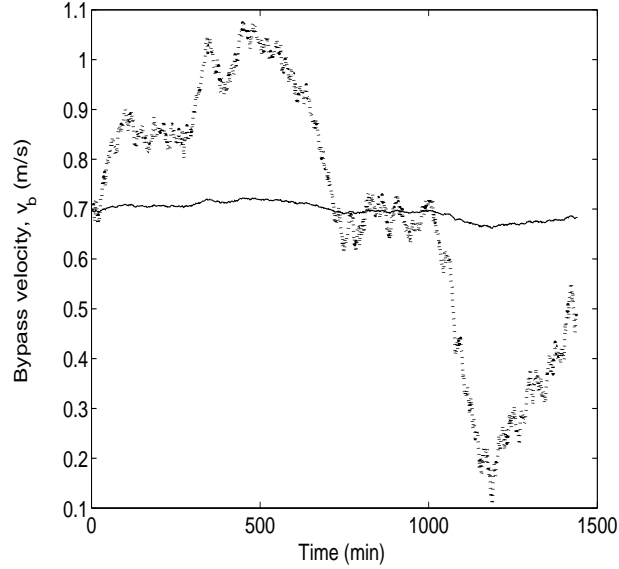


Figure 6.19: Bypass velocity,  $v_b$ , profiles versus time; Open-loop (solid line) and under feed-forward/feedback control with  $P$  and  $v_r$  as controlled outputs (dash-dotted line).

system is designed to handle such high internal pressures.

### **Feed-forward/feedback control: $P$ and $v_r$ are the controlled outputs**

The third simulation scenario does not fall directly under the Lyapunov-based feed-forward/feedback framework utilized in the previous two simulations, however, it is an important one from a practical point of view. For safety reasons, the large internal pressures exemplified in the previous examples motivate the use of feed-forward/feedback control that maintains  $P(t)$  at a constant value,  $P^{ss}$ . In order to accomplish this, another variable (either  $v_b$  or  $v_r$ ) must be used to compensate for the effects of  $C_f$ . The flow rate  $v_r$  is often constrained due to the membrane module capacity, so  $v_b$  is an excellent candidate for this role. The bypass velocity can vary widely with little to no ill-effect on the system:  $v_b$  is readily recycled, there are usually no downstream lines that depend on  $v_b$ , and there are no dominant safety

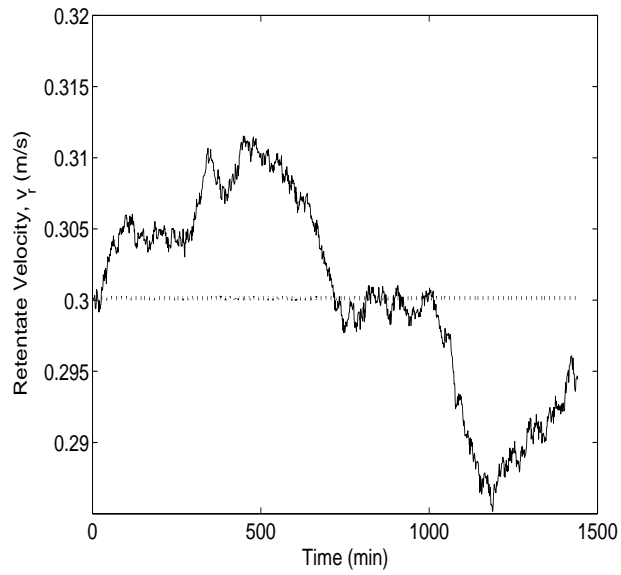


Figure 6.20: Retentate velocity,  $v_r$ , profiles versus time; Open-loop (solid line) and under feed-forward/feedback control with  $P$  and  $v_r$  as controlled outputs (dash-dotted line).

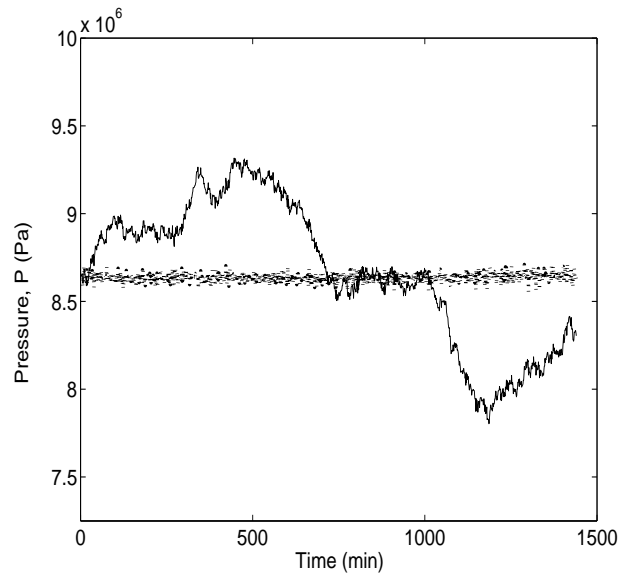


Figure 6.21: Internal pressure,  $P$ , profiles versus time; Open-loop (solid line) and under feed-forward/feedback control with  $P$  and  $v_r$  as controlled outputs (dash-dotted line).

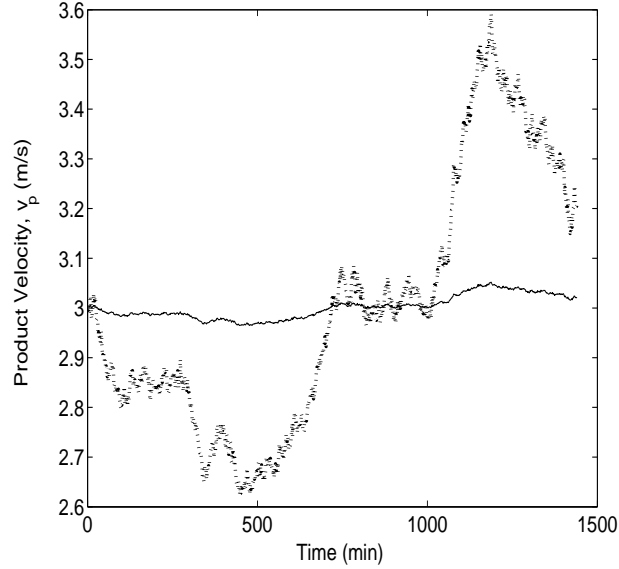


Figure 6.22: Product velocity,  $v_p$ , profiles versus time; Open-loop (solid line) and under feed-forward/feedback control with  $P$  and  $v_r$  as controlled outputs (dash-dotted line).

issues associated with wide  $v_b$  variations.

The third simulation scenario involves using a Lyapunov-based nonlinear feedback control law similar to the one presented in (6.5). Again,  $C_f$  is the same as the previous scenarios, and measurements of the states and disturbance are obtained at a rate of one sample per 60 seconds. The control action is implemented in a sample-and-hold fashion.

The framework for the feedback control with feed-forward compensation with  $P$  and  $v_r$  as the controlled outputs is slightly different than the one used in the previous two examples. Specifically, at each sampling time (6.1), (6.2) and (6.3) are solved for the steady-state corresponding to the current  $C_f$  value and the desired  $P$  and  $v_r$  values. This can be done with the following steps:

1. Choose set points for  $P$  and  $v_r$ , in this case  $8.6 \times 10^6$  Pa and  $0.3$  m/s respectively.
2. Solve (6.3) with the following two boundary conditions using shooting method

where an initial guess is made on  $v_z(z = 0)$ :

(a)  $C_z(z = 0) = C_f$

(b)  $v_z(z = L) = \alpha v_r$

3. The resulting value of  $v_z(z = 0)$  from the previous step is used to calculate  $v_{mf}$
4.  $v_{mf}$  is used with (6.2) to calculate a desired value for  $v_b$ . This  $v_b$  and the set point for  $v_r$  designate a new desired operating point where  $P(t) = P^{ss}$
5. Set (6.1) equal to zero, substitute in the values for  $P$ ,  $v_r$ , and  $v_b$ , and solve for  $e_{v1}^{nom}$  and  $e_{v2}^{nom}$ .

The control law in (6.5) is then used to compute a control action based on this new operating point provided from the above algorithm. This control action is added to the  $e_{v1}^{nom}$  and  $e_{v2}^{nom}$  values from the above algorithm, and implemented on the valves. This process is repeated at each sample time to obtain a new operating point and compute a control action that has feed-forward and feedback components. In other words, at each sampling time the steady state problem of (6.1), (6.2), and (6.3) is solved to find the desired operating point where  $P(t) = P^{ss}$  and  $v_r(t) = v_r^{ss}$ , and a control action from a Lyapunov-based control law is implemented based on this new operating point. The manipulated input profiles resulting from this control algorithm,  $e_{v1}$  and  $e_{v2}$ , are shown in Fig. 6.23.

The closed-loop feed-forward/feedback control with  $P$  and  $v_r$  as controlled outputs can be seen as the dash-dotted line in Figs. 6.19 through 6.22. In this case, the pressure,  $P$ , stays close to the desired set point, and the effects of the disturbance on pressure and retentate velocity are largely damped. To maintain this pressure, however, the bypass velocity,  $v_b$ , now varies to a large degree to act as a buffer and

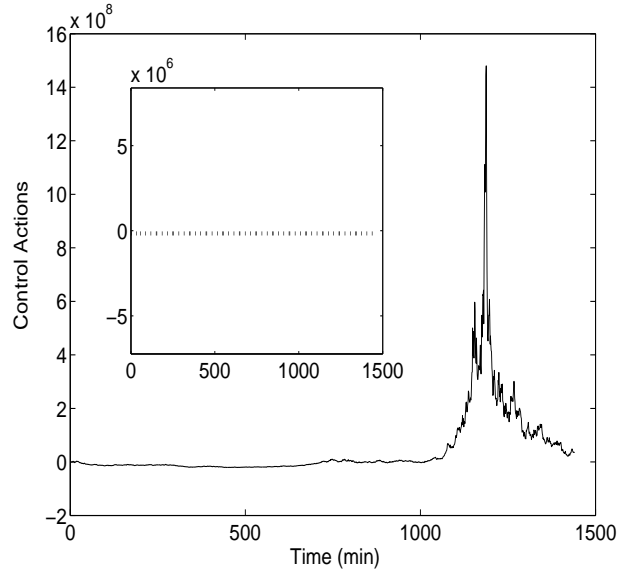


Figure 6.23: Manipulated inputs for the feed-forward/feedback controller with  $P$  and  $v_r$  as the controlled outputs. Control actions applied to  $e_{v1}$  and  $e_{v2}$  are the solid and dashed lines, respectively.

absorb the effects of the feed disturbance. The manipulated input  $e_{v1}$  also varies to accomplish this control task. This type of feed-forward/feedback control is the best to use in a situation where the plant is operating close to the high pressure constraints, which is usually the case at very high recoveries. This type of control is desirable because the bypass velocity can vary widely with little to no ill-effects on the system:  $v_b$  is readily recycled, there are usually no downstream structures that depend on  $v_b$ , and there are no major safety issues associated with wide  $v_b$  variations.

**Remark 6.3:** Energy efficiency is often a critical concern in the operation of RO plants to minimize environmental and economic costs. Inherently, a bypass line without an energy recovery device is an energy waster because pressurized feed water is throttled, and energy is lost to friction. It is possible to operate an RO system under a feed-forward/feedback framework as described above using a variable frequency drive (VFD) on the pump. In this case, the control system could regulate the VFD and

pump speed in order to change system pressure and flow rather than wasting energy by sending pressurized water through a bypass line. For example, if a VFD was used,  $v_f$  could be considered as a manipulated input, and  $v_b$  could be removed from the system. For safety reasons, however, emergency bypass lines that open at a high pressure threshold should still be installed to prevent the accidental over-pressurization of the system.

### 6.6.2 Actuator failures

This section considers the application of FDI and FTC to handle valve actuator failures. This plant model is the same as the system used in the previous example with the same time varying disturbance on the feed concentration. Additionally the system is considered to have noisy sampled measurements. The retentate velocity measurements are subject to gaussian noise with a standard deviation of  $6 \times 10^{-4} \frac{m}{s}$ , and the bypass velocity measurements are subject to gaussian noise with standard deviation of  $1.4 \times 10^{-3} \frac{m}{s}$ . The standard deviation of the noise is 0.2% of the nominal flow values. In order to isolate a failure, the sampling time must be much faster than the system dynamics. If the sampling time is too slow, then a failure occurring between sampling times will propagate to all system states, and the FDI filter will not function properly. An adequate sampling time can be estimated by examining the open-loop response time of the system. The sampling time must be significantly shorter than the response time. This is practically possible given that the filter only requires measurements of  $v_b$  and  $v_r$ . Furthermore, while the FDI component requires a fast sampling rate in order to do isolation, the FTC reconfiguration can happen on a much slower timescale. FTC reconfiguration could be delayed for several moments after a fault is isolated, however the states may move to an undesired region during this delay. This



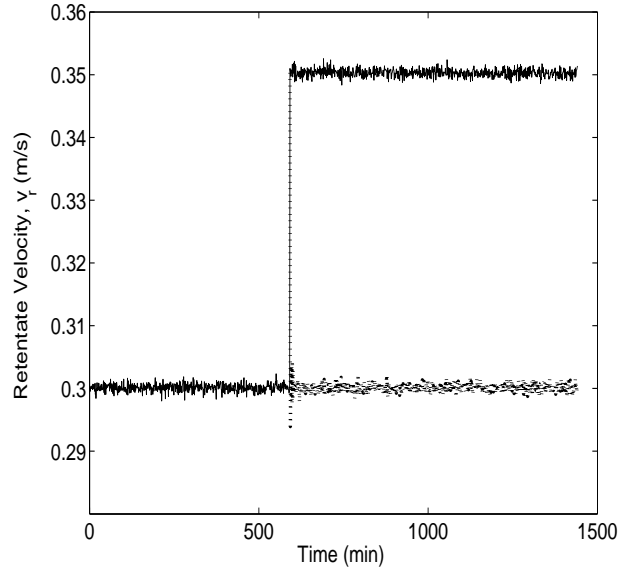


Figure 6.24: Retentate velocity,  $v_r$ , profile versus time; subject to a failure in  $e_{v2}$  (solid line) and with FDIFTC recovery (dotted line).

is clearly better than the alternative of not implementing FDIFTC where the state may move to an undesired region for all time after an actuator fault. For two simulations measurements are available continuously, and the control is sample-and-hold every 60 seconds. It is also assumed that FTC reconfiguration takes place at the next sample and hold interval after detection. A third example displays how FDI performs with sampled measurements that allow for detection but not isolation. The FDIFTC framework allows the resilient operation of the RO system in the presence of valve actuator failures. For this section it is assumed that the fall-back configurations  $k = 2$  and  $k = 3$  discussed in the FDIFTC section are available for the operator to use.

### Failure of the retentate valve

For this simulation, the RO system is subjected to a failure in the retentate valve at  $t = 35,424$  s where the value of  $e_{v2}$  gets fixed at  $1.4 \times 10^8$ . The control in this

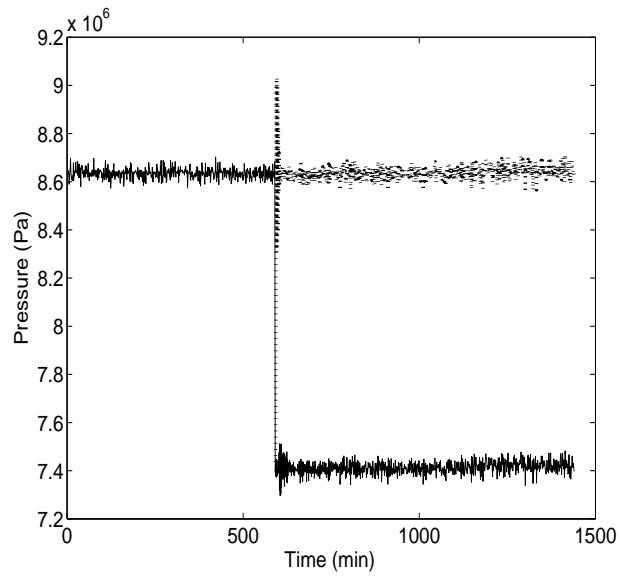


Figure 6.25: System pressure,  $P$ , profile versus time; subject to a failure in  $e_{v2}$  (solid line) and with FDIFTC recovery (dotted line).

