

# **Fault-tolerant Synchronization of Autonomous Underwater Vehicles**

Faegheh Amirarfaei

A Thesis in  
the Department of  
Electrical and Computer Engineering  
Concordia University

Presented in Partial Fulfillment of the Requirements  
For the Degree of Master of Applied Science (Electrical  
Engineering) at  
Concordia University  
Montreal, Quebec, Canada

April 2016

© Copyright by Faegheh Amirarfaei, 2016

**CONCORDIA UNIVERSITY  
SCHOOL OF GRADUATE STUDIES**

This is to certify that the thesis prepared

By: Faegheh Amirarfael

Entitled: "Fault-tolerant Synchronization of Autonomous Underwater Vehicles"

and submitted in partial fulfillment of the requirements for the degree of

**Master of Applied Science**

Complies with the regulations of this University and meets the accepted standards with respect to originality and quality.

Signed by the final examining committee:

_____	Chair
Dr. R. Raut	
_____	Examiner, External
Dr. R. Sedaghati (MIE)	To the Program
_____	Examiner
Dr. S. Hashtrudi Zad	
_____	Supervisor
Dr. K. Khorasani	
_____	Supervisor
Dr. N. Meskin	

Approved by: \_\_\_\_\_  
Dr. W. E. Lynch, Chair  
Department of Electrical and Computer Engineering

\_\_\_\_\_20\_\_\_\_\_

\_\_\_\_\_  
Dr. Amir Asif, Dean  
Faculty of Engineering and Computer  
Science

# ABSTRACT

## Fault-tolerant Synchronization of Autonomous Underwater Vehicles

Faegheh Amirarfaei

The main objective of this thesis is to develop a fault-tolerant and reconfigurable synchronization scheme based on model-based control protocols for stern and sail hydroplanes that are employed as actuators in the attitude control subsystem (ACS) of an autonomous underwater vehicle (AUV). In this thesis two control approaches are considered for synchronization, namely i) state feedback synchronization, and ii) output feedback synchronization. Both problems are tackled by proposing a passive control approach as well as an active reconfiguration (re-designing the control gains).

For the "state feedback" synchronization scheme, to achieve consensus the relative/absolute measurements of the AUV's states (position and attitude) are available. The states of a longitudinal model of an AUV are mainly heave, pitch, and their associated rates. For the state feedback problem we employ a static protocol, and it is shown that the multi-agent system will synchronize in the stochastic mean square sense in the presence of measurement noise. However, the resulting performance index defined as the accumulated sum of variations of control inputs and synchronization errors is high. To deal with this problem, Kalman filtering is used for states estimation that are used in synchronization protocol. Moreover, the effects of parameter uncertainty of the agent's dynamics are also investigated through simulation results. By employing the static protocol it is demonstrated that when a loss of effectiveness (LOE) or float fault occurs the synchronization can still be achieved under some conditions. Finally, one of the main problems that is tackled in the state feedback scenario is our proposed proportional-integral (PI) control methodology to deal with the lock in place (LIP) fault. It is shown that if the LIP fault occurs, by employing a PI protocol the synchronization could still be achieved. Finally, our proposed dynamic synchronization protocol methodology is applied given that the fault (LOE/float) severity is known. Since after a fault occurrence the agents become heterogeneous, employing the dynamic scheme makes the task of reconfiguration (redesigning the gains) more effective.

For the "output feedback" synchronization approach, to achieve consensus relative/absolute measurements of the AUV's states except the pitch rate are available. For the output feedback problem a dynamic protocol through a Luenberger observer is first employed for state estimation

and the synchronization achievement is demonstrated. Then, a system under state and measurement noise is considered, and it is shown that by employing a Kalman filter for the state estimation; the multi-agent system will synchronize in the stochastic mean square sense. Furthermore, by employing the static protocol, it is shown that when a LOE/float fault occurs the synchronization is still achieved under certain conditions. Finally, one of the main problems that is tackled in the output feedback scenario is our proposed dynamic controller methodology. The results of this scheme are compared with another approach that exploits both dynamic controller and dynamic observer. The former approach has less computational effort and results in more a robust control with respect to the actuator fault. The reason is that the later method employs an observer that uses the control input matrix information. When fault occurs, this information will not be correct any more. However, if there is a need to redesign the synchronization gains under faulty scenario, the later methodology is preferred. The reason is that the former approach becomes complicated when there is a fault even though its severity is known.

In this thesis, fault-tolerant synchronization of autonomous underwater vehicles is considered. In the first chapter a brief introduction on the motivation, problem definition, objectives and the methodologies that are used in the dissertation are discussed. A literature review on research dedicated to synchronization, fault diagnosis, and fault-tolerant control is provided. In Chapter 2, a through literature review on unmanned underwater vehicles is covered. It also comprises a comprehensive background information and definitions including algebraic graph theory, matrix theory, and fault modeling. In the problem statement, the two main problems in this thesis, namely state feedback synchronization and output feedback synchronization are discussed. Chapters 3 and 4 will cover these two problems, their solutions, and the corresponding simulation results that are provided. Finally, Chapter 5 includes a discussion of conclusions and future work.

## ACKNOWLEDGMENTS

During my master studies at Concordia University, which was along with my full-time work at Pratt and Whitney Canada as a Control System Design Engineer, I had the chance to meet and interact with many intelligent and inspiring people. This journey would have been much harder and a lot less fun without them. For this, I must sincerely thank all the wonderful people I have met.

First of all, my deepest gratitude must go to my academic supervisor, Prof. Khashayar Khorasani of the Electrical and Computer Engineering Department at Concordia University, Montreal, Canada. His knowledge, insight, vision, inspiration, and encouragement have guided me through all these years.

I would also like to specially thank my co-supervisor, Dr. Nader Meskin, Associate Professor of Electrical Engineering Department at Qatar University and Adjunct Professor at Concordia University, who has contributed to this thesis through his instructive and enlightening feedbacks.

I am incredibly grateful to my parents, Nahid and Mohammadreza, and my beautiful and loving sister, Yasamin, who gave me their never-ending love and support during all these years. My special gratitude is to my mother for her believe in me and her encourage to pursue my goals. I am also very grateful to my wonderful father who has always been a source of aspiration in my life with his intelligence, love, and kindness.

Finally, I would like to thank my loving husband, Aram, for all his love, patience, support, and encouragement over the years. With his vision, he has always helped me to be step forward in my career. His lovely smile, emotional support, and strong encouragement have always been the driving force of my efforts. It is him that makes doing all things worthwhile.

April 2016, Montreal

# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Motivation . . . . .	1
1.2	Problem Definition . . . . .	2
1.3	Objective . . . . .	3
1.4	Methodology . . . . .	3
1.5	Literature Review . . . . .	4
1.5.1	Synchronization versus Network Topology and Agents' Dynamics Complexities . . . . .	5
1.5.2	Heterogeneous Systems and Synchronization . . . . .	9
1.5.3	Fault-Tolerant Control and Fault Detection and Diagnosis . . . . .	9
1.5.4	Fault-Tolerant Control of Multi-agent Systems . . . . .	12
1.6	Contributions . . . . .	14
1.7	Summary . . . . .	15
<b>2</b>	<b>Background Information and Definitions</b>	<b>16</b>
2.1	Unmanned Underwater Vehicles . . . . .	16
2.1.1	AUV Modeling . . . . .	17
2.1.2	AUV Nonlinear Equations of motion . . . . .	17
2.1.3	Hydrodynamic Forces and Moments . . . . .	21
2.1.4	Hydrodynamic Damping . . . . .	23
2.1.5	Restoring Forces and Moments . . . . .	23
2.1.6	AUV's Model Uncertainties . . . . .	24
2.1.7	Environmental Disturbances and Stochastic Differential Equation Model . . . . .	25
2.1.8	Linear Time-Varying ROV Equations of Motion . . . . .	26
2.1.9	Reduced-order Model of AUV . . . . .	27
2.1.10	Sensors and Actuators in an Unmanned Underwater Vehicle . . . . .	29
2.2	Algebraic Graph Theory Background . . . . .	33
2.3	Algebra and Matrix Theory Background . . . . .	35
2.4	Fault Modeling . . . . .	37
2.5	Summary . . . . .	38
<b>3</b>	<b>State Feedback Synchronization</b>	<b>39</b>
3.1	Network Structure and Model Dynamics Equation . . . . .	39
3.2	State Feedback Consensus under Healthy Scenario . . . . .	41
3.2.1	Simulation Results . . . . .	43

3.3	State Feedback Consensus Subject to the Process and Measurement Noises . . . . .	47
3.3.1	Simulation Results . . . . .	50
3.4	State Feedback Consensus and Kalman Filtering . . . . .	60
3.4.1	Simulation Results . . . . .	61
3.5	State Feedback Consensus Subject to the Agents Dynamics Uncertainty . . . . .	67
3.6	Fault-Tolerant Static State Feedback Synchronization Protocol Subject to the LOE Fault . . . . .	74
3.7	Fault-Tolerant Static State Feedback Synchronization Protocol Subject to the Float Fault . . . . .	77
3.8	Simulation Results for the Float and LOE Faults . . . . .	80
3.8.1	AUV with single-input channel . . . . .	80
3.8.2	AUV with Multi-input Channel . . . . .	81
3.9	Fault-Tolerant Static State Feedback Synchronization Protocol of Integrator Systems Subject to the LOE/Float Fault . . . . .	88
3.9.1	Single Integrator Systems . . . . .	88
3.9.2	Double Integrator System . . . . .	90
3.10	Reconfigurable State Feedback Synchronization Protocol Subject to the LIP Fault . . . . .	91
3.11	Reconfigurable State Feedback Dynamic Synchronization Protocol Subject to the LOE/Float Fault . . . . .	99
3.11.1	Simulation Results . . . . .	101
3.12	Summary . . . . .	108
<b>4</b>	<b>Output Feedback Synchronization</b>	<b>109</b>
4.1	Output Feedback Synchronization and Luenberger Observer . . . . .	109
4.1.1	Simulation Results . . . . .	111
4.2	Output Feedback Synchronization and Kalman Filtering . . . . .	119
4.2.1	Simulation Results . . . . .	122
4.3	Output Feedback Synchronization Subject to Faults . . . . .	130
4.4	Fault-Tolerant Static Output Feedback Synchronization Protocol with Single-input Channel Subject to the LOE Fault . . . . .	130
4.5	Observer-based Dynamic Output Feedback Synchronization Protocol Subject to the LOE/Float Fault . . . . .	134
4.5.1	Simulation Results . . . . .	137
4.6	Dynamic Output Feedback Synchronization Protocol Subject to the LOE/Float Fault . . . . .	154
4.6.1	Simulation Results . . . . .	157
4.7	Summary and Comparison of Three Output Feedback Synchronization Methods . . . . .	177
<b>5</b>	<b>Conclusions and Future Work</b>	<b>179</b>
	<b>Appendices</b>	<b>181</b>
<b>A</b>	<b>Fault Diagnosis</b>	<b>182</b>

# List of Figures

1.1	Control structures classifications. . . . .	10
1.2	Control algorithms classifications. . . . .	11
1.3	Classification of model-based FDD methods. . . . .	11
2.1	Inertial reference and body-fixed frames [1]. . . . .	20
2.2	Submarine actuators [2]. . . . .	30
2.3	Submarine rudder and sterns [2]. . . . .	31
2.4	WTC location and weight distribution [2]. . . . .	31
3.1	Network topology. . . . .	40
3.2	Control input signals $u(t)$ for static state feedback protocol under healthy scenario. .	44
3.3	Synchronization of state 1 for static state feedback protocol under healthy scenario.	45
3.4	Synchronization of state 2 for static state feedback protocol under healthy scenario.	45
3.5	Synchronization of state 3 for static state feedback protocol under healthy scenario.	46
3.6	Synchronization of state 4 for static state feedback protocol under healthy scenario.	46
3.7	Control input signal of agent 1 for static state feedback protocol subject to noise - gain set 1. . . . .	50
3.8	Control input signal of agent 2 for static state feedback protocol subject to noise - gain set 1. . . . .	51
3.9	Control input signal of agent 3 for static state feedback protocol subject to noise - gain set 1. . . . .	51
3.10	Control input signal of agent 4 for static state feedback protocol subject to noise - gain set 1. . . . .	52
3.11	Synchronization of state 1 for static state feedback protocol subject to noise - gain set 1. . . . .	52
3.12	Synchronization of state 2 for static state feedback protocol subject to noise - gain set 1. . . . .	53
3.13	Synchronization of state 3 for static state feedback protocol subject to noise - gain set 1. . . . .	53

3.14	Synchronization of state 4 for static state feedback protocol subject to noise - gain set 1. . . . .	54
3.15	Control input signal of agent 1 for static state feedback protocol subject to noise - gain set 2. . . . .	55
3.16	Control input signal of agent 2 for static state feedback protocol subject to noise - gain set 2. . . . .	56
3.17	Control input signal of agent 3 for static state feedback protocol subject to noise - gain set 2. . . . .	56
3.18	Control input signal of agent 4 for static state feedback protocol subject to noise - gain set 2. . . . .	57
3.19	Synchronization of state 1 for static state feedback protocol subject to noise - gain set 2. . . . .	57
3.20	Synchronization of state 2 for static state feedback protocol subject to noise - gain set 2. . . . .	58
3.21	Synchronization of state 3 for static state feedback protocol subject to noise - gain set 2. . . . .	58
3.22	Synchronization of state 4 for static state feedback protocol subject to noise - gain set 2. . . . .	59
3.23	State estimation of state 1 for static state feedback protocol subject to noise and by employing Kalman filter. . . . .	61
3.24	State estimation of state 2 for static state feedback protocol subject to noise and by employing Kalman filter. . . . .	62
3.25	State estimation of state 3 for static state feedback protocol subject to noise and by employing Kalman filter. . . . .	63
3.26	State estimation of state 4 for static state feedback protocol subject to noise and by employing Kalman filter. . . . .	63
3.27	Control input signals $u(t)$ for static state feedback protocol subject to noise and by employing Kalman filter. . . . .	64
3.28	Synchronization of state 1 for static state feedback protocol subject to noise and by employing Kalman filter. . . . .	64
3.29	Synchronization of state 2 for static state feedback protocol subject to noise and by employing Kalman filter. . . . .	65
3.30	Synchronization of state 3 for static state feedback protocol subject to noise and by employing Kalman filter. . . . .	65
3.31	Synchronization of state 4 for static state feedback protocol subject to noise and by employing Kalman filter. . . . .	66