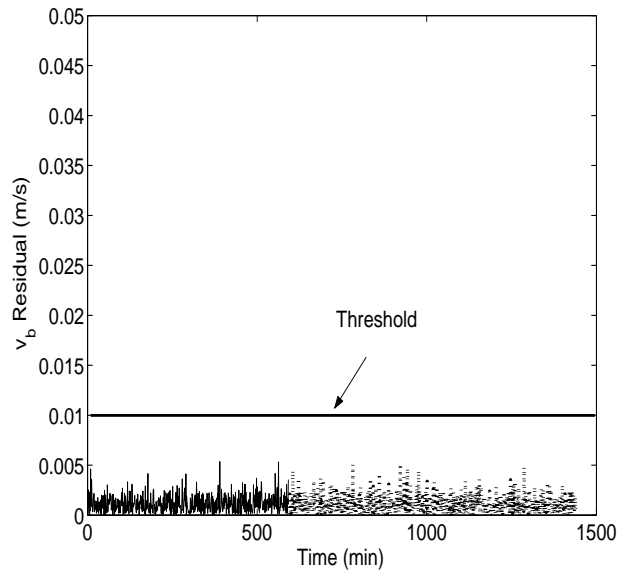


Figure 6.26: Residual corresponding to the bypass valve versus time. No fault is detected on the bypass valve. The solid line is under the primary control configuration, and the dotted line is under the fall-back configuration.

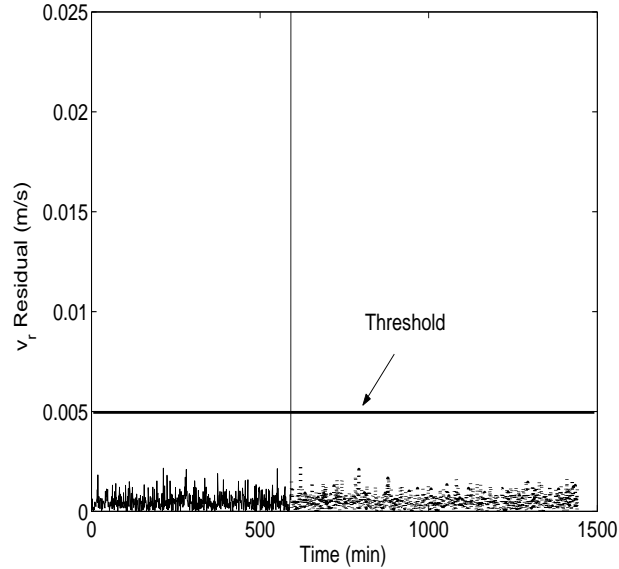


Figure 6.27: Residual corresponding to the retentate valve versus time. A fault is detected in this valve at  $t = 592 \text{ min}$ . The solid line is under the primary control configuration, and the dotted line is under the fall-back configuration.

case is nonlinear feedback control with feed-forward compensation as in section 6.1.4. Measurements are assumed to be available continuously, while control is implemented sample-and-hold with a hold time of 60 seconds. The profiles for retentate velocity,  $v_r$ , and pressure,  $P$ , with and without FDIFTC recovery can be seen in Figs. 6.24 and 6.25. It is clear that if no FDI is used, shown by the solid lines in 6.24 and 6.25, then the system states move away from the desired set-point values. However, the FDI filters shown in Eq. 6.7 can be used with this system to generate residual plots shown in Figs. 6.26 and 6.27. The sampling time is fast enough to effectively detect and isolate the failure, so only one of the two residuals responds to the fault. It is clear from the residual plots that the failure has occurred in the retentate valve, and not in the bypass valve. This actuator fault isolation could not have been done with inspection of the states alone, because both  $v_b$  and  $v_r$  change significantly upon failure. At the time of detection,  $t = 592 \text{ min}$ , the system is switched to the appropriate fall-

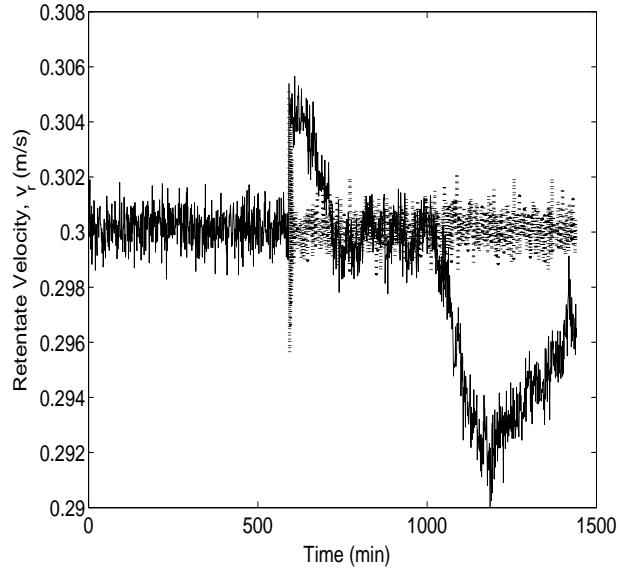


Figure 6.28: Retentate velocity,  $v_r$ , profile versus time; subject to a failure in  $e_{v1}$  (solid line) and with FDIFTC recovery (dotted line).

back configuration under FDIFTC,  $k = 2$  in this case, and the system returns to the desired operating point. At the time of detection the FDI filter is initialized at the current state, and is ready to detect an actuator failure in the new configuration.

### Failure of the bypass valve

This example explores a sudden failure in the bypass valve. At  $t = 35,424$  the bypass valve resistance goes to the nominal value,  $e_{v1} = e_{v1}^{ss}$ , and is fixed. The profiles of the retentate velocity,  $v_r$ , and the pressure,  $P$ , with FDIFTC recovery can be seen in Figs. 6.28 and 6.29.

It is clear that if no FDI is used (solid lines in Figs. 6.28 and 6.29) then the system states respond, and the system moves to an undesirable operating mode. However, FDIFTC can be implemented to regain control. It can be seen in Figs. 6.30 and 6.31 that the failure has occurred in the bypass line. According to the FDIFTC switching

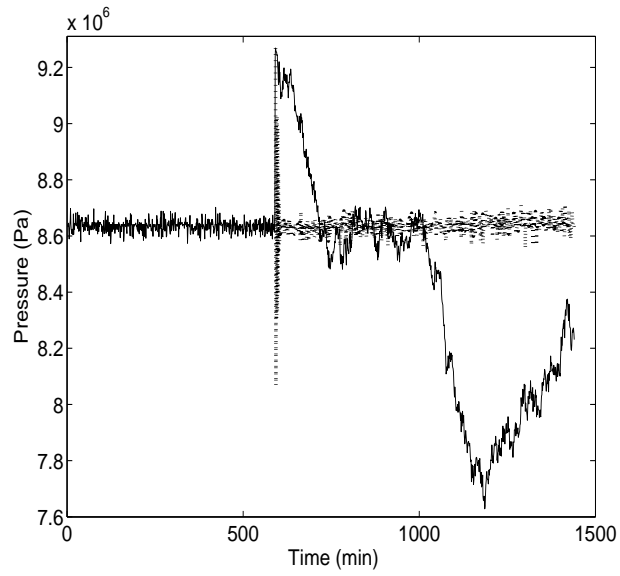


Figure 6.29: System pressure,  $P$ , profile versus time; subject to a failure in  $e_{v1}$  (solid line) and with FDIFTC recovery (dotted line).

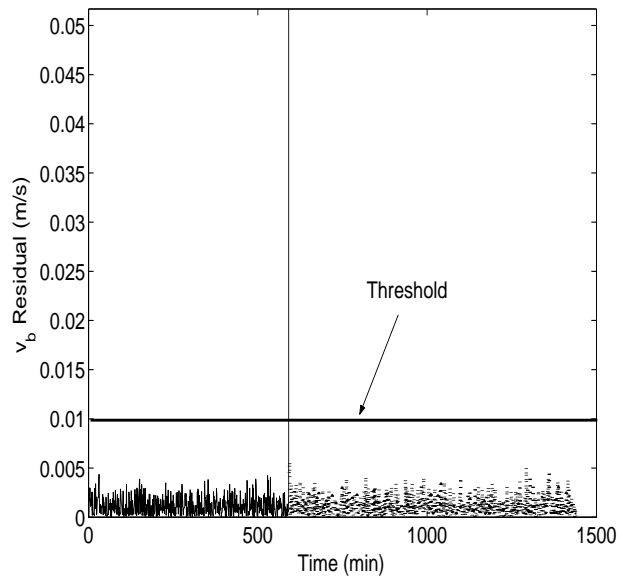


Figure 6.30: Residual corresponding to the bypass valve versus time. A fault is detected in this valve at  $t = 592 \text{ min}$ . The solid line is under the primary control configuration, and the dotted line is under the fall-back configuration.

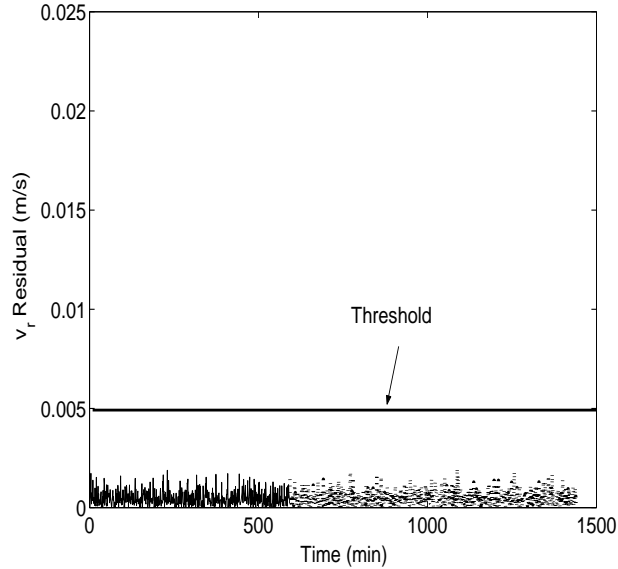


Figure 6.31: Residual corresponding to the retentate valve versus time. No fault is detected on the retentate valve. The solid line is under the primary control configuration, and the dotted line is under the fall-back configuration.

logic, the system can switch to the fall-back configuration where  $k = 3$ . This fall-back configuration uses a fall-back bypass valve to replace  $e_{v1}$ , and the controller is able to move the system back to the desired operating point. The FDI filter is initialized after reconfiguration to isolate additional actuator failures. The system recovery under FDIFTC can be seen as the dotted lines in Figs. 6.28 and 6.29.

### **Failure of the bypass valve; sampled measurements**

The final example presents a case where continuous measurements of the system states are not available. Limitations on sampling time are imposed by the dynamic behavior of flow meters and other sensors. Specifically, flow meters have a dynamic response time that can be characterized by a time constant, and the flow signal takes some finite time to adjust to changes in the pipe flow [82]. To this end, sample-and-hold operation, with a sampling time of 60 seconds, is implemented. The sensor dynamics

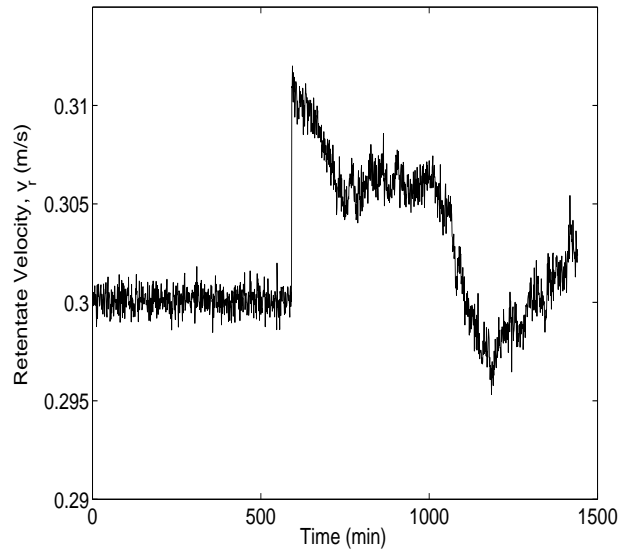


Figure 6.32: Retentate velocity,  $v_r$ , profile versus time; subject to a failure in  $e_{v1}$ .

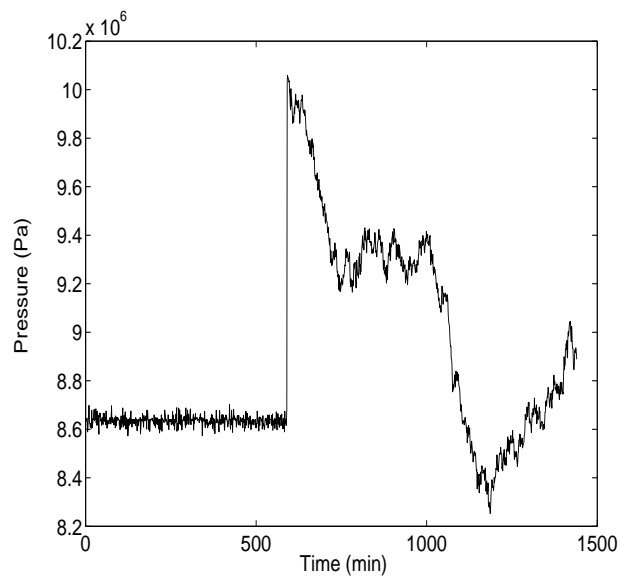


Figure 6.33: System pressure,  $P$ , profile versus time; subject to a failure in  $e_{v1}$ .

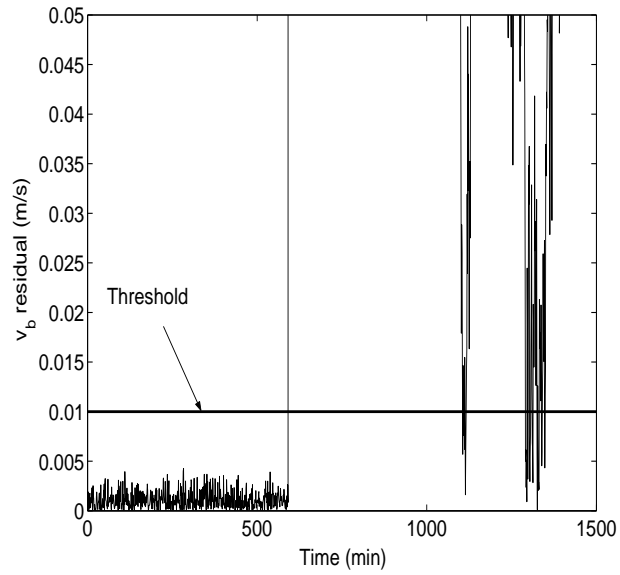


Figure 6.34: Residual corresponding to the bypass valve versus time. The residual is exceeded at  $t = 592 \text{ min}$ .

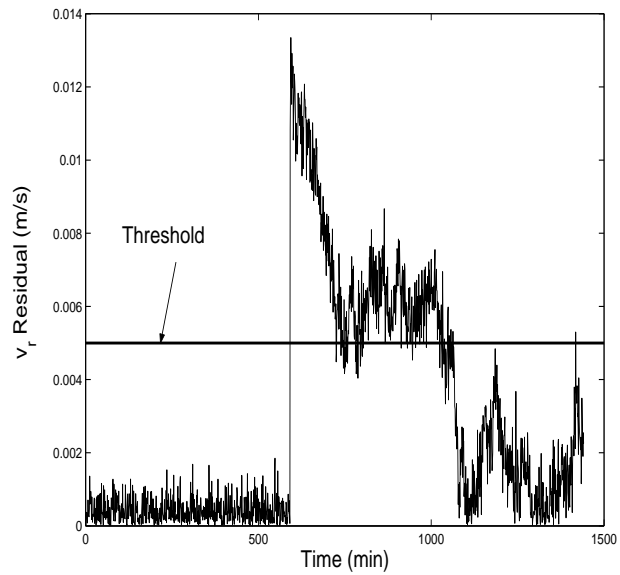


Figure 6.35: Residual corresponding to the retentate valve versus time. The residual is exceeded at  $t = 592 \text{ min}$ .



are assumed to be fast compared to this sampling time, and are neglected.

This example includes a sudden failure in the bypass valve; at  $t = 35,424$  s the bypass valve resistance goes to  $1.5 \times 10^8$  and is fixed. The profiles of the retentate velocity,  $v_r$ , and the pressure,  $P$ , with FDIFTC recovery can be seen in Figs. 6.32 and 6.33. Both states diverge simultaneously due to this failure.

The FDI filters, shown in Figs. 6.34 and 6.35, cannot isolate the failure. The failure propagates to both states between consecutive measurements, and thus both residuals exceed the threshold so that it is impossible to isolate the failed actuator. However, the FDI filters still provide fault detection, and the operator could take action at the time of detection to examine the system and search for source of the failure. Even though FDI is impossible with a large sample period for this system, in practice the dynamics of an RO plant may not be as fast as the dynamics of the model proposed in (6.1). Slower system dynamics would allow for FDI filters to perform adequately even under sampled measurements if the sample time is sufficiently fast compared to the system dynamics.

## 6.7 Conclusions

The contributions of this chapter include the development of a dynamic model for high recovery RO desalination. This model describes the spatial and temporal behavior of a high recovery RO desalination process. Additionally, nonlinear control techniques that include feed-forward/feedback control for disturbance rejection and FDIFTC have been applied to this dynamic model accounting for practical issues such as noisy/sampled measurements, large time varying disturbances, and actuator failures. Nonlinear Lyapunov-based feed-forward/feedback controllers were implemented on the high recovery RO system in simulation examples. The additional feed-

forward component in the controller was able to compensate for large time varying disturbances in the feed concentration. FDIFTC methods were applied in simulation examples in order to detect actuator faults and switch appropriately to fall-back configurations avoiding undesired RO system operation.

The first set of simulation studies examined the ability of the feed-forward/feedback control algorithms to handle a large time varying disturbance on the feed concentration. These simulations account for such practical issues as sampled measurements and time-varying disturbances. The first feed-forward/feedback control simulation demonstrated the ability to mitigate disturbances with the system states,  $v_b$  and  $v_r$ , as controlled outputs. The pressure,  $P$ , in this simulation varied to a large extent (a possible safety concern), and this  $P$  variation motivated the application of feed-forward/feedback control with the pressure and retentate,  $P$  and  $v_r$ , as controlled outputs. The second feed-forward/feedback simulation demonstrated the ability to mitigate the effect of the disturbance on the system pressure,  $P$ . The second set of simulation studies demonstrated the application of a fault-detection and isolation and fault tolerant control structures for this RO system. These simulations account for such practical issues as sampled noisy measurements and plant/model parameter mismatch. The first FDIFTC simulation demonstrated the detection, isolation, and appropriate switching when the system is subjected to a failure on the retentate valve. The second FDIFTC simulation demonstrated the detection, isolation, and appropriate switching when the system is subjected to a failure on the bypass valve.

## Chapter 7

# Fault-Detection and Isolation for Nonlinear Process Systems Using Asynchronous Measurements

### 7.1 Introduction

This work addresses the problem of fault-detection and isolation for nonlinear processes when process measurements are available at asynchronous time instants. First, a fault-detection and isolation (FDI) scheme that employs model-based techniques is proposed that allows for the isolation of faults. This scheme employs model-based FDI filters similar to those proposed in chapter 3 in addition to observers that estimate the fault free evolution of asynchronously measured states during time intervals in which their measurements are not available. Specifically, the proposed FDI scheme provides detection and isolation of any fault that enters into the differential equation of only synchronously measured states, and grouping of faults that enter into the differential equation of any asynchronously measured state. For a fully coupled process system, fault-detection occurs shortly after a fault takes place, and fault isolation, limited by

the arrival of asynchronous measurements, occurs when asynchronous measurements become available. Once the FDI methodology has provided the system supervisor with a fault diagnosis, the supervisor takes appropriate action to seamlessly reconfigure the system to an alternative control configuration that will enforce the desired operation. We present applications of the proposed asynchronous FDI and FTC framework to a polyethylene reactor simulation [60]. Specifically, the polyethylene reactor includes six state variables such as temperatures, species concentrations, and catalyst activity. This polyethylene plant naturally gives rise to measurements that can be sampled synchronously (such as temperature), and those that are sampled asynchronously (such as reactant and catalyst concentrations). Previous work of our group [35] considered fault-tolerant control of the polyethylene reactor assuming that all process variables are continuously measured. In the present work, it is shown through a detailed simulation study that the proposed model-based asynchronous FDI technique can lead to reliable actuator fault detection and isolation and fault tolerant control in a timely manner.

## 7.2 FDI using asynchronous measurements: Problem formulation and solution

### 7.2.1 Class of nonlinear systems

In this work, we consider nonlinear process systems described by the following state-space model

$$\begin{aligned}
 \dot{x}_1 &= f_1(x_1, \dots, x_{n_s}, x_{n_s+1}, \dots, x_{n_s+n_a}, u, d_1, \dots, d_p) \\
 &\vdots \\
 \dot{x}_{n_s} &= f_{n_s}(x_1, \dots, x_{n_s}, x_{n_s+1}, \dots, x_{n_s+n_a}, u, d_1, \dots, d_p) \\
 \dot{x}_{n_s+1} &= f_{n_s+1}(x_1, \dots, x_{n_s}, x_{n_s+1}, \dots, x_{n_s+n_a}, u, d_1, \dots, d_p) \\
 &\vdots \\
 \dot{x}_{n_s+n_a} &= f_{n_s+n_a}(x_1, \dots, x_{n_s}, x_{n_s+1}, \dots, x_{n_s+n_a}, u, d_1, \dots, d_p)
 \end{aligned} \tag{7.1}$$

where  $x_i \in R$  with  $i = 1, \dots, n_s$  denotes the set of state variables that are sampled synchronously,  $x_i \in R$  with  $i = n_s+1, \dots, n_s+n_a$  denotes the set of state variables that are sampled asynchronously,  $u \in R^{n_u}$  denotes the input and  $d_i \in R$  with  $i = 1, \dots, p$  is a model of the set of  $p$  possible faults. The faults are unknown and  $d_i$  can take any value. The state of the system is given by the vector

$$x = \begin{bmatrix} x_1 \\ \vdots \\ x_{n_s} \\ x_{n_s+1} \\ \vdots \\ x_{n_s+n_a} \end{bmatrix} \in R^{n_s+n_a}$$

Using this definition for  $x$ , (7.1) can be written in the following equivalent compact form

$$\dot{x} = f(x, u, d_1, \dots, d_p) \quad (7.2)$$

We assume that  $f$  is a locally Lipschitz vector function and that  $f(0, 0, 0, \dots, 0) = 0$ . This means that the origin is an equilibrium point for the fault-free system with  $u(t) \equiv 0$ . Moreover, we assume that the fault-free system (system (7.1) with  $d_i(t) \equiv 0$  for all  $t$ ) has an asymptotically stable equilibrium at the origin  $x = 0$  for a given feedback control  $h : R^{n_s+n_a} \rightarrow R^{n_u}$  which satisfies  $h(0) = 0$ .

**Remark 7.1:** The assumption of existence of a stabilizing feedback law  $h(x)$  is equivalent to the existence of a control Lyapunov function (CLF) for the system  $\dot{x} = f(x, u, 0, \dots, 0)$ . Explicit stabilizing control laws that provide explicitly-defined regions of attraction for the closed-loop system have been developed using Lyapunov techniques for specific classes of nonlinear systems, particularly input-affine nonlinear systems; the reader may refer to [51, 13] for results in this area. In Section 7.3, a method such as the one presented in [88] is used for the design of  $h(x)$ .

### 7.2.2 Modeling of asynchronous measurements

System (7.1) is controlled using both sampled synchronous and asynchronous measurements. We assume that  $x_i(t) \in R$  with  $i = 1, \dots, n_s$  are available continuously (i.e., at intervals of fixed size  $\Delta > 0$  where  $\Delta$  is a sufficiently small positive number). Each state  $x_i \in R$  with  $i = n_s + 1, \dots, n_s + n_a$  is sampled asynchronously and is only available at time instants  $t_{k,i}$  where  $t_{k \geq 0, i}$  is a random increasing sequence of times. A controller design that takes advantage of the asynchronous measurements must take into account that it will have to operate without complete state information between asynchronous samples. This class of systems arises naturally in process control, where process variables such as temperature, flow, or concentration have to be measured. In such a case, temperature and flow measurements can be assumed to be available continuously. Concentration measurements, however, are available at an asynchronous sampling rate. This model is also of interest for systems controlled through a hybrid communication network in which wireless sensors are used to add redundancy to existing working control loops (which use point-to-point wired communication links and continuous measurements) because wireless communication is often subject to data losses due to interference.

If there exists a non-zero probability that the system operates in open-loop for a period of time large enough for the state to leave the stability region or even diverge to infinity (i.e., finite escape time), it is not possible to provide guaranteed stability properties. In order to study the stability properties in a deterministic framework, in this paper we consider systems where there is a limit on the maximum number of consecutive sampling times in which measurements of  $x_i$ ,  $i = n_s \dots n_s + n_a$ , are not available, i.e.

$$\Delta_M \geq \max t_{k,i} - t_{k+1,i}$$

This bound on the maximum period of time in which the loop is open has been also used in other works in the literature [93, 94, 73, 72, 68] and allows us to study deterministic notions of stability.

### 7.2.3 Asynchronous state observer

An observer that takes advantage of synchronous measurements, asynchronous measurements, and a process model can be constructed to estimate the fault-free evolution of asynchronous states between consecutive measurements. The observer states are updated by setting the observer state equal to the measurement each time a new asynchronous measurement becomes available,  $t_{k,i}$ . The asynchronous state observer takes the form

$$\begin{aligned}\dot{\hat{x}}_{n_s+1} &= f_{n_s+1}(x_1, \dots, x_{n_s}, \hat{x}_{n_s+1}, \dots, \hat{x}_{n_s+n_a}, u, 0, \dots, 0) \\ &\vdots \\ \dot{\hat{x}}_{n_s+n_a} &= f_{n_s+n_a}(x_1, \dots, x_{n_s}, \hat{x}_{n_s+1}, \dots, \hat{x}_{n_s+n_a}, u, 0, \dots, 0)\end{aligned}\tag{7.3}$$

with  $\hat{x}_i(t_{k,i}) = x_i(t_{k,i})$  for all  $t_{k,i}$ ; that is, each time a new asynchronous measurement is received, the estimated states  $\hat{x}_i$  with  $i = n_s + 1, \dots, n_s + n_a$  are reset to match the true process state. The information generated by this observer provides a fault-free estimate for each asynchronous state at any time  $t$  and allows for the design of non-linear control laws that utilize full state information. Using the estimated states, the control input applied to the system is given by

$$u = h(\hat{x})\tag{7.4}$$

with

$$\hat{x} = \begin{bmatrix} x_1 \\ \vdots \\ x_{n_s} \\ \hat{x}_{n_s+1} \\ \vdots \\ \hat{x}_{n_s+n_a} \end{bmatrix} \in R^{n_s+n_a}$$

This control input is defined for all times because it is based on both the synchronous states and the estimated asynchronous states. We assume that  $\Delta_M$  is small enough to guarantee that the system in closed-loop with this control scheme is practically stable, see [93, 94, 73, 72, 68] for details on similar stability results.

#### 7.2.4 Design of fault-detection and isolation filter

Fault tolerant control methods rely on the availability of a fall-back configuration that can maintain system stability and a control supervisor that will orchestrate the mode transition in a timely manner. The stability of the fault tolerant control system depends on accurate and timely FDI, stability of the closed-loop system under the fall-back configuration, and the location of the state in the state space upon FDI and reconfiguration. In this section we construct fault-detection and isolation (FDI) filters that will automatically identify the source of a failure in a timely manner. Utilizing synchronous state measurements,  $x_i(t)$  with  $i = 1, \dots, n_s$  and asynchronous state estimates,  $\hat{x}_i(t)$  with  $i = n + 1, \dots, n_s + n_a$  the following  $n_s + n_a$  filters are defined [70]:

$$\begin{aligned}
\dot{\tilde{x}}_i &= f_i(x_1, \dots, \tilde{x}_i, \dots, x_{n_s}, \hat{x}_{n_s+1}, \dots, \hat{x}_{n_s+n_a}, u_i, 0, \dots, 0), \quad i = 1, \dots, n_s \\
u_i &= h(x_1, \dots, \tilde{x}_i, \dots, x_{n_s}, \hat{x}_{n_s+1}, \dots, \hat{x}_{n_s+n_a}), \quad i = 1, \dots, n_s \\
\dot{\tilde{x}}_i &= f_i(x_1, \dots, x_{n_s}, \hat{x}_{n_s+1}, \dots, \tilde{x}_i, \dots, \hat{x}_{n_s+n_a}, u_i, 0, \dots, 0), \quad i = n_s + 1, \dots, n_s + n_a \\
u_i &= h(x_1, \dots, x_{n_s}, \hat{x}_{n_s+1}, \dots, \tilde{x}_i, \dots, \hat{x}_{n_s+n_a}), \quad i = n_s + 1, \dots, n_s + n_a
\end{aligned} \tag{7.5}$$

where  $\tilde{x}_i$  is filter output for the  $i^{th}$  state. The FDI filters are only initialized at  $t = 0$  such that  $\tilde{x}_i(0) = x_i(0)$ ,  $i = 1 \dots, n_s + n_a$ . For each state, the FDI residual can be defined as

$$r_i(t) = |x_i(t) - \tilde{x}_i(t)|, \quad i = 1 \dots, n_s + n_a.$$

The synchronous residuals  $r_i(t)$  with  $i = 1, \dots, n_s$  are computed continuously because  $x_i(t)$  with  $i = 1, \dots, n_s$  is known for all  $t$ . On the other hand, the asynchronous



residuals  $r_i(t)$  with  $i = n_s + 1, \dots, n_s + n_a$  are computed only at times  $t_{k,i}$  when a new asynchronous measurement of  $x_i(t)$  with  $i = n_s, \dots, n_s + n_a$  is received. These FDI filters operate by essentially predicting the fault-free evolution of each individual state, accounting for faults that enter the system when the predicted evolution of the state diverges from the measured evolution (see chapter 3).

The dynamics of both the asynchronous observers,  $\hat{x}_i$ , and the FDI filters,  $\tilde{x}_i$ , are identical to those of (7.1) when there are no disturbances or noise acting on the system. When the states are initialized as  $\hat{x}_i(0) = \tilde{x}_i(0) = x_i(0)$  both the observer and filter states will track the true process states. For the synchronous case when a fault,  $d_i$ , occurs, only the corresponding residual,  $r_i$ , will become nonzero. A rigorous proof of this FDI filter property can be found in chapter 3. In the case with asynchronous measurements, at least one  $r_i$  will become non-zero when a fault occurs. However, in the asynchronous case some faults (specifically, type two faults as defined below) cause the asynchronous observer  $\hat{x}_i$  to diverge from the true process state  $x_i$  between consecutive measurements, and any FDI filter states that are a function of  $\hat{x}_i$  will no longer accurately track the corresponding true process states. When such a fault occurs more than one residual value may become nonzero.