3.32	State performance index for static state feedback protocol subject to uncertainty on $A(1,1)$ . . . . .	68
3.33	Control input performance index for static state feedback protocol subject to uncertainty on $A(1,1)$ . . . . .	69
3.34	Total performance index for static state feedback protocol subject to uncertainty on $A(1,1)$ . . . . .	69
3.35	State performance index for static state feedback protocol subject to uncertainty on $A(1,2)$ . . . . .	70
3.36	Control input performance index for static state feedback protocol subject to uncertainty on $A(1,2)$ . . . . .	70
3.37	Total performance index for static state feedback protocol subject to uncertainty on $A(1,2)$ . . . . .	71
3.38	State performance index for static state feedback protocol subject to uncertainty on $A(2,1)$ . . . . .	71
3.39	Control input performance index for static state feedback protocol subject to uncertainty on $A(2,1)$ . . . . .	72
3.40	Total performance index for static state feedback protocol subject to uncertainty on $A(2,1)$ . . . . .	72
3.41	State performance index for static state feedback protocol subject to uncertainty on $A(2,2)$ . . . . .	73
3.42	Control input performance index for static state feedback protocol subject to uncertainty on $A(2,2)$ . . . . .	73
3.43	Total performance index for static state feedback protocol subject to uncertainty on $A(2,2)$ . . . . .	74
3.44	State performance index for static state feedback protocol without fault recovery - gain set 1. . . . .	82
3.45	Control input performance index for static state feedback protocol without fault recovery - gain set 1. . . . .	83
3.46	Total performance index for static state feedback protocol without fault recovery - gain set 1. . . . .	83
3.47	State performance index for static state feedback protocol without fault recovery - gain set 2. . . . .	84
3.48	Control input performance index for static state feedback protocol without fault recovery - gain set 2. . . . .	85
3.49	Total performance index for static state feedback protocol without fault recovery - gain set 2. . . . .	85

3.50	State performance index for static state feedback protocol without fault recovery - gain set 3. . . . .	86
3.51	Control input performance index for static state feedback protocol without fault recovery - gain set 3. . . . .	87
3.52	Total performance index for static state feedback protocol without fault recovery - gain set 3. . . . .	87
3.53	State performance index for dynamic state feedback protocol without fault recovery - gain set 1. . . . .	102
3.54	Control input performance index for dynamic state feedback protocol without fault recovery - gain set 1. . . . .	103
3.55	Total performance index for dynamic state feedback protocol without fault recovery - gain set 1. . . . .	103
3.56	State performance index for dynamic state feedback protocol without fault recovery - gain set 2. . . . .	104
3.57	Control input performance index for dynamic state feedback protocol without fault recovery - gain set 2. . . . .	105
3.58	Total performance index for dynamic state feedback protocol without fault recovery - gain set 2. . . . .	105
3.59	States performance index for dynamic state feedback protocol without fault recovery - gain set 3. . . . .	106
3.60	Control inputs performance index for dynamic state feedback protocol without fault recovery - gain set 3. . . . .	107
3.61	Total performance index for dynamic state feedback protocol without fault recovery - gain set 3. . . . .	107
4.1	State estimation of state 1 for dynamic output feedback protocol under healthy scenario. . . . .	112
4.2	State estimation of state 2 for dynamic output feedback protocol under healthy scenario. . . . .	112
4.3	State estimation of state 3 for dynamic output feedback protocol under healthy scenario. . . . .	113
4.4	State estimation of state 4 for dynamic output feedback protocol under healthy scenario. . . . .	113
4.5	Controller state 1 for dynamic output feedback protocol under healthy scenario. . .	114
4.6	Controller state 2 for dynamic output feedback protocol under healthy scenario. . .	114
4.7	Controller state 3 for dynamic output feedback protocol under healthy scenario. . .	115

4.8	Controller state 4 for dynamic output feedback protocol under healthy scenario. . .	115
4.9	Control Input signal $u_1(t)$ for dynamic output feedback protocol under healthy scenario. . . . .	116
4.10	Control input signal $u_2(t)$ for dynamic output feedback protocol under healthy scenario. . . . .	116
4.11	Synchronization of state 1 for dynamic output feedback protocol under healthy scenario. . . . .	117
4.12	Synchronization of state 2 for dynamic output feedback protocol under healthy scenario. . . . .	117
4.13	Synchronization of state 3 for dynamic output feedback protocol under healthy scenario. . . . .	118
4.14	Synchronization of state 4 for dynamic output feedback protocol under healthy scenario. . . . .	118
4.15	State estimation of state 1 for dynamic output feedback protocol subject to noise and by employing Kalman filter. . . . .	123
4.16	State estimation of state 2 for dynamic output feedback protocol subject to noise and by employing Kalman filter. . . . .	123
4.17	State estimation of state 3 for dynamic output feedback protocol subject to noise and by employing Kalman filter. . . . .	124
4.18	State estimation of state 4 for dynamic output feedback protocol subject to noise and by employing Kalman filter. . . . .	124
4.19	Controller state 1 for dynamic output feedback protocol subject to noise and by employing Kalman filter. . . . .	125
4.20	Controller state 2 for dynamic output feedback protocol subject to noise and by employing Kalman filter. . . . .	125
4.21	Controller state 3 for dynamic output feedback protocol subject to noise and by employing Kalman filter. . . . .	126
4.22	Controller state 4 for dynamic output feedback protocol subject to noise and by employing Kalman filter. . . . .	126
4.23	Control input signal $u_1(t)$ for dynamic output feedback protocol subject to noise and by employing Kalman filter. . . . .	127
4.24	Control input signal $u_2(t)$ for dynamic output feedback protocol subject to noise and by employing Kalman filter. . . . .	127
4.25	Synchronization of state 1 for dynamic output feedback protocol subject to noise and by employing Kalman filter. . . . .	128

4.26	Synchronization of state 2 for dynamic output feedback protocol subject to noise and by employing Kalman filter. . . . .	128
4.27	Synchronization of state 3 for dynamic output feedback protocol subject to noise and by employing Kalman filter. . . . .	129
4.28	Synchronization of state 4 for dynamic output feedback protocol subject to noise and by employing Kalman filter. . . . .	129
4.29	State performance index for dynamic output feedback protocol without fault recovery. . . . .	138
4.30	Control input performance index for dynamic output feedback protocol without fault recovery. . . . .	138
4.31	Total performance index for dynamic output feedback protocol without fault recovery. . . . .	139
4.32	State estimation of state 1 for dynamic output feedback protocol without fault recovery (float fault). . . . .	140
4.33	State estimation of state 2 for dynamic output feedback protocol without fault recovery (float fault). . . . .	140
4.34	State estimation of state 3 for dynamic output feedback protocol without fault recovery (float fault). . . . .	141
4.35	State estimation of state 4 for dynamic output feedback protocol without fault recovery (float fault). . . . .	141
4.36	Controller state 1 for dynamic output feedback protocol without fault recovery (float fault). . . . .	142
4.37	Controller state 2 for dynamic output feedback protocol without fault recovery (float fault). . . . .	142
4.38	Controller state 3 for dynamic output feedback protocol without fault recovery (float fault). . . . .	143
4.39	Controller state 4 for dynamic output feedback protocol without fault recovery (float fault). . . . .	143
4.40	Control input signal $u_1(t)$ for dynamic output feedback protocol without fault recovery (float fault). . . . .	144
4.41	Control input signal $u_2(t)$ for dynamic output feedback protocol without fault recovery (float fault). . . . .	144
4.42	Consensus achievement of state 1 for dynamic output feedback protocol without fault recovery (float fault). . . . .	145
4.43	Consensus achievement of state 2 for dynamic output feedback protocol without fault recovery (float fault). . . . .	145

4.44	Consensus achievement of state 3 for dynamic output feedback protocol without fault recovery (float fault).	146
4.45	Consensus achievement of state 4 for dynamic output feedback protocol without fault recovery (float fault).	146
4.46	State estimation of state 1 for dynamic output feedback protocol with fault recovery (float fault).	147
4.47	State estimation of state 2 for dynamic output feedback protocol with fault recovery (float fault).	148
4.48	State estimation of state 3 for dynamic output feedback protocol with fault recovery (float fault).	148
4.49	State estimation of state 4 for dynamic output feedback protocol with fault recovery (float fault).	149
4.50	Controller state 1 for dynamic output feedback protocol with fault recovery (float fault).	149
4.51	Controller state 2 for dynamic output feedback protocol with fault recovery (float fault).	150
4.52	Controller state 3 for dynamic output feedback protocol with fault recovery (float fault).	150
4.53	Controller state 4 for dynamic output feedback protocol with fault recovery (float fault).	151
4.54	Control input signal $u_1(t)$ for dynamic output feedback protocol with fault recovery (float fault).	151
4.55	Control input signal $u_2(t)$ for dynamic output feedback protocol with fault recovery (float fault).	152
4.56	Synchronization of state 1 for dynamic output feedback protocol with fault recovery (float fault).	152
4.57	Synchronization of state 2 for dynamic output feedback protocol with fault recovery (float fault).	153
4.58	Synchronization of state 3 for dynamic output feedback protocol with fault recovery (float fault).	153
4.59	Synchronization of state 4 for dynamic output feedback protocol with fault recovery (float fault).	154
4.60	Controller state 1 for proposed dynamic output feedback protocol under healthy scenario.	158
4.61	Controller state 2 for proposed dynamic output feedback protocol under healthy scenario.	158

4.62	Controller state 3 for proposed dynamic output feedback protocol under healthy scenario. . . . .	159
4.63	Controller state 4 for proposed dynamic output feedback protocol under healthy scenario. . . . .	159
4.64	Control input signal $u_1(t)$ for proposed dynamic output feedback protocol under healthy scenario. . . . .	160
4.65	Control input signal $u_2(t)$ for proposed dynamic output feedback protocol under healthy scenario. . . . .	160
4.66	Synchronization of state 1 for proposed dynamic output feedback protocol under healthy scenario. . . . .	161
4.67	Synchronization of state 2 for proposed dynamic output feedback protocol under healthy scenario. . . . .	161
4.68	Synchronization of state 3 for proposed dynamic output feedback protocol under healthy scenario. . . . .	162
4.69	Synchronization of state 4 for proposed dynamic output feedback protocol under healthy scenario. . . . .	162
4.70	State performance index for proposed dynamic output feedback protocol without fault recovery. . . . .	163
4.71	Control input performance index for proposed dynamic output feedback protocol without fault recovery. . . . .	164
4.72	Total performance index for proposed dynamic output feedback protocol without fault recovery. . . . .	164
4.73	Controller state 1 for proposed dynamic output feedback protocol without fault recovery (float fault). . . . .	165
4.74	Controller state 2 for proposed dynamic output feedback protocol without fault recovery (float fault). . . . .	166
4.75	Controller state 3 for proposed dynamic output feedback protocol without fault recovery (float fault). . . . .	166
4.76	Controller state 4 for proposed dynamic output feedback protocol without fault recovery (float fault). . . . .	167
4.77	Control input signal $u_1(t)$ for proposed dynamic output feedback protocol without fault recovery (float fault). . . . .	167
4.78	Control input signal $u_2(t)$ for proposed dynamic output feedback protocol without fault recovery (float fault). . . . .	168
4.79	Synchronization of state 1 for proposed dynamic output feedback protocol without fault recovery (float fault). . . . .	168

4.80	Synchronization of state 2 for proposed dynamic output feedback protocol without fault recovery (float fault). . . . .	169
4.81	Synchronization of state 3 for proposed dynamic output feedback protocol without fault recovery (float fault). . . . .	169
4.82	Synchronization of state 4 for proposed dynamic output feedback protocol without fault recovery (float fault). . . . .	170
4.83	Controller state 1 for proposed dynamic output feedback protocol with fault recovery (float fault). . . . .	171
4.84	Controller state 2 for proposed dynamic output feedback protocol with fault recovery (float fault). . . . .	172
4.85	Controller state 3 for proposed dynamic output feedback protocol with fault recovery (float fault). . . . .	172
4.86	Controller state 4 for proposed dynamic output feedback protocol with fault recovery (float fault). . . . .	173
4.87	Control input signal $u_1(t)$ for proposed dynamic output feedback protocol with fault recovery (float fault). . . . .	173
4.88	Control input signal $u_2(t)$ for proposed dynamic output feedback protocol with fault recovery (float fault). . . . .	174
4.89	Synchronization of state 1 for proposed dynamic output feedback protocol with fault recovery (float fault). . . . .	174
4.90	Synchronization of state 2 for proposed dynamic output feedback protocol with fault recovery (float fault). . . . .	175
4.91	Synchronization of state 3 for proposed dynamic output feedback protocol with fault recovery (float fault). . . . .	175
4.92	Synchronization of state 4 for proposed dynamic output feedback protocol with fault recovery (float fault). . . . .	176

# List of Tables

2.1	AUV model notations. . . . .	20
3.1	Monte-Carlo simulation results for performance indices of the single agents and network in the presence of noise and by employing the first set of gains. The Network refers to the averaging of the entire team performance. . . . .	54
3.2	Monte-Carlo simulation results for performance indices of the single agents and network in the presence of noise and by employing the second set of gains. . . . .	59
3.3	Monte-Carlo simulation results for performance indices of the single agents and network in presence of noise and by employing Kalman filtering. . . . .	66
3.4	Variation of the real part of Laplacian eigenvalues with change of the actuator effectiveness in agent 1. . . . .	80
3.5	Variation of the real part of Laplacian eigenvalues with change of the actuator effectiveness in agent 2. . . . .	80
3.6	Variation of the real part of Laplacian eigenvalues with change of the actuator effectiveness in agent 3. . . . .	80
3.7	Variation of the real part of Laplacian eigenvalues with change of the actuator effectiveness in agent 4. . . . .	81
3.8	Variation of the real part of Laplacian eigenvalues with change of the actuator effectiveness in agent 5. . . . .	81
4.1	Comparison of state performance indices between healthy case and the one subject to the float fault. . . . .	176
4.2	Comparison of control input performance indices between healthy case and the one subject to the float fault. . . . .	177
4.3	Comparison of performance indices between healthy case and the one subject to the float fault. . . . .	177

## Notations

$\forall$	for all
$\exists$	if there exists
$\nexists$	if there does not exist
$\in$	belongs to
$\notin$	does not belongs to
$\cap$	intersection
$\cup$	union
$\emptyset$	empty set
$\Sigma$	summation
$\otimes$	Kronecker product
$\max$	max
$\min$	min
$\infty$	infinity
$\mathbb{R}^{m \times n}$	set of $m \times n$ real matrices
$z^*$	complex conjugate of number $z$
$\text{Re}z$	Real part of $z$
$\text{Im}z$	Imaginary part of $z$
$A^T$	transpose of a real matrix $A$
$A^H$	conjugate transpose of a real matrix $A$
$A > 0$	a positive matrix $A$
$A \geq 0$	a nonnegative matrix $A$
$A > 0$	a positive definite matrix $A$
$A \geq 0$	a positive semi-definite matrix $A$
$\rho(A)$	spectral radius of matrix $A$
$\lambda_i(A)$	the $i$ th eigenvalue of matrix $A$
$\lambda_{\max}(A)$	the maximum eigenvalue of real symmetric matrix $A$
$\lambda_{\min}(A)$	the minimum eigenvalue of real symmetric matrix $A$
$\det(A)$	the maximum eigenvalue of real symmetric matrix $A$
$\text{rank}(A)$	rank of matrix $A$
$\text{diag}(a_1, \dots, a_N)$	a block matrix with diagonal entries $a_1$ to $a_N$
$\text{bblockdiag}(A_1, \dots, A_N)$	a block diagonal matrix with diagonal blocks $A_1$ to $A_N$
$\text{Co}(\cdot)$	convex hull
$\mathbf{1}_N$	$N \times 1$ column vector of all ones

$\mathbf{0}_N$   $N \times 1$  column vector of all zeros

$I_N$   $N \times N$  identity matrix

$0_N$   $N \times N$  zero matrix

# Chapter 1

## Introduction

### 1.1 Motivation

As oil and gas operators have moved into deeper waters (below 600 feet), the need for an economical access to the deep ocean has emerged in the industry. High quality and reliable pipeline inspection methods are now required to verify the integrity of subsea structure. The necessity of ocean access is not confined to offshore petroleum industry. Scientific and industrial applications include under ice studies, air crash investigations, large scale ocean structure survey, and pollution monitoring.

Historically, manned submersibles and remotely operated vehicles (ROVs) have been used to suit this purpose. As compared to manned submersibles, ROVs cost is lower and it significantly reduces the human risk. However, since ROV still relies on towing method, the process of moving the robot and the surface ship limits the effectiveness and the efficiency of the ROVs survey. In addition, the cables can become tangled, altering the surveys and affecting accuracy. These methods have constraints on payload size, power consumption and data transmission rate. They also suffer from the inherent drag of the tether and for deep-water use, they require a large operating vessel.

As opposed to manned submersibles and ROVs, the autonomous underwater vehicles (AUVs) provide an economical access to the ocean by lacking a tether and having a small vessel size. Powered mainly by fuel cell battery, the AUVs can operate for up to two days. The vehicles can also work in deeper waters than the traditional towing methods. A hybrid platform of an autonomous underwater vehicle and a remotely operated vehicle (ROV) are also used in industry to stay underwater for lengthy periods of time.

The autonomous underwater vehicles were introduced to the oil and gas industry initially for

the use in making detailed maps of the seafloor prior to constructing subsea infrastructure. Their use also makes it possible to survey post-lay pipes including spans, cathode erosion, leaks, movement, and damage monitoring. As another offshore application of AUVs, the environmental effect monitoring (EEM) is an important tool in environmental risk assessment (ERA) which has been conducted worldwide in offshore petroleum industry.

All the above tasks are complex, so that it is super beneficial if underwater vehicles work in parallel and in a collaborative platform. This highly helps the efficiency and quality of results through data redundancy and increased robustness. It also makes it feasible to carry out missions that are impossible or really hard to be performed by one vehicle. This encourages the use of multi-agent platform of autonomous underwater vehicles for underwater exploration, mapping, and threat tackling.

The broad applications of collaborative autonomous underwater vehicles motivate us to address their cooperative control. Within the last decade, a significant research has been devoted to investigate the AUVs and different aspects of consensus problems and cooperative control. However, AUVs in multi-agent platform has been barely covered in the literature. The problem also becomes more challenging in the presence of uncertainty, noise, and sensor or actuator fault. Hence, this thesis addresses the synchronization problem of unmanned underwater vehicles subject to all these imperfect conditions and faults.

## **1.2 Problem Definition**

In this thesis, the main objective is to tackle various faulty scenarios of consensus achievement of unmanned underwater vehicles. The main idea in consensus achievement is that the agents aim to synchronize to a common value. The problem of fault-tolerant and reconfigurable synchronization in multi-agent systems has not been broadly addressed in the literature. This motivates us to tackle this problem. The faults considered here are actuator faults. However, similar approaches could be applicable to sensor faults, as well.

The problem formulation throughout the thesis is general where we deal with a marginally stable linear system. However, the simulation results focus on the decoupled model of AUV's heave and pitch motions.

The problem definition starts with considering healthy scenario as in the literature, and investigates the effects of noise and uncertainty on consensus achievement. The noise models the measurement noise and disturbances that are applied to the system. The type of environmental disturbances that are applied to a deeply submersed AUV are ocean currents which are modeled as random walks. Moreover, the uncertainty of the model parameters is due to the fact that it is

a challenging task to be identified. These uncertainties are modeled by variation of parameters in the state-space dynamic matrix. In addition, as one of the main imperfect conditions addressed in the thesis, faults of loss of effectiveness (LOE), float and lock in place (LIP) types are considered.

The network topology and structure considered throughout the thesis is fixed. Moreover, the achieved consensus is determined based on the network's Laplacian matrix and the initial states of agents, and not a predefined trajectory or one generated by an exogenous system. Finally, all scenarios and problems defined in this thesis are classified under the two main category, namely state feedback and output feedback synchronization.

## 1.3 Objective

The main objective of this dissertation is to propose control schemes for fault-tolerant synchronization of autonomous underwater vehicles subject to actuator faults. In order to fulfill this objective, the longitudinal model of the AUVs is considered where the states are the heave and the pitch, and their associated rates. The AUVs are equipped with stern and sail (bow) hydroplanes. The bow control surface is chosen as the one subject to fault in order to verify the protocols that are used in the synchronization mission.

The research presented in this thesis attempts to keep consensus achievement of an underwater vehicle (AUV). The problem becomes challenging in the presence of uncertainties, noise, and actuator faults. Hence, this research addresses the synchronization problem of unmanned underwater vehicles subject to all these imperfect conditions specially the presence of actuator faults.

## 1.4 Methodology

The synchronization problem methodologies in this thesis are based on state feedback and output feedback synchronization where the objective is to keep and maintain the synchronization even subject to any fault occurrence.

For the "state feedback" synchronization scheme, to achieve consensus the relative measurements of the AUV's states (position and attitude) are available. The states of a longitudinal model of an AUV are mainly heave, pitch, and their associated rates. For the state feedback problem we employ a static protocol and it is shown that the multi-agent system will synchronize in the stochastic mean square sense in the presence of measurement noise. However, the resulting performance index defined as the accumulated sum of variations of control inputs and synchronization errors is high. To deal with this problem, Kalman filtering is used for states estimation that are used in synchronization protocol. Moreover, the effects of parameter uncertainty of the agent's

dynamics are also investigated through simulation results. By employing the static protocol it is demonstrated that when a loss of effectiveness (LOE) or float fault occurs under some conditions the synchronization can still be achieved. Finally, one of the main problems that is tackled in the state feedback scenario is our proposed PI control methodology to deal with the lock in place (LIP) fault. It is shown that if the LIP fault occurs, by employing a PI protocol the synchronization could still be achieved. Finally, our proposed dynamic synchronization protocol methodology is applied given that the fault (LOE/float) severity is known. Since after a fault occurrence the agents become heterogeneous, employing the dynamic scheme makes the task of reconfiguration (redesigning the gains) more effective.

For the "output feedback" synchronization approach, to achieve consensus absolute measurements of the AUV's states except the pitch rate are available. For the output feedback problem a dynamic protocol through a Luenberger observer is first employed for state estimation and the synchronization achievement is demonstrated. Then, a system under state measurement noise is considered, and it is shown that by employing a Kalman filter for the state estimation; the multi-agent system will synchronize in the stochastic mean square sense. Furthermore, by employing the static protocol, it is shown that when a LOE/float fault occurs under certain conditions the synchronization is still achieved. Finally, one of the main problems that is tackled in the output feedback scenario is our proposed dynamic controller methodology. The results of this scheme are compared with another approach that exploits both dynamic controller and dynamic observer. The former approach has less computational effort and results in more robustness. The reason is that the later method employs an observer that uses the control input matrix information. When fault occurs, this information will not be correct any more. However, if there is a need to redesign the synchronization gains under faulty scenario, the later methodology is preferred. The reason is that the former approach becomes complicated when there is a fault even though its severity is known.

## 1.5 Literature Review

The main concern of the consensus problems is to analyze whether consensus is achieved, or not. The value that the agents synchronize to is called the consensus value which is reached by using relative states information. Consensus problems can be divided into leaderless and leader-follower configurations. In the leaderless platform, a consensus value is through an average consensus or resulting from an exogenous system and it is not specified in advance. In average consensus, the consensus value is weighted average of initial states and the associated weights are determined according to the network's topology. As opposed to leaderless configuration, in the leader-follower multi-agent systems, the consensus trajectory is known to the leader. Hence, the consensus prob-

lem can be formulated as a distributed tracking one.

In the upcoming sections, first a comprehensive literature review on multi-agent systems is provided. This literature review is divided into two main categories corresponding to the complexities of the agent's underlying network and the complexities of the agents' dynamics. In the final section, a through literature review on fault detection and isolation as well as fault-tolerant and reconfigurable synchronization methods is introduced.

### **1.5.1 Synchronization versus Network Topology and Agents' Dynamics Complexities**

Since the formal presentation of the consensus problem in the field of management science and statistics back in 1970's (see [3] and references therein), it has been investigated by many researchers from a variety of fields. Researchers from multi-disciplinary domains including management, statistics, social science [4, 5], biology [6–9], physics [10, 11], power systems [12, 13], robotics and unmanned vehicles have worked on the consensus problem.

In the field of robotics and vehicle systems which is the focus of this thesis, many researchers have worked on cooperative control of unmanned vehicles partly due to its broad application on autonomous underwater vehicles (AUV), unmanned aerial vehicles (UAV), unmanned ground vehicles (UGV), automated highway systems and mobile robots. The concept of cooperative control and consensus have also been applied to different problems including formation control [14–17], flocking [11, 18, 19], distributed sensor networks [20] and congestion control in communication networks [21].

One main motivation for the extensive use of multi-agent systems in unmanned vehicles is their capability in performing tasks which are impossible or really hard to achieve by only one agent. Other interesting applications are being utilized in places which are hazardous to humans. Due to these problems, many researches have been devoted to investigate different aspects of consensus problem and cooperative control.

In the field of cooperative control, Borkar and Varaiya [22] and Tsitsiklis and Athans [23] were the pioneers who dealt with the consensus problem. In their works, they have considered the asynchronous consensus problem. Later in [11], Vicsek *et al.* proposed a simple but interesting discrete-time model for multi-agent systems and they conducted a number of simulations using the nearest neighbor rule in a distributed manner. Vicsek model is a special version of the Reynolds model introduced in [18] which performs flocking in a distributed manner. Both these papers are behavioral and simulation based. It could be said that the paper by Jadbabaei et al. [24] and Olfati and Murry [25] are among the first papers that investigate the consensus problem in a theoretic-

cal framework. In [25], linear and nonlinear consensus protocols are proposed for continuous-time agents in a network of multi-agent systems, and then the proposed protocol is applied to the consensus problem of multi-agent systems with switching topology and communication delay. Both these works ([24] and [25]) are studying the consensus achievement of single integrators and consider the switching topology as well. In [26], Ren and Atkins extended [25] by introducing the second order consensus problem.

The research in the domain of multi-agent system and their use for a variety of applications is on progress. Recently, collision avoidance consensus achievement of unmanned aerial vehicles (UAVs) is considered in [27] and [28]. The consensus of multi-agent systems with binary-valued communication on an undirected graph network with a fixed topology is studied in [29]. Ren *et al.* at [30] investigate the distributed consensus problem of multiple UAV systems with nonlinear uncertainty and bounded disturbance under a directed graph.

The main idea in consensus achievement is that the agents aim to synchronize to a common value. It should be noted that consensus, coordination and synchronization are different terminologies that have been used interchangeably in the literature. However, in consensus, the communication graph's topology and constraints are more emphasized than the agents' dynamics. Some of these constraints and issues are as switching topology, time delay, imperfect communication channel, noise contaminated measurement data, among others. In contrast to consensus, in synchronization more emphasis is on the individual dynamics rather than on the communication constraints. Therefore, different complexities associated with agent's dynamics are considered, and the goal is to reach a common solution of the synchronization dynamics. In this thesis, we use them interchangeably.

For synchronization of general linear time-invariant systems which is the focus of this thesis, there are two approaches in the literature; namely the state feedback synchronization protocol and the output feedback synchronization protocol. When all states are available, the state feedback synchronization approach can be applied through static or dynamic controllers [31–33]. As in the state feedback case, output feedback algorithms have been introduced in the literature to reach synchronization as static [34–36] and dynamic [31], [36] protocols.

The cooperative control problems are categorized as either formation control with application to unmanned vehicles or non-formation problems in cooperative traffic control, role assignment, and timing assignment. Using formation of multiple vehicles to accomplish an objective has several advantages. It not only increases the probability of success, e.g. in a search mission, it could also improve the cost saving and energy efficiency as in the deep space micro-spacecraft application for interferometry application. In this thesis consensus which is subject to formation-based cooperative problem is considered. Although history of consensus problems goes back to computer science, our focus will be on their applications to cooperative control of multi-agent systems.