Continuous measurements for asynchronous states are not available, thus the FDI filters of (7.5) cannot always completely isolate all failures. We consider two classes of faults. Type one faults are faults that only affect states that are measured continuously; that is,  $d_j$  is a type one fault if

$$\frac{\partial f_i}{\partial d_j} = 0, \forall i = n_s + 1, \dots, n_s + n_a.$$

Type two faults affect at least one asynchronous state; that is,  $d_j$  is a type two fault

if there exists at least one  $i = n_s + 1, \dots, n_s + n_a$  such that

$$\frac{\partial f_i}{\partial d_j} \neq 0.$$

The FDI filter will detect and isolate a type one fault  $d_j$  because the asynchronous state observers will track the asynchronous states accurately (i.e., the effect of the fault  $d_j(t)$  on an asynchronous observer state is accounted for through the synchronous states, so  $d_j(t)$  is accounted for in the observer (7.3) and hence the FDI filter). A type two fault enters the system in the differential equation of a state that is sampled asynchronously. The effect of type two faults cannot be accounted for by the observer  $\hat{x}_i$ , and such a fault will cause  $\hat{x}_i$  to no longer track  $x_i$  and will eventually affect other coupled filter states as well. Strict isolation cannot take place for a type two fault. The FDI filter will detect and partially isolate disturbances in this case because the asynchronous state observers will diverge from the asynchronous states (i.e., the effect of the fault  $d_j(t)$  on an asynchronous observer state is unmeasured and unaccounted for, thus (7.3) does not track the disturbed state). In other words, if a type one fault occurs, then it can be detected and isolated. If a type two fault occurs, then this fault can be grouped to the subset of type two faults.

A fault is detected at time  $t_f$  if there exists a residual  $i$  such that  $r_i(t_f) > r_{i,max}$ , where  $r_{i,max}$  is an appropriate threshold chosen to account for process and sensor noise. In order to isolate the possible source of the fault, it is necessary to wait until the residuals of all the asynchronous state filters are updated after  $t_f$  to determine if the fault is type one or type two. The residual of each asynchronous state filter  $\tilde{x}_i$  is updated at time

$$t_i(t_f) = \min_k t_{k,i} \mid t_{k,i} > t_f.$$

If  $r_i(t_i(t_f)) \leq r_{i,max}$  with  $i = n_s + 1, \dots, n_s + n_a$ , then the fault occurred at time  $t_f$  is a type one fault and can be appropriately isolated. Otherwise, the fault belongs to

the set of type two faults.

Consider that a synchronous residual  $r_i$  indicates a fault at time  $t_f$ . In this case the fault could have two possible causes, a type one or type two fault. In order to determine the true cause of this fault, one has to wait for the complete set of asynchronous measurements to arrive after  $t_f$ . When all the asynchronous measurements arrive and if all the residuals of the asynchronous states are smaller than the threshold, then the fault can be attributed to a type one fault. If any asynchronous measurement arrives and the corresponding residual indicates a fault, then the fault is type two. Note that when an asynchronous residual indicates a fault, we can also conclude that the fault is type two. When the fault is type two it has been detected, and it is possible to narrow the fault source down to the set of faults that enter the differential equations of asynchronous states.

When the fault can be attributed to a type one fault and it has been detected and isolated, then automated fault tolerant control action can be initiated. For example, when a fault event that is due to a manipulated input failure (i.e., an actuator failure) is detected and isolated, fault tolerant control methods can be initiated as proposed in chapter 3. In general an FTC switching rule may be employed that orchestrates the re-configuration of the control system in the event of control system failure. This rule determines which of the backup control loops can be activated, in the event that the main control loop fails, in order to preserve closed-loop stability. Owing to the limitations imposed by input constraints on the stability region for each control configuration, switching from a malfunctioning configuration to a well-functioning, but randomly selected, backup configuration will not preserve closed-loop stability if the state of the system, at the time of failure, lies outside the stability region of the chosen backup configuration. In this case, stabilization using this configuration requires

more control action than is allowed by its constraints. This observation motivates the development of switching logic, which is to switch to the control configuration for which the closed-loop state resides within the stability region at the time of control failure. Without loss of generality, let the initial actuator configuration be  $k(0) = 1$  and let  $t_d$  be the time when this failure has been isolated, then the switching rule given by

$$k(t) = j \quad \forall t \geq t_d \text{ if } x(t_d) \in \Omega(u_j^{max}) \quad (7.6)$$

for some  $j \in \{2, 3, \dots, N\}$  guarantees closed-loop asymptotic stability.  $\Omega(u_j^{max})$  is the stability region for the  $j^{th}$  control configuration. The implementation of the above switching law requires monitoring the closed-loop state trajectory with respect to the stability regions associated with the various fall-back configurations. The reader may refer to [35] for application of FTC to a polyethylene reactor with constraints on the manipulated inputs. In this work we consider a control law without constraints on the manipulated inputs, and the primary control configuration with a faulty actuator will be deactivated in favor of a fully functional fall-back control configuration where the fall-back configuration can guarantee global stability of the closed-loop system. This integrated FDI/FTC reconfiguration allows for seamless fault-recovery in the event of an actuator failure. Section 7.3 demonstrates integrated FDI/FTC for the polyethylene reactor.

**Remark 7.2:** In the process model of (7.1), process and sensor noise are not explicitly taken into account. However, noise is indirectly accounted for in the FDI method below by means of appropriate tolerance thresholds in the decision criteria for fault detection and isolation; that is,  $r_{i,max}$ . The thresholds are generated on the basis of operating data and take into account both sensor and process noise, allowing for an appropriate FDI performance even if the process model and the measurements are corrupted by noise. To demonstrate this point, process and sensor noise are included

in the simulation study; see Section 7.3 for details.

## **7.3 Application to a polyethylene reactor with asynchronous measurements**

### **7.3.1 Process and measurement modeling**

The proposed model based asynchronous FDI and FTC method will be demonstrated using a model of an industrial gas phase polyethylene reactor. The feed to the reactor consists of ethylene, comonomer, hydrogen, inerts and catalyst. A recycle stream of unreacted gases flows from the top of the reactor and is cooled by passing through a water-cooled heat exchanger. Cooling rates in the heat exchanger are adjusted by mixing cold and warm water streams while maintaining a constant total cooling water flow rate through the heat exchanger. Mass balances on hydrogen and comonomer have not been considered in this study because hydrogen and comonomer have only mild effects on the reactor dynamics [60]. A mathematical model for this reactor has

the following form [15]:

$$\begin{aligned}
\frac{d[In]}{dt} &= \frac{1}{V_g} (F_{In} - \frac{[In]}{[M_1] + [In]} b_t) \\
\frac{d[M_1]}{dt} &= \frac{1}{V_g} (F_{M_1} - \frac{[M_1]}{[M_1] + [In]} b_t - R_{M_1}) + d_4 \\
\frac{dY_1}{dt} &= F_c a_c - k_{d_1} Y_1 - \frac{R_{M_1} M_{W_1} Y_1}{B_w} + d_2 \\
\frac{dY_2}{dt} &= F_c a_c - k_{d_2} Y_2 - \frac{R_{M_1} M_{W_1} Y_2}{B_w} + d_2 \\
\frac{dT}{dt} &= \frac{H_f + H_{g1} - H_{g0} - H_r - H_{pol}}{M_r C_{pr} + B_w C_{ppol}} + Q + d_1 \\
\frac{dT_{w_1}}{dt} &= \frac{F_w}{M_w} (T_{wi} - T_{w_1}) - \frac{UA}{M_w C_{pw}} (T_{w_1} - T_{g_1}) \\
\frac{dT_{g_1}}{dt} &= \frac{F_g}{M_g} (T - T_{g_1}) + \frac{UA}{M_g C_{pg}} (T_{w_1} - T_{g_1}) + d_3
\end{aligned} \tag{7.7}$$

where

$$\begin{aligned}
b_t &= V_p C_v \sqrt{([M_1] + [In]) RRT - P_v} \\
R_{M_1} &= [M_1] k_{p0} e^{-\frac{E_a}{R} (\frac{1}{T} - \frac{1}{T_f})} (Y_1 + Y_2) \\
C_{pg} &= \frac{[M_1]}{[M_1] + [In]} C_{pm1} + \frac{[In]}{[M_1] + [In]} C_{pIn} \\
H_f &= (F_{M_1} C_{pm1} + F_{In} C_{pIn}) (T_{feed} - T_f) \\
H_{g1} &= F_g (T_{g_1} - T_f) C_{pg} \\
H_{g0} &= (F_g + b_t) (T - T_f) C_{pg} \\
H_r &= H_{reac} M_{W_1} R_{M_1} \\
H_{pol} &= C_{ppol} (T - T_f) R_{M_1} M_{W_1}
\end{aligned} \tag{7.8}$$

The definitions for all the variables used in (7.7) and (7.8) are given in Table 7.1 and their values can be found in [15] (see also [35]). Under normal operating conditions, the open-loop system behaves in an oscillatory fashion (i.e., the system possesses an open-loop unstable steady-state surrounded by a stable limit cycle). The open-loop unstable steady-state around which the system will be controlled is

$$\begin{aligned}
[In]_{ss} &= 439.7 \frac{mol}{m^3} & [M_1]_{ss} &= 326.7 \frac{mol}{m^3} \\
Y_{1ss}, Y_{2ss} &= 3.835 mol & T_{ss} &= 356.2 K \\
T_{g1ss} &= 290.4 K & T_{w1ss} &= 294.4 K.
\end{aligned}$$

Note that with the given parameters, the dynamics of  $Y_1, Y_2$  are identical and will be reported in the results as a single combined state. In this example, we consider four possible faults,  $d_1, d_2, d_3$ , and  $d_4$  which represent a heat jacket fault, catalyst deactivation, a change in the recycle gas flow rate, and ethylene consumption, respectively. The primary manipulated input for these studies is the heat input,  $Q$ , and the fall-back manipulated input is the feed temperature,  $T_{feed}$ . In practice the temperature of the feed stream would be manipulated via a heat exchanger positioned on the feed line before it enters the process. A fall-back manipulated input is required to maintain desired system performance in the presence of failure in the primary control configuration.

Simulations have been carried out for several scenarios to demonstrate the effectiveness of the proposed FDI scheme in detecting and isolating the four faults  $d_1, d_2, d_3$ , and  $d_4$  in the presence of asynchronous measurements. The temperature measurements  $(T, T_{g1}, T_{w1})$  are all assumed to be available synchronously, while the concentration measurements  $([In], [M_1], Y)$  arrive at asynchronous intervals. In all the simulations, sensor measurement and process noise are included. The sensor measurement noise trajectory was generated using a sample time of ten seconds and a zero-mean normal distribution with standard deviation  $\sigma_M$ . The autoregressive process noise was generated discretely as  $w_k = \phi w_{k-1} + \xi_k$  where  $k = 0, 1, \dots$  is the discrete time step, with a sample time of ten seconds,  $\phi$  is the autoregressive coefficient and  $\xi_k$  is obtained at each sampling step using a zero-mean normal distribution with standard deviation  $\sigma_p$ . The autoregressive process noise is added to the right-hand side of the differential equations for each state and the sensor measurement noise is added to the measurements of each state. Sensor measurement noise and process noise are evaluated independently for each state variable. The process and sensor

measurement noise for  $Y_1$  and  $Y_2$  are taken to be equal. Table 7.2 provides the values of the noise parameters for each state of system (7.7). The length of time between consecutive asynchronous measurements is generated randomly based on a Poisson process. The time when the system will receive the next  $i^{th}$  asynchronous measurement is given by  $t_{k+1,i} = t_{k,i} + \Delta_a$  where  $\Delta_a = \ln(\xi)/W_a$  and  $\xi \in (0, 1)$  is a random variable chosen from a uniform probability distribution and  $W_a = 0.003 \text{ s}^{-1}$  is the mean rate of asynchronous sampling. There is an upper bound limiting the time between consecutive measurements such that  $\Delta_a \leq \Delta_M = 1200 \text{ s}$ . This value of  $\Delta_M$  is small enough to provide practical closed-loop stability around the desired equilibrium point for the polyethylene reactor of (7.7). An increasing sequence of measurement arrival times is generated independently for each asynchronously measured state.

### 7.3.2 Design of the asynchronous state observers

To perform FDI for the polyethylene reactor system we need to construct the asynchronous state observers of (7.3). The asynchronous state observers for this system



Table 7.1: Polyethylene reactor example process variables.

---

$a_c$	active site concentration of catalyst
$b_t$	overhead gas bleed
$B_w$	mass of polymer in the fluidized bed
$C_{pm1}$	specific heat capacity of ethylene
$C_v$	vent flow coefficient
$C_{pw}, C_{pIn}, C_{ppol}$	specific heat capacity of water, inert gas and polymer
$E_a$	activation energy
$F_c, F_g$	flow rate of catalyst and recycle gas
$F_{In}, F_{M1}, F_w$	flow rate of inert, ethylene and cooling water
$H_f, H_{g0}$	enthalpy of fresh feed stream, total gas outflow stream from reactor
$H_{g1}$	enthalpy of cooled recycle gas stream to reactor
$H_{pol}$	enthalpy of polymer
$H_r$	heat liberated by polymerization reaction
$H_{reac}$	heat of reaction
$[In]$	molar concentration of inerts in the gas phase
$k_{d1}, k_{d2}$	deactivation rate constant for catalyst site 1, 2
$k_{p0}$	pre-exponential factor for polymer propagation rate
$[M_1]$	molar concentration of ethylene in the gas phase
$M_g$	mass holdup of gas stream in heat exchanger
$M_r C_{pr}$	product of mass and heat capacity of reactor walls
$M_w$	mass holdup of cooling water in heat exchanger
$M_{W_1}$	molecular weight of monomer
$P_v$	pressure downstream of bleed vent
$Q$	Heat added/removed by heating jacket
$R, RR$	ideal gas constant, unit of $\frac{J}{mol \cdot K}, \frac{m^3 \cdot atm}{mol \cdot K}$
$T, T_f, T_{feed}$	reactor, reference, feed temperature
$T_{g1}, T_{w1}$	temperature of recycle gas, cooling water stream from exchanger
$T_{wi}$	inlet cooling water temperature to heat exchanger
$UA$	product of heat exchanger coefficient with area
$V_g$	volume of gas phase in the reactor
$V_p$	bleed stream valve position
$Y_1, Y_2$	moles of active site type 1, 2

---

Table 7.2: Polyethylene reactor noise parameters

	$\sigma_p$	$\sigma_m$	$\phi$
$[In]$	1E-4	5E-2	0
$[M_1]$	1E-4	5E-2	0.7
$Y$	1E-4	1E-2	0.7
$T$	5E-3	5E-2	0.7
$T_{g1}$	5E-3	5E-2	0.7
$T_{w1}$	5E-3	5E-2	0.7

have the following form:

$$\begin{aligned}
 \frac{d[\hat{In}]}{dt} &= \frac{1}{V_g} \left( F_{In} - \frac{[\hat{In}]}{[\hat{M}_1] + [\hat{In}]} \hat{b}_t \right) \\
 \frac{d[\hat{M}_1]}{dt} &= \frac{1}{V_g} \left( F_{M_1} - \frac{[\hat{M}_1]}{[\hat{M}_1] + [\hat{In}]} \hat{b}_t - \hat{R}_{M_1} \right) \\
 \frac{d\hat{Y}}{dt} &= F_c a_c - k_{d_1} \hat{Y} - \frac{\hat{R}_{M_1} M_{W_1} Y}{B_w} \\
 \hat{b}_t &= V_p C_v \sqrt{([\hat{M}_1] + [\hat{In}]) RRT(t) - P_v} \\
 \hat{R}_{M_1} &= [\hat{M}_1] k_{p0} e^{\frac{-E_a}{R} \left( \frac{1}{T(t)} - \frac{1}{T_f} \right)} (\hat{Y}) \\
 [\hat{In}](t_{k,[In]}) &= [In](t_{k,[In]}) \\
 [\hat{M}_1](t_{k,[M_1]}) &= [M_1](t_{k,[M_1]}) \\
 \hat{Y}(t_{k,Y}) &= Y(t_{k,Y})
 \end{aligned} \tag{7.9}$$

where  $[\hat{In}]$ ,  $[\hat{M}_1]$ , and  $\hat{Y}$  are the asynchronous observer states. Each asynchronous observer state is initialized each time new measurement information becomes available at the times  $t_{k,i}$ . The observer states provide estimates for the asynchronous states between consecutive measurements allowing the computation of control actions and FDI residuals at each time.