Since the emergence of the consensus problem, different issues associated with the network topology have been considered. They could be stated as switching topology, switching over a random network, time delay, noise contaminated measurement data, disturbance, fault, among other issues. Each of the above mentioned issues may have its own considerations; for example, switching in a network topology could be state dependent or not, time delay could be synchronous [37] or asynchronous [38] and homogeneous or heterogeneous.

Although consensus problems are simplified by considering time-invariant information exchange topology, in reality they may change. The communication links are subject to change due to disturbances and/or communication link limitations. Motivated by [11], extensive research activities have been conducted in this regard. One approach to tackle switching topologies is to use the algebraic graph theory. Also, nonlinear tools have been used to study consensus of time-varying structures.

As previously mentioned, Jadbabaei and *et al.* in [24] considered a discrete-time model for first-order systems, and solved the consensus problem. In that paper, they took into account the possible changes in the nearest neighbors in time and applied the convergence results of infinite products of certain types of nonnegative matrices (for infinite matrix products, the reader can refer to [39, 40]). Furthermore, they showed that consensus can be achieved if the union of the interaction graphs for the team are connected frequently enough as the system evolves. However, the methodology proposed in that paper was for bidirectional information exchanges. In [25], the consensus achievement of single integrators was considered. In that paper, first balanced digraphs in fixed topology were considered and then the switching topology but still in the framework of balanced digraphs.

Moreau in [41] showed that the consensus is achieved exponentially for first order integrators if the union of the directed interaction graphs have a spanning tree frequently enough as the system evolves. Later Ren and Atkins in [26] extended [25] and [24] by introducing the second order consensus problem and solved the switching topology problem under less restrictive constraints. In that paper in the context of both continuous-time and discrete-time systems, they showed that consensus can be achieved asymptotically by assuming the same conditions in [41] for interaction graphs.

Later in [36], Scardovi *et al.* solved the synchronization problem for LTI systems under a possibly time-varying and directed network topology. The main contribution of that paper is to employ dynamic output feedback to synchronize marginally stable systems under a uniformly connected interaction topology. They solved the same problem using static output feedback under some more restrictive conditions.

In [42], the authors considered the consensus achievement of single-integrators over random graphs in which communications between agents is unidirectional and the existence of each edge is

independent from all other agents at current and previous time steps. They considered undirected networks with a common time delay on all the information exchange links. Later in [43], the authors generalized the work in [42] to random weighted directed graphs where each communication link has a different probability to exist. These two works considered almost surely consensus problem over a Bernoulli network. Moreover, authors in [44] studied the mean square consentability problem for a network of double-integrator agents with stochastic switching topology. In [45], given the same assumptions in [43] the local synchronization of chaotic oscillators was considered but by using state feedback information.

As previously mentioned, time delay is another issue which should be considered in consensus achievement of networked systems. Time delays in networks may arise due to various causes as queuing delays and packet losses because of radio interfere, network congestion and other communication failures.

Due to the fact that time delays are caused by different factors, their values generally vary. Thus, it is important to consider multi-agent systems with heterogeneous time delays. In networked delayed systems, the consensus achievement and the final consensus value depend not only on the network structures, but also on the values of the time delays. It means that time delay and specifically heterogeneous time delays may disturb average consensus achievement.

In [25], Olfati and Murray studied consensus problem of single integrators with undirected and delayed interconnections. Later in [46], the authors extended the results to directed networks with non-uniform delays. Moreover, in [47], the leader-follower consensus problem for multi-agents with directed and time-delayed coupling have been considered. In that paper, the authors have employed Lyapunov Razumikhin functions to deal with the convergence and the stability of consensus problem. In [48], authors have investigated the distributed consensus control of second order integrators under a delay-dependent switching topologies. In [49], the sampled-data consensus based on the delayed-input approach was studied. A sampled-data multi-agent system is converted to an equivalent nonlinear system with a time-varying delay. Zhang and Tian in [50] studied the consensus of a group of linear dynamic agents with a uniform communication delay for both fixed and switching topology and investigated an allowable delay bound.

As stated above, consensus and synchronization problems could be investigated from different aspects. One may consider the complexity of the underlying network or the complexity of the agents' dynamics. Therefore, in the subsequent section the complexity associated with agents' dynamics that is casted as heterogeneous systems will be studied.

### **1.5.2 Heterogeneous Systems and Synchronization**

In a real world scenario, two agents are not identical. Therefore, we deal with non-identical (heterogeneous) systems. To deal with this issue, a few methodologies have been developed in the literature. The first method to deal with this problem was based on internal model principle which need an absolute output measurement [51], [52]. In contrast to these two papers, [53] and [54] used only local information. The proposed method in these works was based on tracking a feasible synchronization trajectory which agents agree on. However, it requires a leader-follower communication structure condition. To be more specific, [53] decomposed the problem into three sub-problems: (i) synchronization of identical exo-systems which provides the synchronization trajectory, (ii) a decentralized observer for estimating the absolute state based on the local measurement, and finally (iii) a control law which uses the information of the state estimation to follow the synchronization trajectory.

In [55], the cooperative output regulation of linear multi-agent systems have been considered, and it is assumed that either the agents have access to the exogenous signal or not. Therefore, the problem cannot be solved in a decentralized manner. However, by using a distributed observer, the problem can be solved.

In all of the above papers, it is assumed that all agents' actuators are able to apply the command signal and all sensors' measurements are correct. However, in real applications it is always possible that one/some agents fail to produce the requested command signal or the control unit does not have the correct measurements due to potential sensor faults. This motivates us to study the fault-tolerant control and fault detection and diagnosis approaches in the literature.

### **1.5.3 Fault-Tolerant Control and Fault Detection and Diagnosis**

Fault-tolerant control and fault detection and diagnosis (FDD) problem of single agent systems has been studied extensively in the literature and different methods have been proposed to deal with it. A control system that can accommodate faults among system components automatically while maintaining the system stability along with a desired level of overall performance is considered as a fault-tolerant control system (FTCS). The fault-tolerant controller design approach and the availability of redundancies in the control system determine the performance achieved in control system. From a practical point of view, the aircraft flight control system was the original motivation behind the fault-tolerant control systems. However, recently due to safety and reliability demands, FTCS has drawn more attention in wider range of industries.

Generally speaking, FTCS can be divided into two categories: passive (PFTCS) and active (AFTCS). In PFTCS, controllers are designed in a way to be robust against a class of presumed faults. Although this approach does not need any FDI block, it has limited fault-tolerant capabil-

ities. In contrast to the PFTCS, in the AFTCS the controller is reconfigured (or only the gains are redesigned) in a way to maintain the overall system stability and acceptable performance. In order to choose between PFTCS and AFTCS, first the effects of presumed faults on the system performance should be investigated. The AFTCS can be divided into four sub-systems as

- A reconfigurable controller
- A FDD scheme
- A controller reconfiguration mechanism
- A command/reference governor

The reconfigurable control design methods are classified into: linear quadratic (LQ) [56], pseudo-inverse (PI) [57], gain scheduling (GS)/linear parameter varying (LPV) [58, 59], model following (MF) [60, 61], adaptive control (AC) [62, 63], multiple model (MM) [64], eigenstructure assignment (EA) [65], feedback linearization (FL)/dynamic inversion (DI) [66],  $H_\infty$  and other robust control methods [67], model predictive control (MPC) [68], quantitative feedback theory (QFT) [69], linear matrix inequality (LMI) [70], variable structure control (VSC)/ sliding mode control (SMC) [71] and generalized internal model control (GIMC) [72].

Among the above methods, LQ, PI, MF, EA, MM, MPC, QFT, and GIMC are only applied to linear systems, while GS/LPV, MM, FL/DI, LMI, and VSC/SMC are applied to nonlinear systems as well. From the point of view of controller restructuring or redesign, the above mathematical design tools can be classified as in Figure 1.1. Figure 1.2 categorizes these control approaches according to the control algorithms classifications.

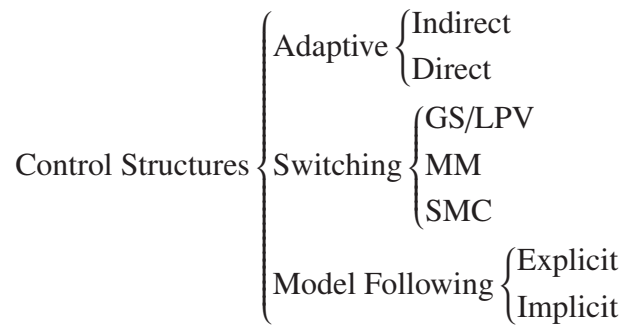


Figure 1.1: Control structures classifications.

As already stated, fault detection and diagnosis (FDD) is one of the key subsystems of AFTCS. Fault diagnosis algorithms are generally categorized into two broad classes: model-based methods and data-driven methods. The data-driven approaches include computationally intelligent methods.

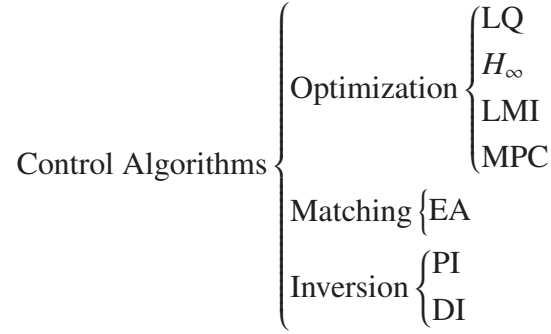


Figure 1.2: Control algorithms classifications.

However, model-based methods are classified into quantitative and qualitative techniques. Figure 1.3 summarizes the model-based FDD approaches with emphasis on quantitative methods.

There are some key features of FDD approaches. Dependent on how a method addresses these features, the FDD approach is opted for a system. These features are:

- Applicability to sensor fault
- Applicability to actuator fault
- Speed of detection
- Isolability
- Identifiability
- Suitability for fault-tolerant control
- Multiple fault identifiability

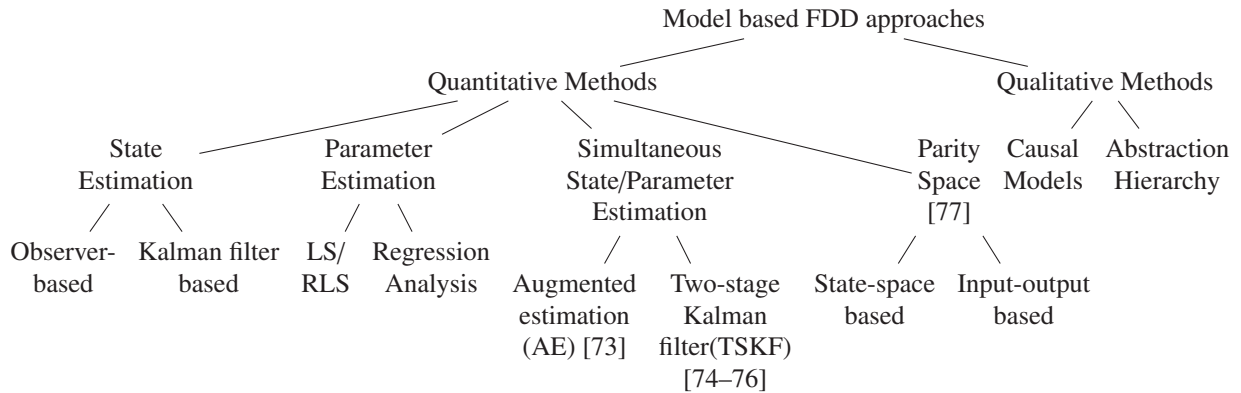


Figure 1.3: Classification of model-based FDD methods.

- Applicability to nonlinear systems
- Robustness
- Computational complexity

Moreover, there are some concerns for AFTCS design beyond the ones in a conventional control system. These challenges are as follows:

- Redundancy as a decision between hardware and analytical redundancy
- Stability, robustness and performance degradation analysis (both transient and steady-state)
- FDD uncertainties and reconfiguration delay
- FDD and FTCS integration
- Real-time issues and networked control system applications
- Safety, reliability and reconfigurability analysis and assessment
- Practical considerations in applications of FTCS

For a complete review of FTCS, an interested reader could refer to the review papers [78] and [79] and the references therein.

### **1.5.4 Fault-Tolerant Control of Multi-agent Systems**

Most of the papers in the literature of consensus achievement in multi-agent systems do not consider the case where the agents are subject to input constraints and faults. However, almost in every physical application the actuator output is bounded and failure can happen in sensors and actuators. Thus, due to the significant importance of safety in vehicle systems, the fault occurrence and its subsequent effect on the system's synchronization and overall performance should be investigated.

Although not fully addressed as single agent systems, there are few works that consider FDD problem for multi-agent systems [80–82]. In [80] and [81], the FDI problem in a team of unmanned vehicles with relative measurements was studied. In these papers three different architectures as centralized, decentralized and semi-decentralized are considered, and the solvability conditions for the FDI problem in these architectures are driven. In [82], a hybrid methodology for fault detection, isolation and recovery (FDIR) of a team of unmanned vehicles was presented. Their suggested methodology has two levels; a low level approach which is based on the classical control methods, and a high level paradigm which is based on discrete-event systems.

In the context of multi-agent systems, references [83–86] are among the first few papers that investigate the effect of potential faults on synchronization. All these three works considered a leader-follower strategy. Semsar and Khorasani in [83] and [84] studied the effects of actuator faults on a team of mobile agents with a modified leader-follower topology based on a semi-decentralized optimal control consensus protocol proposed in [87]. In [85], the effects of agent dynamics faults are compensated by adjusting certain weights of the cooperative protocol. Authors in [86] considered a two-level (low-level and formation level) fault recovery scheme for satellites formation flying. In the low level recovery mechanism, asymptomatic closed loop stability is guaranteed. However, for a biased fault estimation, faulty satellites are partially recovered. This leads to deteriorated formation tracking errors. Therefore, formation level control mechanism is employed to reduce tracking error bound. This means that other satellites allocate more control efforts to compensate for the effect of partially recovered agent.

More recently, the consensus protocol subject to the actuator faults have been investigated in [49] and [88–93]. In [49], the distributed consensus tracking problem of linear higher-order multi-agent systems with occasionally missing control inputs was investigated. The authors in [88] studied the distributed consensus problem of multi-agent systems in the presence of non-identical unknown nonlinear dynamics and undetectable actuation failures and solved the problem with a robust adaptive fault-tolerant control scheme based upon the local agent state information.

In [89], a reconfigurable synchronization protocol was employed such that the control gain is redesigned after fault occurrence to guarantee the consensus achievement with minimum cost in the presence of actuator faults and FDI uncertainty. Using the linear matrix inequality approach, sufficient conditions are derived to show the existence of such a reconfigurable controller.

The authors in [90] proposed a distributed control protocol to guarantee the consensus in the presence of actuator faults and saturations and environmental disturbances. The post-fault controller is redesigned based on the inaccurate information that the fault detection and identification (FDI) module has provided.

In [91], authors employed a virtual actuator technique to compensate the effect of actuator faults. They formulate the consensus problem as a tracking control problem. Moreover, faults are estimated using a sliding mode observer and sufficient conditions are derived for bounded tracking error of all followers. It is to be noted that the tracking errors of healthy individuals converge to zero. However, for faulty agents once accurate fault estimates are available, the tracking errors remain bounded.

Authors in [92] developed a decentralized fault tolerant formation controller for UAVs in leader-follower structure. The proposed solution only needs the FTC mechanism for faulty agent rather than the entire team. As actuator faults occur, a compensation term is added to nominal controller to remove the effect of such permanent faults. Moreover, switching systems approach is

used to analyze the effects of intermittent faults.

Authors in [93] considered the distributed control strategies for the attitude synchronization and set-point tracking control of multiple heterogeneous spacecraft in a formation flying mission. In that paper, the heterogeneous systems synchronization, which may model the faulty scenario, is casted into a tracking problem.

Generally, when a fault occurs in a team of multi-agent systems with identical dynamics, the agents' dynamics become heterogeneous and therefore the conventional approaches for synchronization of homogeneous systems may not be valid any more. However, the methodologies introduced in the class of heterogeneous multi-agents systems might be adjusted and used.

As stated above, consensus and synchronization problems could be investigated from different aspects. One may consider the complexity of the underlying network or the complexity of the agents' dynamics. In addition, fault detection and identification (FDI), and fault-tolerant and reconfigurable synchronization protocols are among the topics that have not been investigated broadly. This motivates us enough to study the consensus problem of multi-agent systems in the presence of faults.

## 1.6 Contributions

The main objective of this thesis is to investigate the synchronization protocols in networked multi-vehicle systems of autonomous underwater vehicles. As mentioned, fault detection and isolation and fault-tolerant control of multi-agent systems have not been already addressed extensively. Therefore, in this thesis we will focus on fault-tolerant control of AUVs. The limited papers in the literature of faulty scenarios cover the integrator systems, or leader-follower architectures. However, this thesis and specifically the following contributions deal with a general marginally stable linear systems and in a leaderless platform which has not been addressed in the literature.

In this thesis, the problems will be addressed in two main categories: (i) state feedback synchronization (ii) output feedback synchronization. The main contributions in these two methodologies could be mentioned as follows:

- In Chapter 3 for state feedback problem, a static approach in the presence of measurement noise is employed where consensus achievement in a stochastic mean square sense (MSS) is developed. Moreover, a dynamic control protocol is employed where the control gains are redesigned after the fault occurrence.
- In Chapter 3 for state feedback problem, a PI controller is proposed to keep consensus in the presence of LIP fault. It is shown that by defining the synchronization states as the error

between the faulty agent and the rest of the agents, the effects of LIP fault is presented as a constant disturbance applied to the new state-space model. Therefore, by employing a PI controller, and based on the internal model principle, the effects of the disturbance could be rejected.

- In Chapter 4 for output feedback problem, a static as well as two different dynamic protocols have been developed. The first dynamic methodology exploits a dynamic observer as well as a dynamic controller; however, the second approach only employs a dynamic controller. The latter methodology is more robust in the presence of faults; and therefore, it is well suited to be used as a passive approach. However, when accurate fault information is available, the former approach is preferred due to the convenience of the controller gain redesign.

## **1.7 Summary**

This chapter is an preliminary introduction to fault-tolerant synchronization of autonomous underwater vehicle systems. First, motivation, problem definition, objective, and the methodologies of the main problems solved in the thesis have been discussed. Following that, the literature review on multi-agent systems and synchronization, and fault detection and isolation as well as fault-tolerant control have been introduced. Finally, the outline of main contributions that are achieved in this thesis are provided.

## Chapter 2

# Background Information and Definitions

This chapter includes the background information on the AUV modeling, algebraic graph theory, matrix theory, and fault modeling. It also contains an introduction of the notations that are used in the thesis. The literature review on AUV modeling starts by providing the equations of motion of a rigid body. Then, the effect of hydrodynamic damping, gravitational and buoyant forces and environmental disturbances are added. Finally, it ends by introducing the AUV's sensors and actuators as well as its propulsion system. The second part of the background information provides a review of the mathematics and the definitions that are used through out the thesis. This includes algebraic graph theory, matrix theory, and fault modeling.

## 2.1 Unmanned Underwater Vehicles

The underwater vehicles are in general classified into two main categories: manned submersibles and unmanned underwater vehicles. All types of underwater robots that are operated without or with minimal human interaction are called unmanned underwater vehicles. It is generally used to describe the remotely operated vehicles (ROV) or autonomous underwater vehicles.

As opposed to ROVs, AUVs operate without constant human monitoring and operation. Due to the fact that AUVs do not have the limitation of umbilical cables in ROVs, they are extensively used for certain types of missions such as long-range oceanographic data collection [94]. AUVs are classified into four main categories: survey AUVs, gliders, micro AUVs, and inspection or hybrid AUVs.

In order to accomplish an exploration task by AUVs (especially in the oil and gas industries), a reasonable approach is to coordinate the vehicles to form a particular configuration or to accomplish a common task. In this case, the most important aim is to move the group of vehicles while

keeping the formation or reaching consensus on a common goal.

There are a couple of challenges in the design of the AUVs control system. The most important ones are as follows:

- Inherent nonlinearity of the underwater vehicle
- Uncertainty of the hydrodynamic parameters
- Limited operational underwater navigation sensors
- Under-actuated systems

Since the only practical way for underwater communication is via acoustic channels, the transmitted data is subject to noise, packet loss, time delays, and the fading power of signals.

Due to all the aforementioned issues, designing a stabilizing feedback controller for an under-actuated system is a challenge in practice and may not be addressed by a smooth static state feedback law [95].

### 2.1.1 AUV Modeling

AUV modeling includes the study of its dynamics and statics. Statics consider the equilibrium of the body at rest or moving with constant velocity whereas the dynamics are concerned with the accelerated motions of the body. It is common to study the dynamics divided as kinematics and kinetics. Kinematics deals with the geometrical aspects of motion, and kinetics analyzes the forces causing the motion. The dynamical behavior of an AUV can be described by using a 6-degrees of freedom (DOF) nonlinear equation of a rigid body.

### 2.1.2 AUV Nonlinear Equations of motion

The AUV's nonlinear equations of motion can be represented both in the body-fixed and the earth-fixed reference frames. The body fixed vectorial representation of the 6-DOF rigid body is as [96]

$$M_{RB}\dot{v} + C_{RB}(v)v = \tau_{RB}, \quad \dot{\eta}_1 = J_1(\eta_2)v_1, \quad \dot{\eta}_2 = J_2(\eta_2)v_2 \quad (2.1)$$

where  $\eta = [\eta_1^T, \eta_2^T]^T$  with  $\eta_1 = [x, y, z]^T$  and  $\eta_2 = [\phi, \theta, \psi]^T$  is the vector of positions and orientations in the earth-fixed frame,  $v = [v_1^T, v_2^T]^T$  with  $v_1 = [u, v, w]^T$  and  $v_2 = [p, q, r]^T$  is the vector of linear velocities and angular rates in the body-fixed frame,  $M_{RB} \in \mathbb{R}^{6 \times 6}$  is the inertia matrix,  $C_{RB} \in \mathbb{R}^{6 \times 6}$  is the Coriolis and centripetal matrix, and  $\tau_{RB} = [X, Y, Z, K, M, N]^T$  is a generalized vector of external forces and moments.

The inertia and Coriolis matrices of  $M_{RB}$  and  $C_{RB}$  are obtained as

$$M_{RB} = \begin{bmatrix} m & 0 & 0 & 0 & mz_G & -my_G \\ 0 & m & 0 & -mz_G & 0 & mx_G \\ 0 & 0 & m & my_G & -mx_G & 0 \\ 0 & -mz_G & my_G & I_x & -I_{xy} & -I_{xz} \\ mz_G & 0 & -mx_G & -I_{yx} & I_y & -I_{yz} \\ -my_G & mx_G & 0 & -I_{zx} & -I_{zy} & I_z \end{bmatrix}$$

and

$$C_{RB} = \begin{bmatrix} 0 & 0 & 0 & m(y_G q + z_G r) \\ 0 & 0 & 0 & -m(y_G p + w) \\ 0 & 0 & 0 & -m(z_G p - v) \\ -m(y_G q + z_G r) & m(y_G p + w) & m(z_G p - v) & 0 \\ m(x_G q - w) & -m(z_G r + x_G p) & m(z_G q + u) & I_{yz} q + I_{xz} p - I_z r \\ m(x_G r + v) & m(y_G r - u) & -m(x_G p + y_G q) & -I_{yz} r - I_{xy} p + I_y q \\ & & & -m(x_G q - w) & -m(x_G r + v) \\ & & & m(z_G r + x_G p) & -m(y_G r - u) \\ & & & -m(z_G q + u) & m(x_G p + y_G q) \\ & & & -I_{yz} q - I_{xz} p + I_z r & I_{yz} r + I_{xy} p - I_y q \\ & & & 0 & -I_{xz} r - I_{xy} q + I_x p \\ & & & I_{xz} r + I_{xy} q - I_x p & 0 \end{bmatrix}$$

where  $M_{RB} = M_{RB}^T > 0$ ,  $\dot{M}_{RB} = 0$  and  $C_{RB}(v) = -C_{RB}^T(v) \forall v \in \mathbb{R}^6$ .

Here,  $m, r_G = [x_G, y_G, z_G]^T$ , and  $I_0 = \begin{bmatrix} I_x & -I_{xy} & -I_{xz} \\ -I_{yx} & I_y & -I_{yz} \\ -I_{zx} & -I_{zy} & I_z \end{bmatrix}$  are the vehicle's mass, center of gravity

and body's inertia tensor, respectively.

The transformation matrices of the kinematics equation are:

$$J_1(\eta_2) = \begin{bmatrix} c(\psi)c(\theta) & -s(\psi)c(\phi) + c(\psi)s(\theta)s(\phi) & s(\psi)s(\phi) + c(\psi)s(\theta)c(\phi) \\ s(\psi)c(\theta) & c(\psi)c(\phi) + s(\psi)s(\theta)s(\phi) & -c(\psi)s(\phi) + s(\psi)s(\theta)c(\phi) \\ -s(\phi) & c(\theta)s(\phi) & c(\theta)c(\phi) \end{bmatrix} \quad (2.2)$$

$$J_2(\eta_2) = \begin{bmatrix} 1 & s(\phi)t(\theta) & c(\phi)t(\theta) \\ 0 & c(\phi) & -s(\phi) \\ 0 & s(\phi)/c(\theta) & c(\phi)/c(\theta) \end{bmatrix} \quad (2.3)$$

where  $s(.) = \sin(.)$ ,  $c(.) = \cos(.)$  and  $t(.) = \tan(.)$ .

The rigid body dynamics can be written in compact form as:

$$\begin{aligned} m[\dot{u} - vr + wq - x_G(q^2 + r^2) + y_G(pq - \dot{r}) + z_G(pr + \dot{q})] &= X \\ m[\dot{v} - wp + ur - y_G(r^2 + p^2) + z_G(qr - \dot{p}) + x_G(qp + \dot{r})] &= Y \\ m[\dot{w} - uq + vp - z_G(p^2 + q^2) + x_G(rp - \dot{q}) + y_G(rq + \dot{p})] &= Z \\ I_x \dot{p} + (I_z - I_y)qr - (\dot{r} + pq)I_{xz} + (r^2 - q^2)I_{yz} + (pr - \dot{q})I_{xy} + m[y_G(\dot{w} - uq + vp) - z_G(\dot{v} - wp + ur)] &= K \\ I_y \dot{q} + (I_x - I_z)rp - (\dot{p} + qr)I_{xy} + (p^2 - r^2)I_{zx} + (qp - \dot{r})I_{yz} + m[z_G(\dot{u} - vr + wq) - x_G(\dot{w} - uq + vp)] &= M \\ I_z \dot{r} + (I_y - I_x)rp - (\dot{q} + rp)I_{yz} + (q^2 - p^2)I_{xy} + (rq - \dot{p})I_{zx} + m[x_G(\dot{v} - wp + ur) - y_G(\dot{u} - vr + wq)] &= N \end{aligned}$$

The equation (2.1) can be described as linear and rotational motions as following:

### Linear Motion

- Surge: a linear longitudinal (front/back) motion.
- Sway: a linear lateral (side-to-side) motion.
- Heave: a linear vertical (up/down) motion.

### Rotation Axes

- Roll: a rotation about longitudinal (front/back) axis.
- Pitch: a rotation about transverse (side-to-side) axis.
- Yaw: a rotation about vertical (up/down) axis.

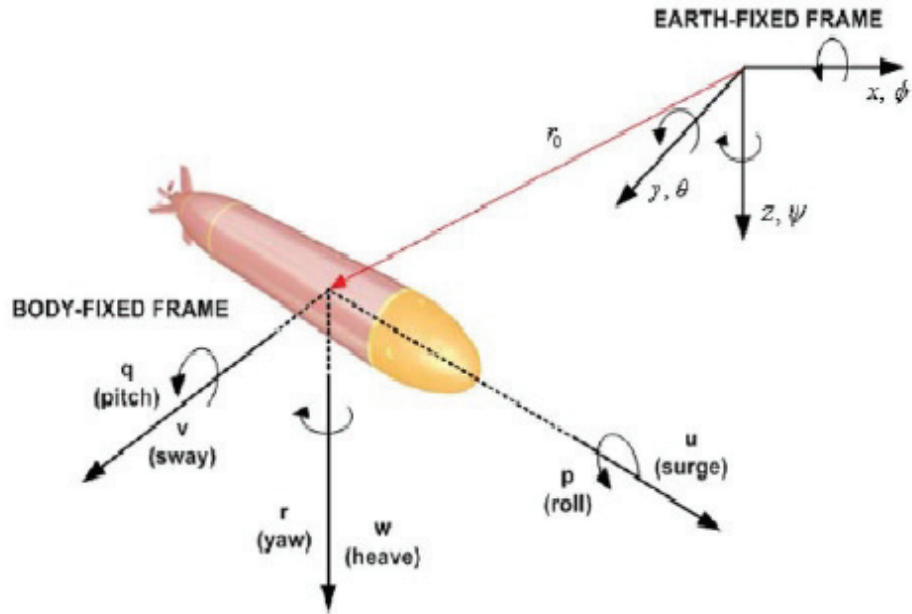


Figure 2.1: Inertial reference and body-fixed frames [1].