### 7.3.3 Design of the state feedback controller

The control objective is to stabilize the system at the open-loop unstable steady state. A nonlinear Lyapunov-based feedback controller that enforces asymptotic stability of the closed-loop system is synthesized using the method proposed in [88] (see also [24]). This is a single input controller that utilizes synchronous measurements as well as observer states generated by (7.9). System (7.7) belongs to the following class of non-linear systems

$$\dot{x}(t) = f(x(t)) + g_1(x(t))u_1(t) + g_2(x(t))u_2(t) + w(x(t))d(t) \quad (7.10)$$

where

$$x(t) = \begin{bmatrix} [In] - [In]_{ss} \\ [M_1] - [M_1]_{ss} \\ Y - Y_{ss} \\ T - T_{ss} \\ T_{g1} - T_{g1ss} \\ T_{w1} - T_{w1ss} \end{bmatrix}$$

and

$$u_1(t) = Q, \quad u_2(t) = T_{feed}.$$

Consider the quadratic control Lyapunov function  $V(x) = x^T P x$  where

$$P = 1 \times 10^{-2} \text{diag}[0.5 \ 0.5 \ 0.5 \ 1 \ 0.005 \ 0.005].$$

The values of the weighting matrix P are chosen to account for the different range of numerical values for each state. The following feedback laws [88] asymptotically stabilize the open-loop and possibly unstable steady-state of the nominal system (i.e.,  $d(t) \equiv 0$ )

$$h_i(x) = \begin{cases} -\frac{L_f V + \sqrt{L_f V^2 + L_{g_i} V^4}}{L_{g_i} V} & \text{if } L_{g_i} V \neq 0 \\ 0 & \text{if } L_{g_i} V = 0 \end{cases}, \quad i = 1, 2. \quad (7.11)$$

where  $L_f V$  and  $L_{g_i} V$  denote the Lie derivatives of the scalar function  $V$  with respect to the vectors fields  $f$  and  $g_i$  respectively.

In the simulations, the primary control configuration is given by

$$u_1(t) = h_1(\hat{x}(t))$$

and the fall-back control configuration is given by

$$u_2(t) = h_2(\hat{x}(t))$$

where

$$\hat{x}(t) = \begin{bmatrix} [\hat{I}n] - [In]_{ss} \\ [\hat{M}_1] - [M_1]_{ss} \\ \hat{Y} - Y_{ss} \\ T - T_{ss} \\ T_{g1} - T_{g1ss} \\ T_{w1} - T_{w1ss} \end{bmatrix}.$$

### 7.3.4 Design of FDI/FTC scheme

Fault-detection and isolation for the system in closed-loop with the primary configuration is accomplished by generating FDI filters from (7.5), and for the polyethylene system the FDI filters take the following form:

$$\begin{aligned} \frac{d[\tilde{I}n]}{dt} &= \frac{1}{V_g} (F_{In} - \frac{[\tilde{I}n]}{[\hat{M}_1] + [\tilde{I}n]} \tilde{b}_t^{[In]}) \\ \frac{d[\tilde{M}_1]}{dt} &= \frac{1}{V_g} (F_{M_1} - \frac{[\tilde{M}_1]}{[\hat{M}_1] + [\tilde{I}n]} \tilde{b}_t^{[M_1]} - \tilde{R}_{M_1}^{[M_1]}) \\ \frac{d\tilde{Y}}{dt} &= F_c a_c - k_{d_1} \tilde{Y} - \frac{\tilde{R}_{M_1}^Y M_{W_1} \tilde{Y}}{B_w} \\ \frac{d\tilde{T}}{dt} &= \frac{H_f + \tilde{H}_{g_1}^T - \tilde{H}_{g_0}^T - \tilde{H}_r^T - \tilde{H}_{pol}^T}{M_r C_{pr} + B_w C_{ppol}} + h_1(\hat{x}(t)) \\ \frac{d\tilde{T}_{w_1}}{dt} &= \frac{F_w}{M_w} (T_{wi} - \tilde{T}_{w_1}) - \frac{UA}{M_w C_{pw}} (\tilde{T}_{w_1} - T_{g_1}) \\ \frac{d\tilde{T}_{g_1}}{dt} &= \frac{F_g}{M_g} (T - \tilde{T}_{g_1}) + \frac{UA}{M_g \tilde{C}_{pg}} (T_{w_1} - \tilde{T}_{g_1}) \end{aligned} \tag{7.12}$$

where

$$\begin{aligned}
\tilde{b}_t^{[In]} &= V_p C_v \sqrt{([\hat{M}_1] + [\tilde{I}n])RRRT - P_v} \\
\tilde{b}_t^{[M_1]} &= V_p C_v \sqrt{([\tilde{M}_1] + [\hat{I}n])RRRT - P_v} \\
\tilde{b}_t^{[T]} &= V_p C_v \sqrt{([\hat{M}_1] + [\hat{I}n])RR\tilde{T} - P_v} \\
\tilde{R}_{M_1}^{[M_1]} &= [\tilde{M}_1]k_{p0}e^{-\frac{E_a}{R}(\frac{1}{T} - \frac{1}{T_f})}(\hat{Y}) \\
\tilde{R}_{M_1}^Y &= [\hat{M}_1]k_{p0}e^{-\frac{E_a}{R}(\frac{1}{T} - \frac{1}{T_f})}(\tilde{Y}) \\
\tilde{R}_{M_1}^T &= [\hat{M}_1]k_{p0}e^{-\frac{E_a}{R}(\frac{1}{T} - \frac{1}{T_f})}(\hat{Y}) \\
\tilde{C}_{pg} &= \frac{[\hat{M}_1]}{[\hat{M}_1] + [\hat{I}n]}C_{pm1} + \frac{[\hat{I}n]}{[\hat{M}_1] + [\hat{I}n]}C_{pIn} \\
\tilde{H}_{g_1}^T &= F_g(T_{g_1} - T_f)\tilde{C}_{pg} \\
\tilde{H}_{g_0}^T &= (F_g + \tilde{b}_t^T)(\tilde{T} - T_f)\tilde{C}_{pg} \\
\tilde{H}_r^T &= H_{reac}M_{W_1}\tilde{R}_{M_1}^T \\
\tilde{H}_{pol}^T &= C_{ppol}(\tilde{T} - T_f)\tilde{R}_{M_1}^T M_{W_1}
\end{aligned} \tag{7.13}$$

In addition, the FDI residuals take the following form:

$$\begin{aligned}
r_{[In]} &= |[In](t_k) - [\tilde{I}n](t_k)| \\
r_{[M_1]} &= |[M_1](t_k) - [\tilde{I}n](t_k)| \\
r_Y &= |Y(t_k) - \tilde{Y}(t_k)| \\
r_T &= |T - \tilde{T}| \\
r_{T_{g_1}} &= |T_{g_1} - \tilde{T}_{g_1}| \\
r_{T_{w_1}} &= |T_{w_1} - \tilde{T}_{w_1}|.
\end{aligned} \tag{7.14}$$

In the case with measurement and process noise, the residuals will be nonzero even without a failure event. This motivates the use of detection thresholds such that a fault is declared when a residual exceeds a specific threshold value,  $r_{i,max}$  (note that a different threshold value can be used for each residual, see Remark 7.2). This threshold value must be selected to avoid false alarms due to process and measurement noise, but it should also be sensitive enough (small enough) to detect faults in a timely manner so that efficient FTC action can be initiated. The threshold values used for each residual in the numerical simulations can be seen as the dashed lines in Figures 7.3, 7.7, 7.11, and 7.15.

If the fault can be isolated to  $d_1$  (i.e.,  $r_T$  exceeds  $r_{T,max}$  at  $t = t_f$ , while  $r_i(t_i(t_f)) \leq r_{i,max}$  with  $i = [In], [M_1], Y$ ), then one can invoke fault tolerant control methods to handle actuator failures by activation of a fall-back control configuration. In the simulation studies, it is assumed that a fall-back configuration, where the fall-back manipulated input  $u_2 = T_{feed}$ , is available. The control law of (7.11) enforces stability when the control actuator is functioning properly, thus switching to the operational fall-back configuration will guarantee stability in the case of failure of the primary control configuration,  $u_1 = Q$ .

### 7.3.5 Closed-loop process simulation results

This section consists of four simulation studies, each examining one of the faults  $d_1$ ,  $d_2$ ,  $d_3$ , or  $d_4$  as shown in (7.7). The first simulation considers a fault,  $d_1$ , on the heating jacket which is the primary manipulated input. In this case the simulation includes fault tolerant control that automatically reconfigures the plant so that the fall-back manipulated input,  $u_2 = T_{feed}$ , is activated to maintain stability. Specifically, the supervisory control element will deactivate the primary control configuration,  $u_1$  and activate the fall-back configuration  $u_2$  when  $r_T > r_{T,max}$  and  $r_i(t_i(t_f)) \leq r_{i,max}$  with  $i = [In], [M_1], Y$ . This specific fault signature corresponds to a type one fault that can be isolated to  $d_1$ . The reader may refer to [35] to obtain more information on FTC and reconfiguration rules for a polyethylene reactor with constraints on the manipulated inputs that give rise to stability regions. This work does not consider constraints on the manipulated inputs, hence, the fall-back configuration can guarantee stability from anywhere in the state space because the closed-loop system under the fall-back control configuration is globally asymptotically stable. The remaining simulation studies explore faults that disturb the system, but do not arise from actu-

ator failures. Since they are not caused by actuation component malfunctions these failures cannot be resolved simply by actuator reconfiguration. However, these simulations demonstrate quick detection and isolation in the presence of asynchronous measurements that enables the operator to take appropriate and focused action in a timely manner.

For the fault  $d_1$  a simulation study has been carried out to demonstrate the proposed asynchronous fault-detection and isolation and fault tolerant control method. The sequence of asynchronous measurements for this scenario is shown in Figure 7.1. This first simulation uses the primary control configuration in which  $Q$  is the manipulated input and has a fall-back configuration, in which  $T_{feed}$  is the manipulated input, available in case of a fault in  $d_1$ . A fault takes place where  $d_1 = 1 K/s$  at  $t = 0.5 hr$  representing a failure in the heating jacket,  $Q$ . At this time the synchronous states in Figure 7.2 all move away from the equilibrium point. Additionally, as asynchronous measurements become available, it is clear the asynchronous states also move away from the equilibrium point after the failure. It is unclear from the state information alone what caused this faulty behavior. However, if the FDI residuals in Figure 7.3 generated by (7.12) are examined, it is clear that the residual  $r_T$  that is associated with the manipulated input  $Q$ , violates its threshold at  $t_f = 0.5003 hr$ . The fault is detected upon this threshold violation. However, isolation cannot take place until one new measurement for each asynchronous state becomes available. At  $t = 0.5944 hr$  all three required asynchronous measurements have arrived, and the asynchronous residuals remain below their thresholds, hence  $r_i(t_i(t_f)) \leq r_{i,max}$  with  $i = [In], [M_1], Y$ . This signals that this is a type one fault that can be isolated to  $d_1$ . At this time, the system is reconfigured to the fall-back configuration where  $T_{feed}$  is the manipulated input, and the resulting state trajectory, shown as the dotted line

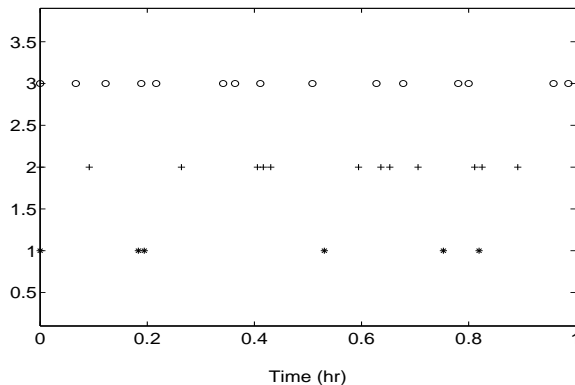


Figure 7.1: Asynchronous sampling times  $t_{k,[In]}$  (star),  $t_{k,[M_1]}$  (cross), and  $t_{k,Y}$  (circle) with a fault  $d_1$  at  $t = 0.5$  hr.

in Figure 7.2, moves back to the desired operating point. The manipulated input for this scenario can be seen in Figure 7.4 where the solid line is the manipulated input without detection and reconfiguration, and the dotted line represents the input after FDI and reconfiguration.

The second simulation demonstrates the proposed asynchronous model-based fault-detection and isolation method when a type two fault occurs. The sequence of asynchronous measurements for this scenario is shown in Figure 7.5. This simulation uses the primary control configuration in which  $Q$  is the manipulated input. A fault takes place where  $d_2 = -0.001$  mol/s at  $t = 0.5$  hr representing a catalyst deactivation event. After the failure, two synchronous states in Figure 7.6 move away from the equilibrium point. Additionally, as asynchronous measurements become available it can be seen that asynchronous states also move away from the equilibrium point after the failure. It is unclear from the state information alone what caused this faulty behavior. However, if the FDI residuals in Figure 7.7 generated by (7.14) are examined, it is clear that the residuals  $r_{[M_1]}$ ,  $r_Y$ , and  $r_T$  violate their thresholds. The fault is detected upon the first threshold violation ( $r_Y$  at  $t = 0.5333$ ). When the residual



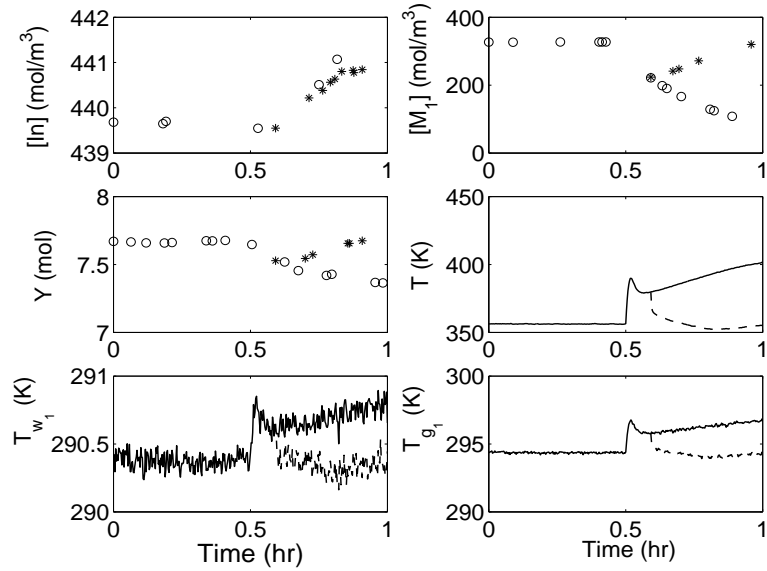


Figure 7.2: State trajectories of the closed-loop system without fault-tolerant control (circle/solid) and with appropriate fault-detection and isolation and fault-tolerant control where the fall-back control configuration is activated (star/dotted) with a fault  $d_1$  at  $t = 0.5$  hr.