Table 2.1 contains a complete list of all notations used in the AUV modeling.

Table 2.1: AUV model notations.

DOF	Motion	Force and Moment	Linear and Angular Velocity	Linear and Angular Position
1	Motion in x-direction (surge)	X	u	x
2	Motion in y-direction (sway)	Y	v	y
3	Motion in z-direction (heave)	Z	w	z
4	Rotation about x-axes (roll)	K	p	$\phi$
5	Rotation about y-axes (pitch)	M	q	$\theta$
6	Rotation about z-axes (yaw)	N	r	$\psi$

## 6-DOF Simplified Rigid-Body Equations of Motion

The equation (2.1) can be simplified by choosing the origin of the body-fixed coordinate system as

- Origin O of body-fixed coordinate coincides with the principal axes of inertia and therefore the center of gravity becomes  $r_G = [0 \ 0 \ 0]^T$ .
- Origin O and the body-fixed frames are chosen such that  $I_0$  is diagonal.

### 2.1.3 Hydrodynamic Forces and Moments

By considering the added mass and inertia terms, the 6-DOF equations of motion become

$$M\dot{v} + C(v)v + D(v)v + g(\eta) = \tau + \tau_E, \quad (2.4)$$

where  $M \triangleq M_{RB} + M_A$ ,  $C(v) \triangleq C_{RB}(v) + C_A(v)$  and  $D(v)$ ,  $g(\eta)$  and  $\tau_E$  are used to describe the hydrodynamic damping, restoring and environmental forces and moments acting on the vehicle, respectively. For AUVs, the constant zero frequency added mass terms are:

$$M_A = M_A^T = - \begin{bmatrix} X_{\ddot{u}} & X_{\ddot{v}} & X_{\ddot{w}} & X_{\dot{p}} & X_{\dot{q}} & X_{\dot{r}} \\ Y_{\ddot{u}} & Y_{\ddot{v}} & Y_{\ddot{w}} & Y_{\dot{p}} & Y_{\dot{q}} & Y_{\dot{r}} \\ Z_{\ddot{u}} & Z_{\ddot{v}} & Z_{\ddot{w}} & Z_{\dot{p}} & Z_{\dot{q}} & Z_{\dot{r}} \\ K_{\ddot{u}} & K_{\ddot{v}} & K_{\ddot{w}} & K_{\dot{p}} & K_{\dot{q}} & K_{\dot{r}} \\ M_{\ddot{u}} & M_{\ddot{v}} & M_{\ddot{w}} & M_{\dot{p}} & M_{\dot{q}} & M_{\dot{r}} \\ N_{\ddot{u}} & N_{\ddot{v}} & N_{\ddot{w}} & N_{\dot{p}} & N_{\dot{q}} & N_{\dot{r}} \end{bmatrix}$$

$$C_A(v) = -C_A(v)^T = \begin{bmatrix} 0 & 0 & 0 & 0 & -a_3 & a_2 \\ 0 & 0 & 0 & a_3 & 0 & -a_1 \\ 0 & 0 & 0 & -a_2 & a_1 & 0 \\ 0 & -a_3 & a_2 & 0 & -b_3 & b_2 \\ a_3 & 0 & -a_1 & b_3 & 0 & -b_1 \\ -a_2 & a_1 & 0 & -b_2 & b_1 & 0 \end{bmatrix} \quad (2.5)$$

where

$$\begin{aligned}
a_1 &= X_{\dot{u}}u + X_{\dot{v}}v + X_{\dot{w}}w + X_{\dot{p}}p + X_{\dot{q}}q + X_{\dot{r}}r \\
a_2 &= Y_{\dot{u}}u + Y_{\dot{v}}v + Y_{\dot{w}}w + Y_{\dot{p}}p + Y_{\dot{q}}q + Y_{\dot{r}}r \\
a_3 &= Z_{\dot{u}}u + Z_{\dot{v}}v + Z_{\dot{w}}w + Z_{\dot{p}}p + Z_{\dot{q}}q + Z_{\dot{r}}r \\
b_1 &= K_{\dot{u}}u + K_{\dot{v}}v + K_{\dot{w}}w + K_{\dot{p}}p + K_{\dot{q}}q + K_{\dot{r}}r \\
b_2 &= M_{\dot{u}}u + M_{\dot{v}}v + M_{\dot{w}}w + M_{\dot{p}}p + M_{\dot{q}}q + M_{\dot{r}}r \\
b_3 &= N_{\dot{u}}u + N_{\dot{v}}v + N_{\dot{w}}w + N_{\dot{p}}p + N_{\dot{q}}q + N_{\dot{r}}r
\end{aligned}$$

For a rigid body at rest ( $U \approx 0$ ), under the assumptions of an ideal fluid (no sea current, no incident waves, and frequency independent), the added inertia matrix  $M_A$  and Coriolis and centripetal matrix  $C_A$  satisfy the following properties:

$$M_A = M_A^T > 0$$

$$C_A = -C_A^T$$

Thus, assuming a 6-DOF motion at high speed, the underwater vehicle will have a highly nonlinear and coupled behavior. However, in many applications it moves at a very low speed. If the vehicle also has three planes of symmetry, the contribution of off-diagonal elements in the added mass matrix could be neglected. Therefore,  $M_A$  and  $C_A$  are:

$$M_A = -diag\{X_{\dot{u}}, Y_{\dot{v}}, Z_{\dot{w}}, K_{\dot{p}}, M_{\dot{q}}, N_{\dot{r}}\}$$

$$C_A(v) = \begin{bmatrix} 0 & 0 & 0 & 0 & -Z_{\dot{w}}w & -Y_{\dot{v}}v \\ 0 & 0 & 0 & -Z_{\dot{w}}w & 0 & -X_{\dot{u}}u \\ 0 & 0 & 0 & -Y_{\dot{v}}v & -X_{\dot{u}}u & 0 \\ 0 & -Z_{\dot{w}}w & -Y_{\dot{v}}v & 0 & -N_{\dot{r}}r & -M_{\dot{q}}q \\ -Z_{\dot{w}}w & 0 & -X_{\dot{u}}u & -N_{\dot{r}}r & 0 & -K_{\dot{p}}p \\ -Y_{\dot{v}}v & -X_{\dot{u}}u & 0 & -M_{\dot{q}}q & -K_{\dot{p}}p & 0 \end{bmatrix}$$

For instance, the hydrodynamic added mass force  $Y_A$  along the y-axis due to an acceleration  $\dot{v}$  in the y-direction is written as

$$Y_A = Y_{\dot{v}}\dot{v} \quad \text{and} \quad Y_{\dot{v}} \triangleq \frac{\partial Y}{\partial \dot{v}}$$

### 2.1.4 Hydrodynamic Damping

The hydrodynamic damping for ocean vehicles is mainly caused by:

$D_P(v)$ = radiation induced potential damping due to forced body oscillations.

$D_S(v)$ = linear skin friction due to laminar boundary layers and quadratic skin friction due to turbulent boundary layers.

$D_W(v)$ = Wave drift damping.

$D_M(v)$ = damping due to vortex shedding.

The total hydrodynamic damping matrix can be written as a sum of these components as

$$D(v) \triangleq D_P(v) + D_S(v) + D_W(v) + D_M(v)$$

For a rigid body moving through an ideal fluid, the hydrodynamic damping matrix can be written as sum of a linear damping term  $D$  and a nonlinear damping term  $D_n(v)$  such that

$$D(v) = D + D_n(v)$$

where

$$D = D^T = - \begin{bmatrix} X_u & X_v & X_w & X_p & X_q & X_r \\ Y_u & Y_v & Y_w & Y_p & Y_q & Y_r \\ Z_u & Z_v & Z_w & Z_p & Z_q & Z_r \\ K_u & K_v & K_w & K_p & K_q & K_r \\ M_u & M_v & M_w & M_p & M_q & M_r \\ N_u & N_v & N_w & N_p & N_q & N_r \end{bmatrix} \quad (2.6)$$

Assuming a 6-DOF motion at high speed, the underwater vehicle will have a highly nonlinear and coupled behavior. However, assuming that the vehicle is performing a non-coupled motion and has three planes of symmetry, the higher than second order terms are negligible. Therefore, the hydrodynamic damping  $D(v)$  is:

$$\begin{aligned} D(v) = & -diag\{X_u, Y_v, Z_w, K_p, M_q, N_r\} \\ & - diag\{X_{u|u}|u|, Y_{v|v}|v|, Z_{w|w}|w|, K_{p|p}|p|, M_{q|q}|q|, N_{r|r}|r|\} \end{aligned}$$

### 2.1.5 Restoring Forces and Moments

The gravitational and buoyant forces are called restoring forces. The gravitational force  $W = mg$  and buoyant force  $B = \rho g \Delta$  are acting on the center of gravity  $r_G = [x_G \ y_G \ z_G]^T$  and the center of buoyancy  $r_B = [x_B \ y_B \ z_B]^T$ , respectively. The parameters  $m$ ,  $g$ ,  $\rho$  and  $\Delta$  are defined as

- $m$ : The vehicle's mass including water in free floating spaces
- $\Delta$ : The volume of fluid displaced by the vehicle
- $\rho$ : The fluid density
- $g$ : The acceleration of gravity

By transforming the weight and buoyancy forces on to the body-fixed coordinate system, we have

$$f_G(\eta_2) = J_1^{-1}((\eta_2)) \begin{bmatrix} 0 \\ 0 \\ W \end{bmatrix}$$

$$f_B(\eta_2) = -J_1^{-1}((\eta_2)) \begin{bmatrix} 0 \\ 0 \\ B \end{bmatrix}$$

Therefore, the vector of restoring forces and momentums in the body-fixed coordinate system are

$$g(\eta) = - \begin{bmatrix} f_G(\eta) + f_B(\eta) \\ r_G \times f_G(\eta) + r_B f_B(\eta) \end{bmatrix} = \begin{bmatrix} (W - B)s\theta \\ -(W - B)c\theta s\phi \\ -(W - B)c\theta s\phi \\ -(y_G W - y_B B)c\theta c\phi + (z_G W - z_B B)c\theta s\phi \\ -(z_G W - z_B B)s\theta + (x_G W - x_B B)c\theta c\phi \\ -(x_G W - x_B B)c\theta s\phi + (y_G W - y_B B)s\theta \end{bmatrix}$$

For a neutrally buoyant underwater vehicle, we will have  $W = B$ . Therefore, the vector of restoring forces and momentums are simplified as

$$g(\eta) = \begin{bmatrix} 0 \\ 0 \\ 0 \\ -(y_G - y_B)Wc\theta c\phi + (z_G - z_B)Wc\theta s\phi \\ -(z_G - z_B)Ws\theta + (x_G - x_B)Wc\theta c\phi \\ -(x_G - x_B)Wc\theta s\phi + (y_G - y_B)Ws\theta \end{bmatrix}$$

### 2.1.6 AUV's Model Uncertainties

The dynamics of an underwater vehicle is described by highly nonlinear high-order systems with uncertain models and disturbances that are difficult to model and measure. AUV does not pos-

sess hydrodynamically shaped profiles and the hydrodynamic forces are uncertain and difficult to predict. This leads to high uncertainty in the AUV model. There is an extensive research in the literature that addresses the control of AUVs with uncertain dynamics (and in general uncertain systems). To list some, adaptive and robust control approaches could be mentioned.

### 2.1.7 Environmental Disturbances and Stochastic Differential Equation Model

In general, three kinds of disturbances that could be considered for a marine vehicle are:

- Waves (wind generated)
- Wind
- Ocean Currents

These disturbances could have an affect on the dynamics of motion in an additive or multiplicative manner. However, in this thesis we will assume that it is additive. This is a valid assumption for most marine control applications.

Wave-induced disturbances can usually be neglected for a deeply submerged vessel. Hence, the only environmental disturbance will be ocean currents. These ocean currents are horizontal and vertical circulating systems of ocean waters as a result of gravity, wind friction and water density variation in different parts of the ocean.

Let the earth-fixed ocean current velocity vector be denoted by  $[u_c^E, v_c^E, w_c^E]$ . Then, the body-fixed current velocity vector is

$$v_1^c = \begin{bmatrix} u_c \\ v_c \\ w_c \end{bmatrix} = J_1^T(\eta_2) \begin{bmatrix} u_c^E \\ v_c^E \\ w_c^E \end{bmatrix}$$

Therefore, considering the environmental disturbances, the nonlinear equations of motion are

$$\begin{aligned} M\dot{v}_r + C(v_r)v_r + D(v_r)v_r + g(\eta) &= \tau \\ \dot{\eta} &= J(\eta)v \end{aligned} \tag{2.7}$$

where,  $v_r = v - v^c$ , and  $v^c = [v_1^c, 0, 0, 0]^T$ .

The earth-fixed fluid velocity components  $(u_c^E, v_c^E, w_c^E)$  can be related to the average current velocity  $V_c$  by defining two angles of  $\alpha$  (angle of attack) and  $\beta$  (sideslip angle). In other words, the current speed  $V_c$  is defined in the earth-fixed referencing frame using flow axes as  $[V_c, 0, 0]^T$ .

Therefore, 3-dimensional current velocities are found by performing two principal rotations as

$$\begin{aligned}u_c^E &= V_c \cos\alpha \cos\beta \\v_c^E &= V_c \sin\beta \\w_c^E &= V_c \sin\alpha \cos\beta\end{aligned}$$

For computer simulation purpose, the average current velocity is mostly generated by using a first-order Gauss-Markov process as

$$\dot{V}_c(t) + \mu_0 V_c(t) = w(t) \quad (2.8)$$

where  $w(t)$  is a zero-mean white Gaussian noise, and  $\mu_0 > 0$ . When  $\mu_0 = 0$ ,  $V_c(t)$  corresponds to a random walk which is a time integration of white noise.