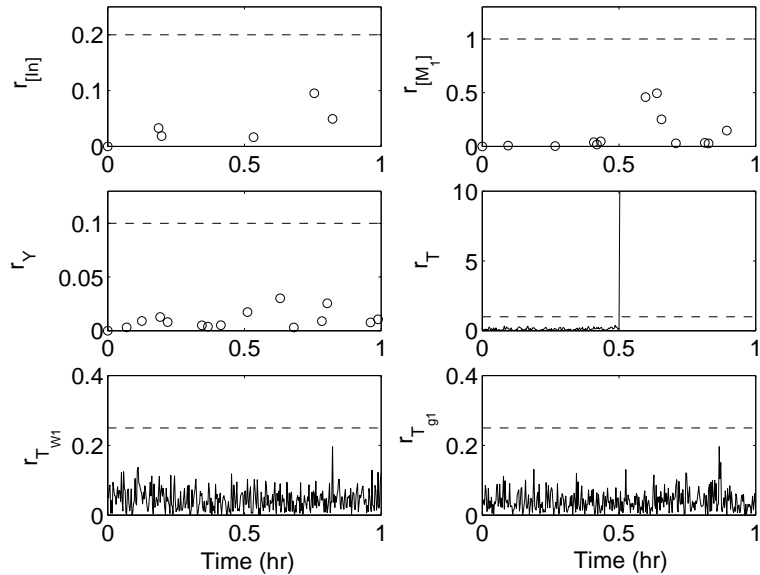


Figure 7.3: Fault-detection and isolation residuals for the closed-loop system with a fault  $d_1$  at  $t = 0.5$  hr. The fault is detected immediately, but isolation occurs at  $t = 0.59$  hr when all three asynchronous states have reported a residual below their detection threshold. This signals a type one fault, and we can isolate the source of this fault as  $d_1$

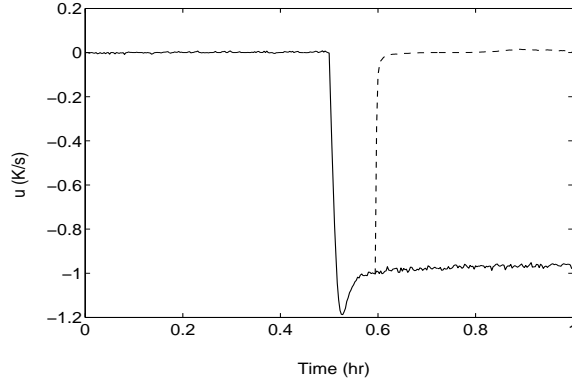


Figure 7.4: Manipulated input for the closed-loop system without fault-tolerant control (solid) and with appropriate fault-tolerant control where the fall-back control configuration is activated (dotted) with a fault  $d_1$  at  $t = 0.5 \text{ hr}$ .

associated with  $Y$  exceeds the threshold this signals that the fault is type two and entered the system in the differential equation of an asynchronous state. When the fault is type two it cannot be isolated. However, such a fault can be grouped in the subset of faults that enter into the differential equation of an asynchronous state, (i.e., the group of type two faults, specifically,  $d_2$  or  $d_4$ ). At this time, the system operator can utilize the above partial isolation to examine the plant and determine the exact source of the failure. The manipulated input for this scenario can be seen in Figure 7.8.

The third simulation study examines FDI in the presence of a type one fault,  $d_3$ , representing a change in the recycle gas flow rate. The sequence of asynchronous measurements for this scenario is shown in Figure 7.9. This simulation study uses the primary control configuration in which  $Q$  is the manipulate input, and a fault takes place where  $d_3 = 300 \text{ K/s}$  at  $t = 0.5 \text{ hr}$ . At this time the synchronous states in Figure 7.10 all move away from the equilibrium point. Additionally, as asynchronous measurements become available it is observed that the asynchronous states also move away from the equilibrium point after the failure. It is unclear from the state in-

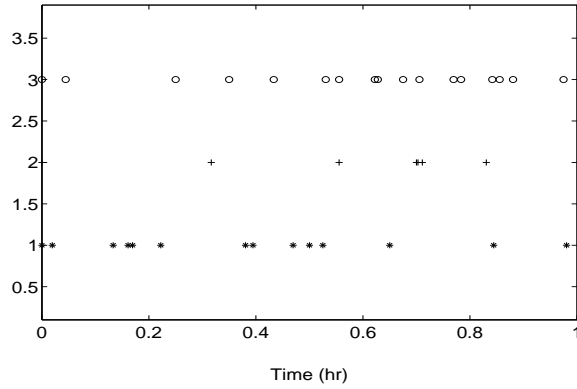


Figure 7.5: Asynchronous sampling times  $t_{k,[In]}$  (star),  $t_{k,[M_1]}$  (cross), and  $t_{k,Y}$  (circle) with a fault  $d_2$  at  $t = 0.5 \text{ hr}$ .

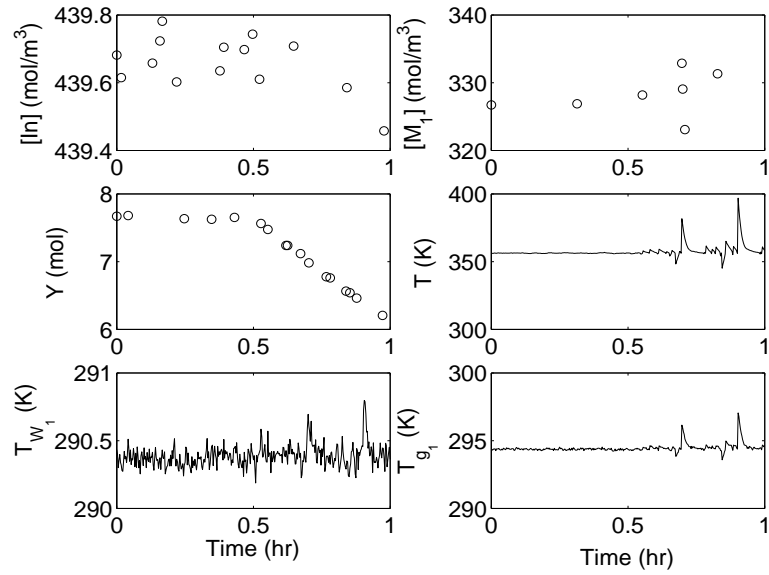


Figure 7.6: State trajectories of the closed-loop system with a fault  $d_2$  at  $t = 0.5 \text{ hr}$ .

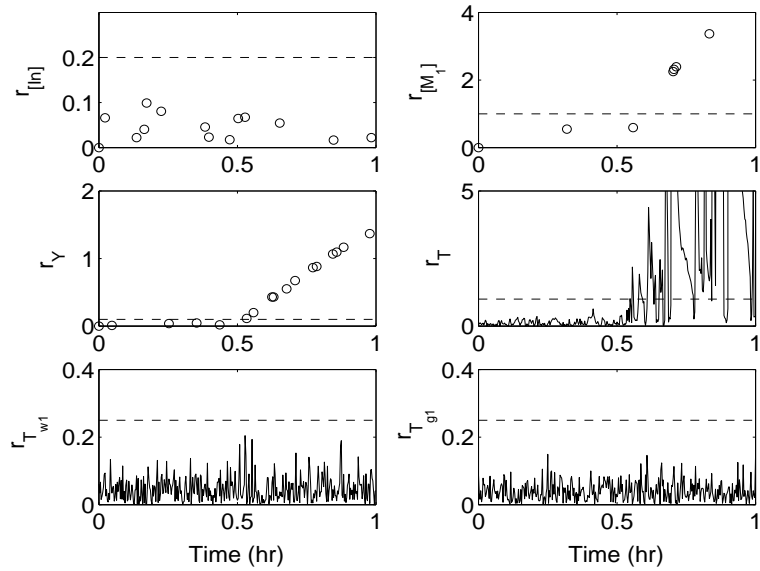


Figure 7.7: Fault-detection and isolation residuals for the closed-loop system with a fault  $d_2$  at  $t = 0.5$  hr. The fault is detected when residual for  $Y$  exceeds the threshold. Subsequently,  $T$  and  $[M_1]$  exceed their thresholds. When any asynchronous residual violates the threshold this indicates that the fault is in the set of type two faults;  $d_2$  or  $d_4$ .

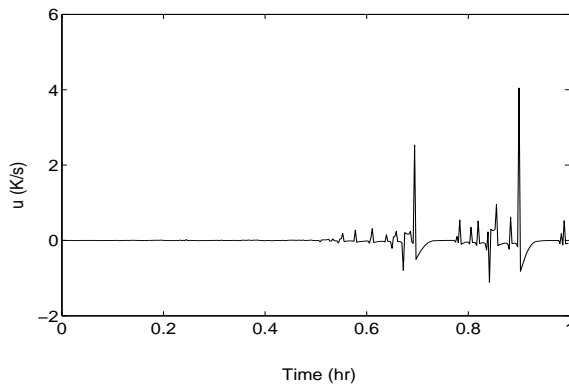


Figure 7.8: Manipulated input for the closed-loop system with a fault  $d_2$  at  $t = 0.5$  hr.

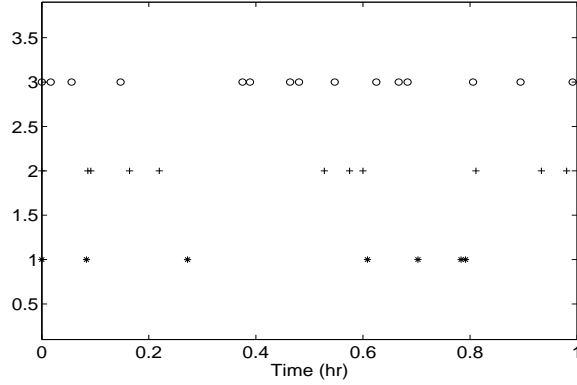


Figure 7.9: Asynchronous sampling times  $t_{k,[I_n]}$  (star),  $t_{k,[M_1]}$  (cross), and  $t_{k,Y}$  (circle) with a fault  $d_3$  at  $t = 0.5 \text{ hr}$ .

formation alone what caused this faulty behavior. However, if the FDI residuals in Figure 7.11 generated by (7.12), (7.13), and (7.14) are examined, the residual associated with  $T_{g1}$ , violates its threshold at  $t = 0.5003 \text{ hr}$ . The fault is detected upon this threshold violation. However, isolation cannot take place until one new measurement for each asynchronous state becomes available. At  $t = 0.6086 \text{ hr}$  all three required asynchronous measurements have become available, and the residuals signal a type one fault, allowing the isolation of the fault to  $d_3$ . The manipulated input for this scenario can be seen in Figure 7.12.

The final simulation study demonstrates the proposed asynchronous model-based fault-detection and isolation method when a type two fault occurs. The sequence of asynchronous measurements for this scenario is shown in Figure 7.13. This simulation uses the primary control configuration in which  $Q$  is the manipulated input. A fault takes place where  $d_4 = -0.2 \text{ mol/s}$  at  $t = 0.5 \text{ hr}$  representing unexpected monomer consumption. After the failure the synchronous states in Figure 7.14 diverge from their desired values. Additionally, as asynchronous measurements become available it can be seen that asynchronous states also diverge after the failure. It is unclear from

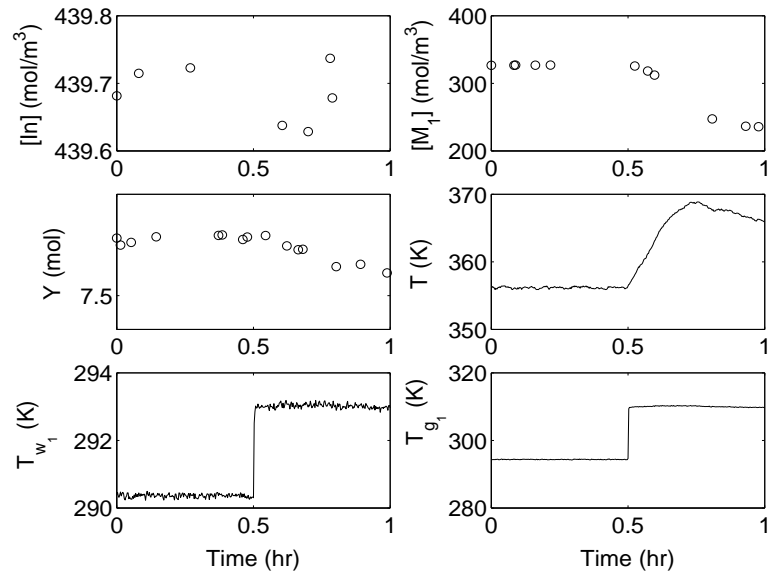


Figure 7.10: State trajectories of the closed-loop system with a fault  $d_3$  at  $t = 0.5$  hr.

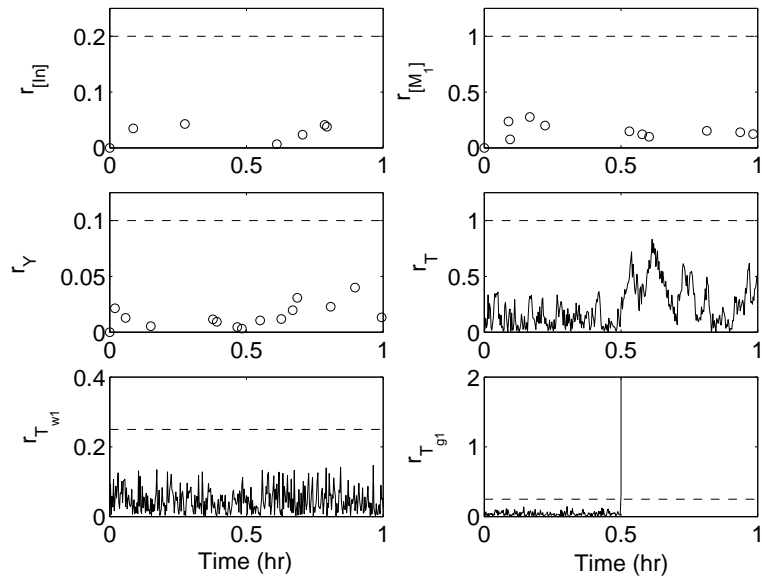


Figure 7.11: Fault-detection and isolation residuals for the closed-loop system with a fault  $d_3$  at  $t = 0.5$  hr. A fault is detected immediately when residual for  $T_{g1}$  exceeds the threshold. Subsequently, none of the asynchronous residuals exceed their thresholds, indicating that the fault source can be isolated as  $d_3$ .

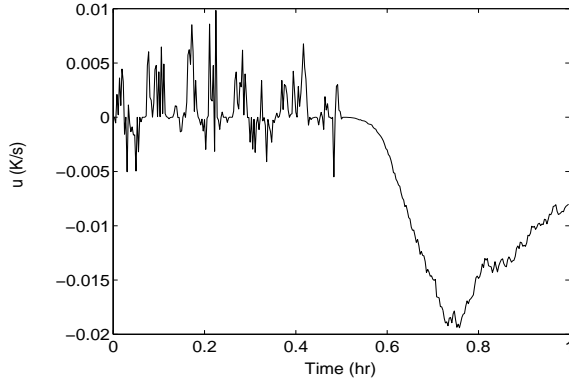


Figure 7.12: Manipulated input for the closed-loop system with a fault  $d_3$  at  $t = 0.5$  hr.