Finally, using Equations (2.7) and (2.8), the lump sum effect of disturbance could be modeled as

$$\begin{aligned}M\dot{v} + C(v)v + D(v)v + g(\eta) &= \tau + w^c(t) \\ \dot{\eta} &= J(\eta)v\end{aligned} \quad (2.9)$$

where  $w^c(t)$  is a white noise vector.

### 2.1.8 Linear Time-Varying ROV Equations of Motion

The linear equations of motion are obtained by linearization of (2.4) about a time-varying reference trajectory or an equilibrium point. In other words, by defining the new state and input variables as

$$\begin{aligned}\Delta v(t) &= v(t) - v_0(t) \\\Delta \eta(t) &= \eta(t) - \eta_0(t) \\\Delta \tau(t) &= \tau(t) - \tau_0(t)\end{aligned}$$

where

$$\begin{aligned}v_0(t) &= [u_0(t), v_0(t), w_0(t), p_0(t), q_0(t), r_0(t)]^T \\ \eta_0(t) &= [x_0(t), y_0(t), z_0(t), \phi_0(t), \theta_0(t), \psi_0(t)]^T\end{aligned}$$

and defining  $f_c(v) = C(v)v$  and  $f_d(v) = D(v)v$ , the linearized equations could be obtained as,

$$\begin{aligned} M\dot{x}_1 + C(t)x_1 + D(t)x_1 + G(t)x_2 &= \Delta\tau \\ \dot{x}_2 &= J(t)x_1 + J^*(t)x_2 \end{aligned}$$

where

$$\begin{aligned} C(t) &= \left. \frac{\partial f_c(v)}{\partial v} \right|_{v_0(t)} & D(t) &= \left. \frac{\partial f_d(v)}{\partial v} \right|_{v_0(t)} & G(t) &= \left. \frac{\partial g(\eta)}{\partial \eta} \right|_{\eta_0(t)} \\ J(t) &= J(\eta_0(t)) & J^*(t) &= \left. \frac{\partial J(\eta)}{\partial \eta} \right|_{\eta_0(t)} & v_0(t) &= J^*(v_0(t), \eta_0(t)) \end{aligned}$$

and  $x_1 = \Delta v$ ,  $x_2 = \Delta \eta$ ,  $J(\eta) = \begin{bmatrix} J_1(\eta_2) & 0 \\ 0 & J_2(\eta_2) \end{bmatrix}$  and  $J_1(\eta_2)$  and  $J_2(\eta_2)$  are defined in (2.2) and (2.3), respectively.

Finally, defining  $x = [x_1^T, x_2^T]^T$  and  $u = \Delta\tau$ , we have

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} -M^{-1}[C(t) + D(t)] & -M^{-1}G(t) \\ J(t) & J^*(t) \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} M^{-1} \\ 0 \end{bmatrix} u \quad (2.10)$$

In many applications including the one in this thesis, it is valid to assume the AUV has non-zero velocities along  $x$  and  $z$  axes (non-zero  $u_0$  and  $w_0$ ). Moreover, in steady-state the other linear and angular velocities will be  $v_0 = p_0 = q_0 = r_0 = 0$ , and the equilibrium point is defined by  $\phi = \theta = 0$ . Therefore, the time-varying matrices in (2.10) simplify to constant matrices.

## 2.1.9 Reduced-order Model of AUV

As most marine crafts do not have motions along or rotations around all their axes, the reduced-order models are often used to design a feedback control system. The most commonly used reduced-order models in the literature [96] are as:

### 1DOF

These models are used to design a forward speed controller (surge control).

### 3DOF

3-DOF models are usually as

- Horizontal plane models (surge, sway and yaw control) .
- Longitudinal (Vertical) plane models (surge, heave and pitch control) for forward speed, diving and pitch control.

- Lateral models (sway, roll and yaw control) for turning and heading control.

#### 4DOF

This is obtained by adding the roll equation to 3-DOF horizontal plane model (surge, sway, yaw and roll).

#### Longitudinal Model of AUV

In many applications the 6-DOF equations of motion can be divided into two non-interacting (or lightly interacting) models as

- Longitudinal subsystem: with states as  $u, w, q$  and  $x, z, \theta$
- Lateral subsystem: with states as  $v, p, r$  and  $y, \phi, \psi$

This is a valid assumption for slender submarine systems with large length over width ratio [97]. It is a common assumption for a submarine.

In this thesis, we will design a control system for the decoupled model of pitch and heave motions. The design will be conducted using a linearized model of AUV [97] as

$$\begin{aligned}
 & \begin{bmatrix} m - X_{\dot{u}} & -X_{\dot{w}} & mz_g - X_{\dot{q}} & 0 & 0 \\ -X_{\dot{w}} & m - Z_{\dot{w}} & mx_g - Z_{\dot{q}} & 0 & 0 \\ mz_g - X_{\dot{q}} & mx_g - Z_{\dot{q}} & I_y - M_{\dot{q}} & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \dot{u} \\ \dot{w} \\ \dot{q} \\ \dot{\theta} \\ \dot{z} \end{bmatrix} + \\
 & \begin{bmatrix} -X_u & -X_w & -X_q & 0 & 0 \\ -Z_u & -Z_w & mu_0 - Z_q & 0 & 0 \\ -M_u & -M_w & mx_G u_0 - M_q & (z_G - z_B)W & 0 \\ 0 & 0 & -1 & 0 & 0 \\ s\theta_0 & -c\theta_0 & 0 & s\theta_0 w_0 + c\theta_0 u_0 & 0 \end{bmatrix} \begin{bmatrix} u \\ w \\ q \\ \theta \\ z \end{bmatrix} = \begin{bmatrix} X \\ Z \\ M \\ 0 \\ 0 \end{bmatrix} \quad (2.11)
 \end{aligned}$$

A further reduction of the model order could be achieved by assuming zero pitch angle ( $\theta_0 = 0$ ) and constant surge speed ( $u_0 = \text{constant}$ ) at steady state. Therefore, we have

$$\begin{bmatrix} m - Z_{\dot{w}} & mx_g - Z_{\dot{q}} & 0 & 0 \\ mx_g - Z_{\dot{q}} & I_y - M_{\dot{q}} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \dot{w} \\ \dot{q} \\ \dot{\theta} \\ \dot{z} \end{bmatrix} + \begin{bmatrix} -Z_w & mu_0 - Z_q & 0 & 0 \\ -M_w & mx_G u_0 - M_q & (z_G - z_B)W & 0 \\ 0 & -1 & 0 & 0 \\ -1 & 0 & u_0 & 0 \end{bmatrix} \begin{bmatrix} w \\ q \\ \theta \\ z \end{bmatrix} = \begin{bmatrix} Z \\ M \\ 0 \\ 0 \end{bmatrix} \quad (2.12)$$

### 2.1.10 Sensors and Actuators in an Unmanned Underwater Vehicle

Autonomous underwater vehicles are used for deep-water ocean applications where accurate control of vehicle position (mainly depth) and attitude is needed. This accurate control requires precise measurement of AUV's position and attitude.

The states of a longitudinal model of an AUV are mainly heave, pitch, and their associated rates. However, in practice it is always impossible or too costly to measure the full-state vector of AUV. In the case of the INFANTE AUV, which is used in this thesis for simulation scenarios, it is difficult to measure the heave rate, the angle of side-slip and the angle of attack in the horizontal and vertical planes, respectively. However, it is crucial to achieve stabilization along all directions and about all rotations. This motivates to use algorithms that employ only output variables. Throughout the thesis, when referring to the partial state measurements, the measurement matrix  $C$  is denoted by

$$C = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.13)$$

The water level of ballast tank is manually adjusted for neutral buoyancy before each mission while the vehicle is on the surface. Once it is submerged, it would be very difficult to continuously control the desired depth and trajectory by taking air in and out of the ballast tanks. As a result, AUVs are equipped with hydroplanes. Dependant on the mission, different hydroplanes may be used to meet the control objective. The most commonly used ones are stern and sail (bow) planes, and a rudder.

The propeller thrust is the force applied to a submarine which enables the vehicle's forward speed and lift. This thrust is produced due to the forces created by the propeller wings. The blades of propellers can be configured as two blades, three blades and four blades. Propellers produce the forward speed by pushing the fluid back. The speed of rotation of propellers and their pitch angle determine the amount of displaced fluid, and consequently the vehicle forward speed. The operating power in the AUV comes mainly from a fuel cell battery.

By distributing the heavier weights at the bottom of AUV, it keeps the right side up. Moreover, the submarines balance is kept by the stern and bow planes. The servo motors as control surfaces' drivers and a DC motor as propeller's driver are contained inside a water-tight container (WTC). The location of WTC is shown in Fig. 2.4.

The propeller and the control surfaces of an AUV have been shown in detail in Figures 2.2 and 2.3. Rudder and stern (bow) planes enable the submarine to control yaw and pitch motions, respectively. Rotation of stern planes changes the angle of the entire submarine that directly enables the depth control. It is clear when there is no vehicle forward velocity, the control surfaces will not have any effect.

It is assumed that INFANTE AUV, which is used for the simulations in this thesis, is controlled to have a constant forward speed using the propeller controller. Hence, the vehicle's forward speed will not be a control variable. This vehicle also uses only stern and sail(bow) hydroplanes which control vehicle pitch angle and consequently its depth.

Fault could occur at any component of the above INFANTE AUV system including the thruster that may result in a non-constant forward speed. The main reasons for thruster faults are thruster blocking and flooded thruster. Thruster blocking is the result of a solid body between propellers. While flooded thruster is because of flooded water which may result in the force higher than the desired one. In this thesis, no propeller fault is assumed.

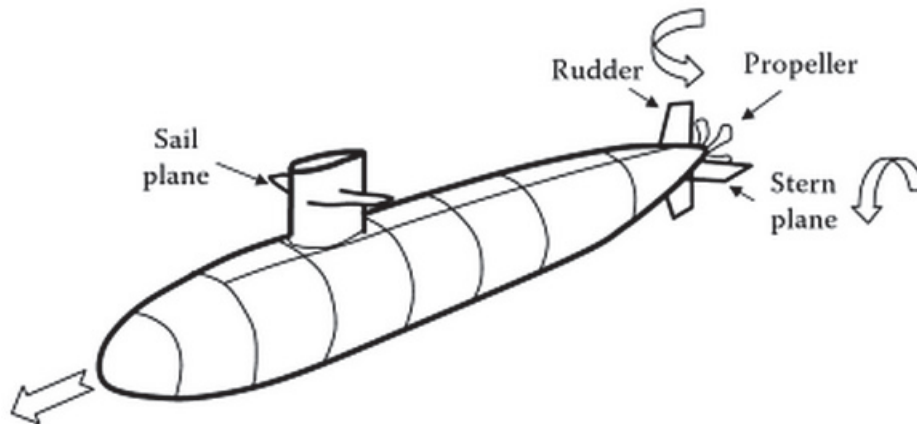


Figure 2.2: Submarine actuators [2].

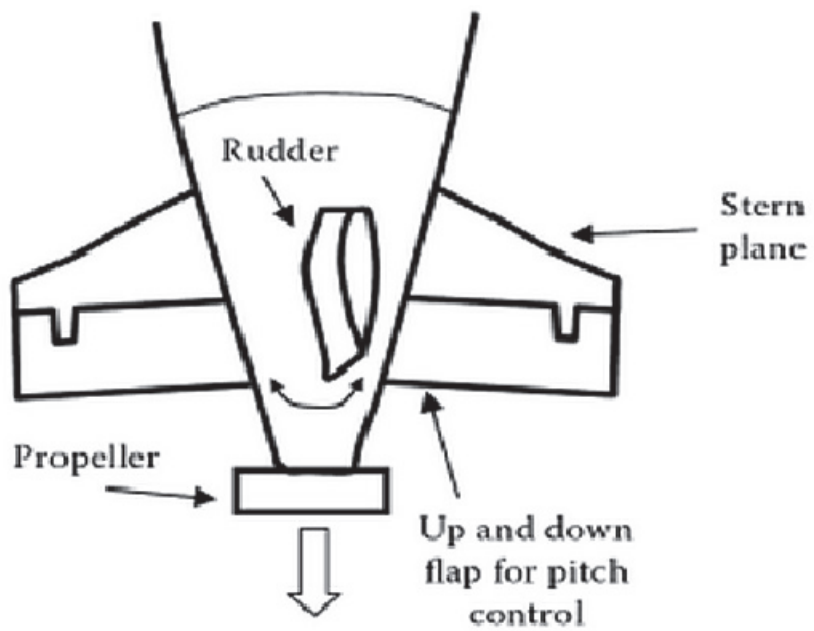


Figure 2.3: Submarine rudder and sterns [2].

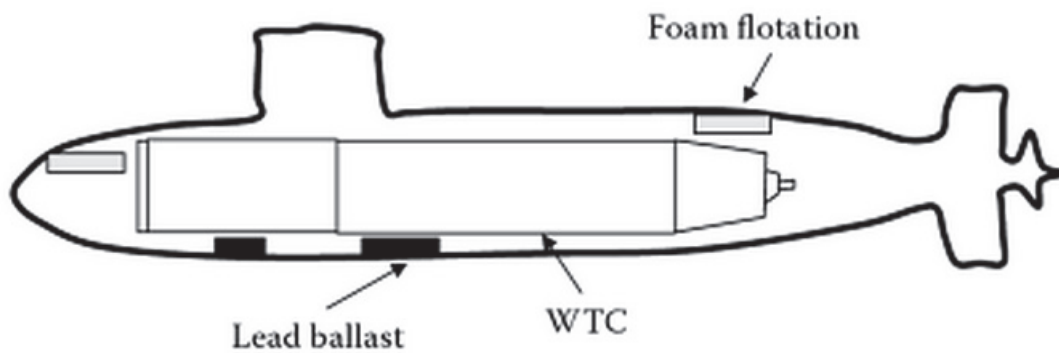


Figure 2.4: WTC location and weight distribution [2].