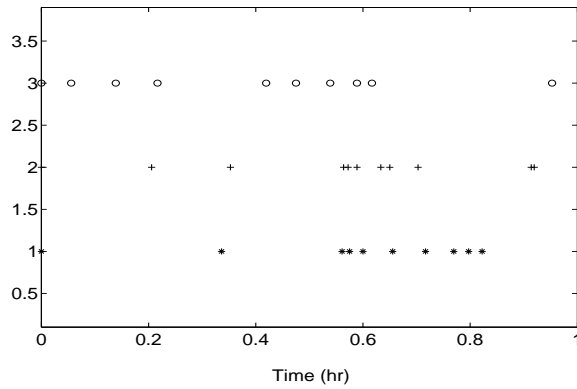


Figure 7.13: Asynchronous sampling times  $t_{k,[In]}$  (star),  $t_{k,[M_1]}$  (cross), and  $t_{k,Y}$  (circle) with a fault  $d_4$  at  $t = 0.5$  hr.

the state information alone what caused this faulty behavior. However, if the FDI residuals in Figure 7.15 generated by (7.12) are examined, the residuals  $r_{[In]}$ ,  $r_{[M_1]}$ ,  $r_T$ , and  $r_{T_{g1}}$  violate their thresholds. The fault is detected upon the first threshold violation ( $r_{[M_1]}$  at  $t = .05667$  hr). When the residual  $r_{[M_1]}$  exceeds the threshold this signals that a type two fault has occurred. When a type two fault occurs it cannot be isolated. As in the second simulation, such a fault can be grouped in the subset of type two faults  $d_2$  or  $d_4$ . At this time, the system operator can utilize the partial isolation to examine the plant and determine the exact source of the failure. The manipulated input for this scenario can be seen in Figure 7.16.

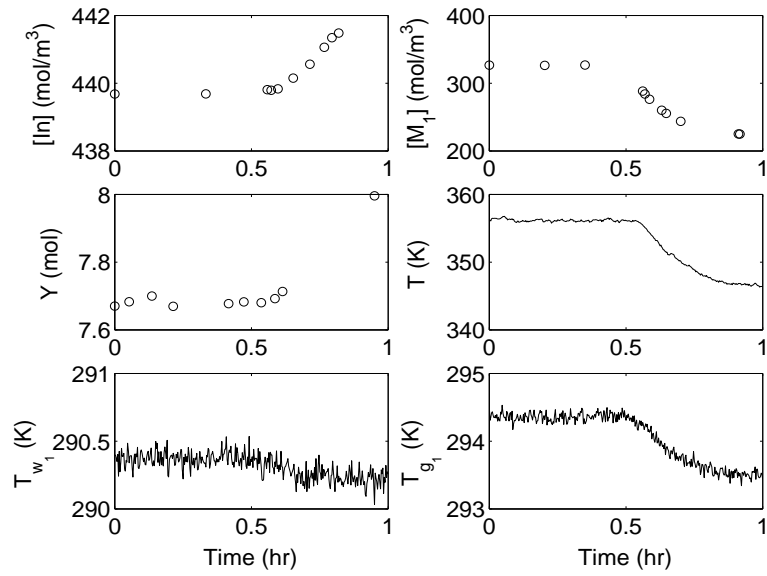


Figure 7.14: State trajectories of the closed-loop system with a fault  $d_4$  at  $t = 0.5$  hr.

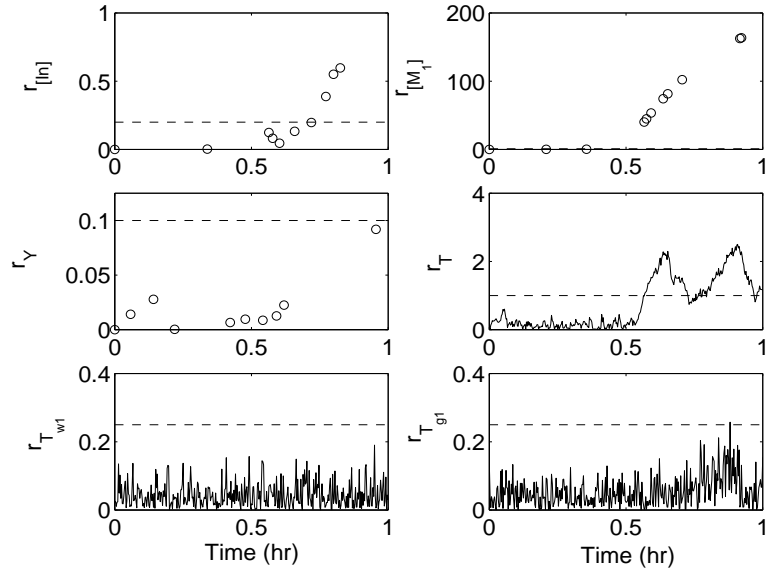


Figure 7.15: Fault-detection and isolation residuals for the closed-loop system with a fault  $d_4$  at  $t = 0.5$  hr. The fault is detected when residual for  $[M_1]$  exceeds the threshold. Subsequently,  $T$  and  $[In]$  exceed their thresholds. When any asynchronous residual violates the threshold this indicates the fault is in the set of type two faults;  $d_2$  or  $d_4$ .



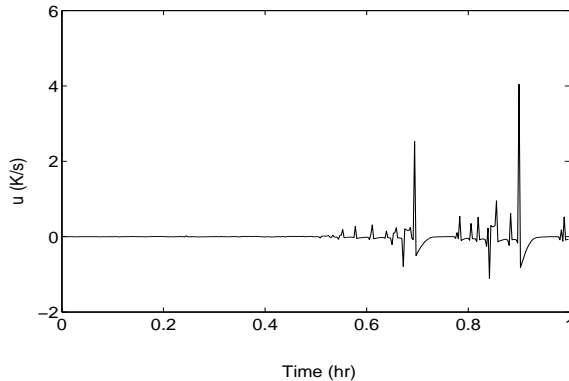


Figure 7.16: Manipulated input for the closed-loop system with a fault  $d_4$  at  $t = 0.5$  hr.

## 7.4 Conclusions

This work addressed the problem of fault-detection and isolation and fault-tolerant control when several process measurements are not available synchronously. First, a fault-detection and isolation scheme that employs model-based techniques was proposed that allowed for the isolation of faults. This scheme employed model-based FDI filters in addition to observers that estimate the fault-free evolution of asynchronously measured states during times when they are unmeasured. Specifically, the proposed FDI scheme provides detection and isolation for a type one fault where the fault enters into the differential equation of only synchronously measured states, and grouping of type two faults that enter into the differential equation of any asynchronously measured state. The detection occurs shortly after a fault takes place, and the isolation, limited by the arrival of asynchronous measurements, occurs once all of the asynchronous measurements become available. Once the FDI methodology provided the system supervisor with a fault diagnosis, the supervisor took appropriate action to seamlessly reconfigure the system to an alternative control configuration that enforces the desired operation. We presented applications of the proposed asynchronous FDI and FTC framework to a polyethylene reactor simulation.

## Chapter 8

# Conclusions

This dissertation developed a general and practical framework for the design of nonlinear automated fault tolerant control systems that seamlessly integrate the tasks of nonlinear fault-detection and isolation and control system reconfiguration for fault handling. Working with general nonlinear dynamic models of chemical processes, we designed nonlinear dynamic filters that allow for timely detection and isolation of actuator/control system faults using limited plant measurements. The key idea is to design a fault detection and isolation scheme for nonlinear process systems that decouples the effect of a fault on all process variables except one. This allows fault detection and isolation for nonlinear chemical processes even with highly coupled variables. The nonlinear dynamic filters were coupled with suitable control system reconfiguration strategies which achieve quick fault recovery and guarantee closed-loop system stability. In addition, these fault-tolerant control methods deal explicitly with the practical issues of limited control actuator capacity, model uncertainty and disturbances, measurement noise and sensor faults. We presented applications of the proposed fault-tolerant control system design framework to a number of process systems.

It is important to point out several major conclusions that can be drawn from this dissertation. First and foremost this work develops theory and techniques for non-linear system applications. The focus on non-linear system is in itself a major area of contribution because it is an areas that has not been well-developed even though virtually all process systems with integrated control are non-linear. The consideration of guaranteed stability as a key additional constraint is substantially enriched as a result. One major contribution is the development of fault detection and isolation filters for a general class of nonlinear systems. Previous work has focused extensively on linear filters. The extension to nonlinear systems required considerable expansion of the theory and integration with other components in the overall FTC system. The proposed fault detection and isolation framework is general enough to define the class of nonlinear systems for which isolation is possible, thus the framework can also be used in the design of control systems for which full fault isolation is achievable. For example, assumption 3.1 can be used by the control engineer to help determine the states that are important to measure, and the variables that are best to use for manipulated inputs in order to enable full isolation of actuator faults. In addition, the proposed fault detection and isolation methodologies are useful in application to existing chemical process systems, as many process systems are naturally structured to allow actuator fault detection and isolation. Chapter 7 provides results for process systems subject to asynchronous measurements that can assist the engineer in selecting appropriate manipulated inputs, asynchronous sensors, and synchronous sensors that will provide the desired level of isolability. Another contribution is the development of a fault-tolerant control structure in the presence of sensor data losses. This fault-tolerant control structure allows the control engineer to determine acceptable level of data loss while still providing a guarantee of stability. The final major contribution is the seamless integration of newly developed fault detection and isolation

methods with fault tolerant control methods. These fault tolerant control methods consist of constrained nonlinear controllers, which provide explicit stability regions for each fall-back closed-loop configuration, in conjunction with a supervisory control system that can make timely decisions about reconfiguration in the event of a fault. The integration of fault detection and isolation and fault-tolerant control enables the automated supervisory control system to react to abnormal situations quickly and optimally, thus freeing up the human plant operator to focus on strategic plant management.

Specifically, chapters 2 and 3 focused on the development of integrated fault tolerant control methodologies for general process systems. These methodologies were developed in a general manner so they may be applied to a wide variety of chemical processing plants. Specifically, chapter 2 focused on fault detection for single-input process systems with manipulated input constraints, and chapter 3 extended these results to include multi-input nonlinear process systems with constraints on the manipulated inputs. Necessary conditions for the design of state- and output-feedback fault detection and isolation filters were derived. Filters were designed that essentially capture the difference between fault-free and observed (or estimated) evolution of the system states to detect and isolate faults in the control actuators. Reconfiguration rules were devised to identify the appropriate backup control configuration accounting for the faulty actuator and constraints. The implementation of the fault detection and isolation filters and reconfiguration strategy as well as robustness with respect to plant-model mismatch, measurement sampling and delay and measurement noise were demonstrated via chemical process examples.

Chapter 4 considered the problem of designing a fault-tolerant controller for nonlinear process systems subject to constraints and sensor data losses. Having identified

candidate control configurations for a given system, we first explicitly characterized the stability properties that is, the set of initial conditions starting from where closed-loop stabilization under continuous availability of measurements is guaranteed as well as derived a bound on the maximum allowable data loss rate which preserves closed-loop stability. This characterization was utilized in designing a reconfiguration logic that was shown to achieve practical stability in the presence of sensor data losses. The application of the proposed method was illustrated using a chemical process example and was also applied to a polyethylene reactor.

The contributions of chapters 5 and 6 include the development of a dynamic model for high recovery RO desalination. This model describes the spatial and temporal behavior of a high recovery RO desalination process. Additionally, nonlinear control techniques that include feed-forward/feedback control for disturbance rejection and FDIFTC were applied to this dynamic model accounting for practical issues such as noisy/sampled measurements, large time varying disturbances, and actuator failures. The feed-forward component in the controller was able to compensate for large time varying disturbances in the feed concentration. FDIFTC methods were applied in simulation examples in order to detect actuator faults and switch appropriately to fall-back configurations avoiding undesired RO system operation.

Finally, chapter 7 addressed the problem of fault detection and isolation and fault-tolerant control when several process measurements are not available synchronously. First, a fault detection and isolation scheme that employs model-based techniques was proposed that allowed for the isolation of faults. The detection occurs shortly after a fault takes place, and the isolation, limited by the arrival of asynchronous measurements, occurs once all of the asynchronous measurements become available. Once the FDI methodology provided the system supervisor with a fault diagnosis,

the supervisor took appropriate action to seamlessly reconfigure the system to an alternative control configuration that enforces the desired operation. We presented applications of the proposed asynchronous FDI and FTC framework to a polyethylene reactor simulation.

There are several opportunities for future work in the area of nonlinear fault detection and isolation and fault tolerant control to follow up on the research direction developed in this dissertation. The implementation of data based fault detection and isolation schemes on processes with asynchronous measurements is a natural extension of the work in chapter 7. Augmenting chemical processes, with existing synchronous closed-loop control in place, by addition of asynchronous sensors/actuators could be studied. Such augmentation would improve fault isolability and closed-loop performance. The reverse osmosis models proposed in chapters 5 and 6 could be verified and improved via laboratory scale experiments, and fault detection and isolation and fault-tolerant control methods could be applied to experimental reverse osmosis systems. Based on the work of chapter 7, one could consider the case of output feedback where some process variables are measured synchronously, some are measured asynchronously, and some are not measured at all. In this case the unmeasured states must be reconstructed through state estimation. Furthermore, future work in this area could include the application of the results from chapter 3, 4, and 7 to a large scale chemical processing plant based on an industrial example, such as a highly interconnected refinery, to explore unique issues for model-based FDI and FTC that arise when very large numbers of states and manipulated inputs are involved. Eventually this could lead to the verification of the proposed frameworks on a large scale based on data from chemical processes that are currently in operation.

# Bibliography

- [1] A. Abbas. Model predictive control of a reverse osmosis desalination unit. *Desalination*, 194:268–280, 2006.
- [2] I. M. Alatiqi, A. H. Ghabris, and S. Ebrahim. System identification and control of reverse osmosis desalination. *Desalination*, 75:119–140, 1989.
- [3] H. B. Aradhye, B. R. Bakshi, J. F. Davis, and S. C. Ahalt. Clustering in wavelet domain: A multiresolution art network for anomaly detection. *AIChE J.*, 50:2455–2466, 2004.
- [4] H. B. Aradhye, B. R. Bakshi, R. A. Strauss, and J. F. Davis. Multiscale SPC using wavelets: Theoretical analysis and properties. *AIChE J.*, 49:939–958, 2003.
- [5] J. Z. Assef, J. C. Watters, P. B. Deshpande, and I. M. Alatiqi. Advanced control of a reverse osmosis desalination unit. *J. Proc. Contr.*, 7:283–289, 1997.
- [6] J. Bao, W. Z. Zhang, and P. L. Lee. Decentralized fault-tolerant control system design for unstable processes. *Chem. Eng. Sci.*, 58:5045–5054, 2003.
- [7] W. B. Bequette. Nonlinear control of chemical processes: A review. *Ind. & Eng. Chem. Res.*, 30:1391–1413, 1991.

- [8] S.P. Bhat and D.S Bernstein. Finite-time stability of continuous autonomous systems. *SIAM Journal on Control and Optimization*, 38:751–766, 2000.
- [9] R. B. Bird, W. E. Stewart, and E. N. Lightfoot. *Transport Phenomena, Second Edition*. Wiley, 2002.
- [10] A. C. Burden, P. B. Deshpande, and J. C. Watters. Advanced control of a B-9 Permasep permeator desalination pilot plant. *Desalination*, 133:271–283, 2001.
- [11] S. C. Chapra and R. P. Canale. *Numerical Methods for Engineers, Fourth Edition*. McGraw Hill, 2002.
- [12] P. D. Christofides. Robust output feedback control of nonlinear singularly perturbed systems. *Automatica*, 36:45–52, 2000.
- [13] P. D. Christofides and N. H. El-Farra. *Control of Nonlinear and Hybrid Process Systems: Designs for Uncertainty, Constraints and Time-Delays*. Springer-Verlag, Berlin, Germany, 2005.
- [14] P. D. Christofides and A. R. Teel. Singular perturbations and input-to-state stability. *IEEE Trans. Autom. Contr.*, 41:1645–1650, 1996.
- [15] S. A. Dadebo, M. L. Bell, P. J. McLellan, and K. B. McAuley. Temperature control of industrial gas phase polyethylene reactors. *Journal of Process Control*, 7:83–95, 1997.
- [16] J. F. Davis, M. L. Piovoso, K. Kosanovich, and B. Bakshi. Process data analysis and interpretation. *Advances in Chemical Engineering*, 25:1–103, 1999.
- [17] R. A. DeCarlo, M. S. Branicky, S. Pettersson, and B. Lennartson. Perspectives and results on the stability and stabilizability of hybrid systems. *Proceedings of the IEEE*, 88:1069–1082, 2000.



- [18] C. DePersis and A. Isidori. A geometric approach to nonlinear fault detection and isolation. *IEEE Trans. Automat. Contr.*, 46:853–865, 2001.
- [19] C. DePersis and A. Isidori. On the design of fault detection filters with game-theoretic-optimal sensitivity. *Int. J. Rob. & Non. Contr.*, 12:729–747, 2002.
- [20] J. M. Dickson, J. Spencer, and M. L. Costa. Dilute single and mixed solute systems in a spiral wound reverse osmosis module Part I: Theoretical model development. *Desalination*, 89:63–88, 1992.
- [21] S. Djurjebic and N. Kazantzi. A new Lyapunov design approach for nonlinear systems based on Zubov’s method. *Automatica*, 38:1999–2007, 2002.
- [22] R. Dunia and S. J. Qin. Subspace approach to multidimensional fault identification and reconstruction. *AIChE J.*, 44:1813–1831, 1998.
- [23] R. Dunia, S. J. Qin, T. F. Edgar, and T. J. McAvoy. Identification of faulty sensors using principal component analysis. *AIChE J.*, 42:2797–2812, 1996.
- [24] N. H. El-Farra and P. D. Christofides. Integrating robustness, optimality and constraints in control of nonlinear processes. *Chem. Eng. Sci.*, 56:1841–1868, 2001.
- [25] N. H. El-Farra and P. D. Christofides. Bounded robust control of constrained multivariable nonlinear processes. *Chem. Eng. Sci.*, 58:3025–3047, 2003.
- [26] N. H. El-Farra and P. D. Christofides. Coordinated feedback and switching for control of hybrid nonlinear processes. *AIChE J.*, 49:2079–2098, 2003.
- [27] N. H. El-Farra, A. Gani, and P. D. Christofides. Fault-tolerant control of process systems using communication networks. *AIChE J.*, 51:1665–1682, 2005.

- [28] N. H. El-Farra, P. Mhaskar, and P. D. Christofides. Hybrid predictive control of nonlinear systems: Method and applications to chemical processes. *Inter. J. Rob. & Non. Contr.*, 14:199–225, 2004.
- [29] N. H. El-Farra, P. Mhaskar, and P. D. Christofides. Output feedback control of switched nonlinear systems using multiple Lyapunov functions. *Sys. & Contr. Lett.*, 54:1163–1182, 2005.
- [30] R. Findeisen, L. Imsland, F. Allgower, and B. A. Foss. State and output feedback nonlinear model predictive control: An overview. *Eur. J. Contr.*, 9:190–206, 2003.
- [31] P. M. Frank. Fault diagnosis in dynamic systems using analytical and knowledge-based redundancy – a survey and some new results. *Automatica*, 26:459–474, 1990.
- [32] P. M. Frank and X. Ding. Survey of robust residual generation and evaluation methods in observer-based fault detection systems. *J. Proc. Contr.*, 7:403–424, 1997.
- [33] R. A. Freeman and P. V. Kokotovic. *Robust Nonlinear Control Design: State-Space and Lyapunov Techniques*. Birkhauser, Boston, 1996.
- [34] A. Gambier and E. Badreddin. Application of hybrid modeling and control techniques to desalination plants. *Desalination*, 152:175–184, 2002.
- [35] A. Gani, P. Mhaskar, and P. D. Christofides. Fault-tolerant control of a polyethylene reactor. *Journal of Process Control*, 17:439–451, 2007.
- [36] C. E. Garcia, D. M. Prett, and M. Morari. Model predictive control - theory and practice - a survey. *Automatica*, 25:335–348, 1989.

- [37] E. A. Garcia and P. M. Frank. Deterministic nonlinear observer-based approaches to fault diagnosis: A survey. *Contr. Eng. Prac.*, 5:663–670, 1997.
- [38] V. Garcia-Onorio and B. E. Ydstie. Distributed, asynchronous and hybrid simulation of process networks using recording controllers. *Inter. J. Rob. & Non. Contr.*, 14:227–248, 2004.
- [39] C. J. Geankoplis. *Transport Processes and Separation Process Principles, Fourth Edition*. Prentice Hall, 2003.
- [40] I. E. Grossmann, S. A. van den Heever, and I. Harjukoski. Discrete optimization methods and their role in the integration of planning and scheduling. In *Proceedings of 6th International Conference on Chemical Process Control*, pages 124–152, Tucson, AZ, 2001.
- [41] T. J. Harris, F. Boudreau, and J. F. MacGregor. Performance assessment of multivariable feedback controllers. *Automatica*, 32:1505–1518, 1996.
- [42] A. Hassibi, S. P. Boyd, and J. P. How. Control of asynchronous dynamical systems with rate constraints on events. In *Proceedings of 38th IEEE Conference on Decision and Control*, pages 1345–1351, Phoenix, AZ, 1999.
- [43] M. A. Henson and D. E. Seborg. *Nonlinear Process Control*. Prentice-Hall, Englewood Cliffs, NJ, 1997.
- [44] D. Herold and A. Neskakis. A small PV-driven reverse osmosis desalination plant on the island of Gran Canaria. *Desalination*, 137:285–292, 2001.
- [45] J. P. Hespanha and A. S. Morse. Stability of switched systems with average dwell time. In *Proceedings of 38th IEEE Conference on Decision and Control*, pages 2655–2660, Phoenix, AZ, 1999.

- [46] N. Kapoor and P. Daoutidis. Stabilization of systems with input constraints. *Int. J. Contr.*, 34:653–675, 1998.
- [47] N. Kazantzis and C. Kravaris. Nonlinear observer design using lyapunov’s auxiliary theorem. *Syst. & Contr. Lett.*, 34:241–247, 1999.
- [48] N. Kazantzis, C. Kravaris, and R. A. Wright. Nonlinear observer design for process monitoring. *Ind. & Eng. Chem. Res.*, 39:408–419, 2000.
- [49] H. K. Khalil. *Nonlinear Systems*. Macmillan Publishing Company, New York, second edition, 1996.
- [50] H. K. Khalil and F. Esfandiari. Semiglobal stabilization of a class of nonlinear systems using output feedback. *IEEE Trans. Automat. Contr.*, 38:1412–1415, 1993.
- [51] P. V. Kokotovic and M. Arca. Constructive nonlinear control: a historical perspective. *Automatica*, 37:637–662, 2001.
- [52] J. V. Kresta, J. F. Macgregor, and T. E. Marlin. Multivariate statistical monitoring of process operating performance. *Can. J. Chem. Eng.*, 69:35–47, 1991.
- [53] N. Krstic, I. Kanellakopoulos, and P. Kokotovic. *Nonlinear and Adaptive Control Design*. Wiley, New York, first edition, 1995.
- [54] Y. Lin and E. D. Sontag. A universal formula for stabilization with bounded controls. *Sys. & Contr. Lett.*, 16:393–397, 1991.
- [55] C. K. Liu, Jae-Woo Park, R. Migita, and G. Qin. Experiments of a prototype wind-driven reverse osmosis desalination system with feedback control. *Desalination*, 150:277–287, 2002.

- [56] Y. Lua, Y. Hua, X. Zhang, L. Wu, and Q. Liu. Optimum design of reverse osmosis system under different feed concentration and product specification. *J. Membr. Sci.*, 287:219–229, 2007.
- [57] N. A. Mahmoud and H. K. Khalil. Asymptotic regulation of minimum phase nonlinear systems using output feedback. *IEEE Trans. Automat. Contr.*, 41:1402–1412, 1996.
- [58] M. Massoumnia, G. C. Verghese, and A. S. Wilsky. Failure detection and identification. *IEEE Trans. Automat. Contr.*, 34:316–321, 1989.
- [59] D. Q. Mayne, J. B. Rawlings, C. V. Rao, and P. O. M. Scokaert. Constrained model predictive control: Stability and optimality. *Automatica*, 36:789–814, 2000.
- [60] K. B. McAuley, D. A. Macdonald, and P. J. McLellan. Effects of operating conditions on stability of gas-phase polyethylene reactors. *AIChE Journal*, 41:868–879, 1995.
- [61] N. Mehranbod, M. Soroush, and C. Panjapornpon. A method of sensor fault detection and identification. *J. Proc. Contr.*, 15:321–339, 2005.
- [62] P. Mhaskar, N. H. El-Farra, and P. D. Christofides. Hybrid predictive control of process systems. *AIChE J.*, 50:1242–1259, 2004.
- [63] P. Mhaskar, N. H. El-Farra, and P. D. Christofides. Predictive control of switched nonlinear systems with scheduled mode transitions. *IEEE Trans. Automat. Contr.*, 50:1670–1680, 2005.
- [64] P. Mhaskar, N. H. El-Farra, and P. D. Christofides. Robust hybrid predictive control of nonlinear systems. *Automatica*, 41:209–217, 2005.

- [65] P. Mhaskar, N. H. El-Farra, and P. D. Christofides. Stabilization of nonlinear systems with state and control constraints using Lyapunov-based predictive control. *Syst. & Contr. Lett.*, 55:650–659, 2006.
- [66] P. Mhaskar, A. Gani, and P. D. Christofides. Fault-tolerant control of nonlinear processes: Performance-based reconfiguration and robustness. *Int. J. Rob. & Non. Contr.*, 16:91–111, 2006.
- [67] P. Mhaskar, A. Gani, N. H. El-Farra, C. McFall, P. D. Christofides, and J. F. Davis. Integrated fault-detection and fault-tolerant control of process systems. *AIChE J.*, 52:2129–2148, 2006.
- [68] P. Mhaskar, A. Gani, C. McFall, P. D. Christofides, and J. F. Davis. Fault-tolerant control of nonlinear process systems subject to sensor faults. *AIChE J.*, 53:654–668, 2007.
- [69] P. Mhaskar, C. McFall, A. Gani, P.D. Christofides, and J. F. Davis. Fault-tolerant control of nonlinear systems: Fault-detection and isolation and controller reconfiguration. In *Proceedings of American Control Conference*, pages 5115–5122, Minneapolis, MN, 2006.
- [70] P. Mhaskar, C. McFall, A. Gani, P.D. Christofides, and J.F. Davis. Isolation and handling of actuator faults in nonlinear systems. *Automatica*, 44:53–62, 2008.
- [71] H. Michalska and D. Q. Mayne. Moving horizon observers and observer-based control. *IEEE Trans. Automat. Contr.*, 40:995–1006, 1995.
- [72] D. Nešić and A. R. Teel. Input-output stability properties of networked control systems. *IEEE Transactions on Automatic Control*, 49(10):1650–1667, 2004.

- [73] D. Nešić and A. R. Teel. Input-to-state stability of networked control systems. *Automatica*, 40(12):2121–2128, 2004.
- [74] A. Negiz and A. Cinar. Statistical monitoring of multivariable dynamic processes with state-space models. *AIChE J.*, 43:2002–2020, 1997.
- [75] P.R.C. Nelson, P.A Taylor, and J.F. MacGregor. Missing data methods in PCA and PLS: Score calculations with incomplete observations. *Chemometrics and Intelligent Laboratory Systems*, 35:45–65, 1996.
- [76] H. Niemann, A. Saberi., A. A. Stoorvogel, and P. Sannuti. Exact, almost and delayed fault detection - an observer based approach. *Inter. J. Rob. & Non. Contr.*, 9:215–238, 1999.
- [77] I. Nimmo. Adequately address abnormal operations. *Chem. Eng. Prog.*, 91:36–45, 1995.
- [78] P. Nomikos and J. F. Macgregor. Monitoring batch processes using multiway principal component analysis. *AIChE J.*, 40:1361–1375, 1994.
- [79] B. A. Ogunnaike and W. H. Ray. *Process Dynamics, Modeling, and Control*. Oxford University Press, New York, 1994.
- [80] R. J. Patton. Fault-tolerant control systems: The 1997 situation. In *Proceedings of the IFAC Symposium SAFEPROCESS 1997*, pages 1033–1054, Hull, United Kingdom, 1997.
- [81] C. V. Rao and J. B. Rawlings. Constrained process monitoring: Moving-horizon approach. *AIChE J.*, 48:97–109, 2002.
- [82] J. B. Riggs and M. N. Karim. *Chemical and Bio-Process control, 3rd ed.* Ferret, Lubbock, Texas, 2006.

- [83] M. W. Robertson, J. C. Watters, P. B. Desphande, J. Z. Assef, and I.M. Alatiqi. Model based control for reverse osmosis desalination processes. *Desalination*, 104:59–68, 1996.
- [84] D. R. Rollins and J. F. Davis. An unbiased estimation technique when gross errors exist in process measurements. *AIChE J.*, 38:563–572, 1992.
- [85] D. R. Rollins and J. F. Davis. Unbiased estimation of gross errors when the covariance matrix is unknown. *AIChE J.*, 39:1335–1341, 1993.
- [86] A. Saberi, A. A. Stoorvogel, P. Sannuti, and H. Niemann. Fundamental problems in fault detection and identification. *Inter. J. Rob. & Non. Contr.*, 10:1209–1236, 2000.
- [87] D. D. Siljak. Reliable control using multiple control systems. *Int. J. Contr.*, 31:302–339, 1980.
- [88] E. D. Sontag. A ‘universal’ construction of Artstein’s theorem on nonlinear stabilization. *Systems & Control Letters*, 13:117–123, 1989.
- [89] M. Soroush and N. Zambare. Nonlinear output feedback control of a class of polymerization reactors. *IEEE Trans. Contr. Syst. Tech.*, 8:310–320, 2000.
- [90] E. Tatara and A. Cinar. An intelligent system for multivariate statistical process monitoring and diagnosis. *ISA Transactions*, 41:255–270, 2002.
- [91] A. R. Teel. Global stabilization and restricted tracking for multiple integrators with bounded controls. *Syst. & Contr. Lett.*, 18:165–171, 1992.
- [92] S. Valluri, M. Soroush, and M. Nikravesh. Shortest-prediction-horizon nonlinear model-predictive control. *Chem. Eng. Sci.*, 53:273–292, 1998.



- [93] G. Walsh, O. Beldiman, and L. Bushnell. Asymptotic behavior of nonlinear networked control systems. *IEEE Transactions on Automatic Control*, 46(7):1093–1097, 2001.
- [94] G. Walsh, H. Ye, and L. Bushnell. Stability analysis of networked control systems. *IEEE Transactions on Control Systems Technology*, 10(3):438–446, 2002.
- [95] N. E. Wu. Coverage in fault-tolerant control. *Automatica*, 40:537–548, 2004.
- [96] G. H. Yang, J. L. Wang, and Y. C. Soh. Reliable  $H_\infty$  control design for linear systems. *Automatica*, 37:717–725, 2001.
- [97] G. H. Yang, S. Y. Zhang, J. Lam, and J. Wang. Reliable control using redundant controllers. *IEEE Trans. Autom. Contr.*, 43:1588–1593, 1998.
- [98] E. B. Ydstie. Certainty equivalence adaptive control: Paradigms puzzles and switching. In *Proceedings of 5th International Conference on Chemical Process Control*, pages 9–23, Tahoe City, CA, 1997.
- [99] E. B. Ydstie. New vistas for process control: Integrating physics and communication networks. *AIChE J.*, 48:422–426, 2002.
- [100] X. D. Zhang, T. Parisini, and M. M. Polycarpou. Adaptive fault-tolerant control of nonlinear uncertain systems: An information-based diagnostic approach. *IEEE Trans. Automat. Contr.*, 49:1259–1274, 2004.
- [101] D. H. Zhou and P. M. Frank. Fault diagnostics and fault tolerant control. *IEEE Transactions on Aerospace and Electronic Systems*, 34:420–427, 1998.