As mentioned earlier, in this thesis the synchronization of AUVs for offshore oil application will be studied. Hence, the synchronization of multi-agent systems and the mathematics behind should be profoundly investigated. In this regard, the following section includes a brief introduction to the graph theory.

## 2.2 Algebraic Graph Theory Background

In order to model the interaction of a team of agents, a directed or an undirected graph may be used. This interaction can be through a communication or sensing network or a combination of both. Let  $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$  be a weighted graph (digraph) of order  $N$  with nodes' set of  $\mathcal{V} = \{v_1, v_2, \dots, v_N\}$ , and edges' set of  $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ . A directed edge of  $\mathcal{G}$  is denoted by  $e_{ij} = (v_i, v_j)$ . For a digraph,  $e_{ij} \in \mathcal{E}$  does not mean  $e_{ji} \in \mathcal{E}$ . The neighboring set of node  $i$  is the set of all nodes that communicate with node  $i$ . The neighboring set is denoted by  $\mathcal{N}_i = \{v_j \in \mathcal{V} : (v_j, v_i) \in \mathcal{E}\}$ . For edge  $e_{ij}$ ,  $i$  is the parent node and  $j$  is the child node. The union of a collection of graphs is a graph whose node and edge sets are the unions of the node and edge sets of the graphs in the collection.

A graph (digraph) can be used to model the interaction topology among a group of agents, where every node corresponds to an agent and an edge  $e_{ij}$  represents the information exchange link from  $v_i$  (parent) to  $v_j$  (child). Since the information exchange among the agents may vary dynamically, generally the interaction graph is time-dependent. The set of possible interaction graphs defined for a group of agents is a finite set.

**Definition 2.1.** A directed path in graph  $\mathcal{G}$  is a sequence of edges  $e_{i_1 i_2}, e_{i_2 i_3}, e_{i_3 i_4} \dots$  in that graph.

**Definition 2.2.** A graph  $\mathcal{G}$  is called strongly connected if there is a directed path from  $v_i$  to  $v_j$  and  $v_j$  to  $v_i$  between any pair of distinct vertices  $v_i$  and  $v_j$  [98]. In addition,  $\mathcal{G}$  is strongly connected if and only if the Laplacian matrix  $L$  is irreducible [99].

**Definition 2.3.** A tree is a graph in which every pair of nodes is connected by exactly one undirected path.

**Definition 2.4.** A directed tree is a directed graph where every node except the root has exactly one parent.

**Definition 2.5.** A spanning tree of a directed graph is a tree formed by graph edges that connect all the vertices of the graph.

**Definition 2.6.** A directed tree is a directed graph, where every node except the root has exactly one parent. A spanning tree of a directed graph is a directed tree formed by graph edges that connect all the nodes of the graph [100].

The weighted adjacency matrix  $A = [a_{ij}]$  of a directed graph  $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$  is defined such that  $a_{ij}$  is positive if  $e_{ij} \in \mathcal{E}$ , and  $a_{ij} = 0$ , otherwise. A weighted graph associates a weight with

every edge in the graph. The *in-degree* and *out-degree* of node  $i$  are defined as  $\sum_{j=1}^N a_{ij}$  and  $\sum_{j=1}^N a_{ji}$ , respectively. A node  $i$  is balanced if its *in-degree* and *out-degree* are the same. The graph is balanced if  $\sum_{j=1}^N a_{ij} = \sum_{j=1}^N a_{ji}$ . Therefore, every undirected graph is symmetric.

The Laplacian matrix of a graph  $L = [l_{ij}]$  is a zero row sum matrix, with  $l_{ii} = \sum_{j \in \mathcal{N}_i} a_{ij}$ , and  $l_{ij} = -a_{ij}$  for  $i \neq j$ . Note that  $L$  can be defined as  $L = D - A$ , where  $D$  is a diagonal matrix composed of *in-degrees* of the nodes. The eigenvalues of  $L$  ( $\lambda_i$ 's) are set in an ascending order for  $i = 1, 2, \dots, N$  for a connected graph where, except for  $\lambda_1$  which is zero, all the other eigenvalues have positive real parts. Since  $L$  has zero row sums, 0 is an eigenvalue of  $L$  with an associated eigenvector of  $\mathbf{1}_N$ . Moreover, according to Gershgorin disk theorem, since  $L$  is diagonally dominant with nonnegative diagonal entries, all nonzero eigenvalues of  $L$  are in a closed right half-plane. Therefore, for an undirected graph, which has a symmetric Laplacian matrix, all nonzero eigenvalues of  $L$  are positive. In addition,  $L\mathbf{1}_N = \mathbf{0}$  and there exists a nonnegative vector  $p \in \mathbb{R}^N$  such that  $p^T L = \mathbf{0}$  and  $p^T \mathbf{1}_N = 1$ .

**Lemma 2.1.** *The Laplacian matrix of a graph has a single zero eigenvalue if and only if the graph has a spanning tree.*

*Proof.* See the proof of Theorem 2 in [101]. □

**Definition 2.7.** *The graph  $\mathcal{G}(t)$  is said to be uniformly connected if there exists a time horizon  $T > 0$  and an index  $k$  such that for all  $t$ , all the nodes  $v_j (j \neq k)$  are connected to node  $v_k$  across  $[t, t + T]$ .*

**Lemma 2.2.** *Consider the linear system,*

$$\dot{\mathbf{x}}_i = A\mathbf{x}_i + \mathbf{u}_i \quad i = 1, \dots, N \quad (2.14)$$

where all eigenvalues of  $A$  belong to the imaginary axis. Assume that the communication graph is uniformly connected and the corresponding Laplacian matrix  $L(t)$  is piecewise continuous and bounded. Then, the control law of equation (2.15) uniformly exponentially synchronizes all the solutions of (2.14) to a solution of the system  $\dot{\mathbf{x}}_0 = A\mathbf{x}_0$ .

$$\mathbf{u}_i = \sum_{j \in \mathcal{N}_i} a_{ij}(t)(\mathbf{x}_j - \mathbf{x}_i) \quad i = 1, \dots, N \quad (2.15)$$

*Proof.* Refer to Lemma 1 of [36]. As in the proof of this Lemma, the result is still valid if some of the eigenvalues of  $A$  have negative real parts. This means that exponentially stable modes synchronize to zero.  $\square$

## 2.3 Algebra and Matrix Theory Background

In this section, the following definitions and lemmas from the literature in algebra and matrix theory are introduced.

**Definition 2.8.** Matrix  $B \in \mathbb{R}^{n \times n}$  is said to be similar to matrix  $A \in \mathbb{R}^{n \times n}$  if there exist a nonsingular matrix  $S \in \mathbb{R}^{n \times n}$  such that

$$B = S^{-1}AS \quad (2.16)$$

**Theorem 2.1.** Let  $A, B \in \mathbb{R}^{n \times n}$ . If  $B$  is similar to  $A$  then the characteristic polynomial of  $B$  is the same as  $A$ . Therefore,  $A$  and  $B$  have the same eigenvalues.

The Kronecker product of two arbitrary matrices  $A \in \mathbb{R}^{m \times n}$  and  $B \in \mathbb{R}^{p \times q}$  is defined as

$$A \otimes B = \begin{bmatrix} a_{11}B & \cdots & a_{1n}B \\ \vdots & \ddots & \vdots \\ a_{m1}B & \cdots & a_{mn}B \end{bmatrix}$$

which satisfies the following properties for any  $A, B, C$  and  $D$  with appropriate dimensions [102],

$$(A \otimes B)(C \otimes D) = (AC) \otimes (BD)$$

$$A \otimes B + A \otimes C = A \otimes (B + C)$$

$$\exp(A \otimes B) = \exp(A) \otimes \exp(B)$$

For a square block matrix, one can obtain an equivalent expression for its inverse as

$$\begin{bmatrix} A & B \\ C & D \end{bmatrix}^{-1} = \begin{bmatrix} K & L \\ M & N \end{bmatrix} \quad (2.17)$$

where

$$\begin{aligned}
K &= (A - BD^{-1}C)^{-1} \\
L &= -(A - BD^{-1}C)^{-1}BD^{-1} \\
M &= -D^{-1}C(A - BD^{-1}C)^{-1} \\
N &= (D - CA^{-1}B)^{-1}
\end{aligned} \tag{2.18}$$

### Quadratic Stabilizability

Consider the following LTI system

$$\begin{aligned}
\dot{x}(t) &= Ax(t) + Bu(t) \\
u(t) &= Kx(t)
\end{aligned} \tag{2.19}$$

where  $A$ ,  $B$ , and  $K$  are of appropriate dimensions. This system is said to be *quadratically stabilizable* (via linear state-feedback) if there exists a control gain  $K$  such that the closed loop system (2.19) is quadratically stable. Therefore, the above LTI system is stable (quadratically stable) iff there exists  $P > 0$  such that

$$(A + BK)^T P + P(A + BK) < 0 \tag{2.20}$$

This equation can be easily used for analysis purposes. However, in order to design the control gain  $K$ , its equivalent expression can be used as

$$Q(A + BK)^T + (A + BK)Q < 0 \tag{2.21}$$

Neither of the above conditions is jointly convex in terms of  $K$  and  $P$  or  $Q$ . However, by defining a new variable as  $Y = KQ$ , the LMI condition of (2.21) can be rewritten as

$$AQ + QA^T + BY + Y^T B^T < 0 \tag{2.22}$$

which is convex in terms of both  $Q$  and  $K$ . Therefore, the system of (2.19) is stable if there exist matrices  $Q \geq 0$  and  $K$  such that LMI (2.22) holds. Finally, gain  $K$  is obtained as  $K = YQ^{-1}$ .

**Remark 2.1.** *In the following sections, depending on the controller design method, one of the above mentioned LMI conditions may be applied.*

## 2.4 Fault Modeling

The focus of this thesis will be on the fault-tolerant control of the actuator fault in AUV systems. This may seem to make the topic relatively narrow. However, in many applications (such as aerospace) it is well known that most sensor faults are handled via hardware redundancy and voting strategies [103], and plant faults are relatively rare. Moreover, as extensively in literature, a sensor fault could be modeled as an actuator fault. Thus, the actuator faults deserve great attention for a safe and reliable operation of the system.

Let us now investigate how an actuator fault is demonstrated. As originally stated in [104], an actuator fault could be modeled as

$$u_k^f = \gamma_k u_k + \gamma_{k0} \beta_k^f \quad (2.23)$$

where  $u_k^f$  and  $u_k$  are the control input signals after and before fault occurrence and  $m$  is the number of control inputs.  $\beta_k^f$  is also a bounded disturbance applied to the system through the control input, and  $\gamma_k$  and  $\gamma_{k0}$  satisfy

$$\begin{cases} 0 \leq \underline{\gamma}_k \leq \gamma_k \leq \bar{\gamma}_k, & k = 1, \dots, m \\ 0 \leq \gamma_{k0} \leq \bar{\gamma}_{k0}, & k = 1, \dots, m \end{cases}$$

$$\begin{aligned} \gamma &= \text{diag}[\gamma_1, \gamma_2, \dots, \gamma_m] \\ \gamma_o &= \text{diag}[\gamma_{1o}, \gamma_{2o}, \dots, \gamma_{mo}] \\ \bar{\gamma}_o &= \text{diag}[\bar{\gamma}_{1o}, \bar{\gamma}_{2o}, \dots, \bar{\gamma}_{mo}] \\ \bar{\gamma} &= \text{diag}[\bar{\gamma}_1, \bar{\gamma}_2, \dots, \bar{\gamma}_m] \\ \underline{\gamma} &= \text{diag}[\underline{\gamma}_1, \underline{\gamma}_2, \dots, \underline{\gamma}_m] \\ \beta^f &= [\beta_1^f, \beta_2^f, \dots, \beta_m^f]^T \end{aligned} \quad (2.24)$$

Three types of actuator faults that are investigated in this thesis are: loss of effectiveness (LOE), lock in place (LIP), and float fault [105]. Depending on the type of a fault, the parameters  $\gamma_k$  and  $\gamma_{k0}$  that are introduced in (2.23) are:

$$\begin{cases} \gamma_k \neq 0, \gamma_{k0} = 0 & \text{LOE} \\ \gamma_k = 0, \gamma_{k0} = 0 & \text{float fault} \\ \gamma_k = 0, \gamma_{k0} \neq 0 & \text{LIP} \end{cases} \quad (2.25)$$

Moreover, there is another type of fault called hard-over-failure, which is modeled as actuator saturation.

## 2.5 Summary

This chapter is a preliminary introduction to the background information and definitions. First, a detailed literature review on AUV modeling including the equations of motion of a rigid body, hydrodynamic damping, gravitational and buoyant forces, environmental disturbances, and AUV's sensors and actuators as well as its propulsion system is provided. Following that, a review of the mathematics and the definitions that are used throughout the thesis is presented. This review includes the algebraic graph theory, the matrix theory, and the fault modeling.

# Chapter 3

## State Feedback Synchronization

In the previous chapter, besides the introduction of the AUV's equations of motion, three relevant points were underlined. Firstly, environmental disturbances, particularly ocean currents, were discussed. As previously mentioned, the effects of ocean currents could be modeled as a white Gaussian noise. Secondly, the uncertainty of the AUV's model parameters as a significant practical issue was investigated. Finally, the actuator faults, which are among the main real world problems, were studied. This chapter will focus on the state feedback synchronization of AUVs by addressing all three parts above.

### 3.1 Network Structure and Model Dynamics Equation

The network structure that is considered throughout the thesis is a constant topology (fixed) network. Moreover, the consensus strategy is an average consensus approach, which is determined based on the network's Laplacian matrix and the initial states of the agents. For simulations purposes, a network of five agents is considered as in Figure 3.1.

The Laplacian matrix for this graph is:

$$L = \begin{bmatrix} 2 & -1 & 0 & 0 & -1 \\ -1 & 3 & -1 & 0 & -1 \\ -1 & -1 & 4 & -1 & -1 \\ 0 & 0 & 0 & 1 & -1 \\ -1 & -1 & 0 & 0 & 2 \end{bmatrix} \quad (3.1)$$

Throughout the thesis, a longitudinal plane model of the autonomous underwater vehicle (AUV) is considered. The longitudinal model is composed of the heave, the pitch and their associated rates and the vehicle's forward speed. However, given that the vehicle's forward speed is kept constant

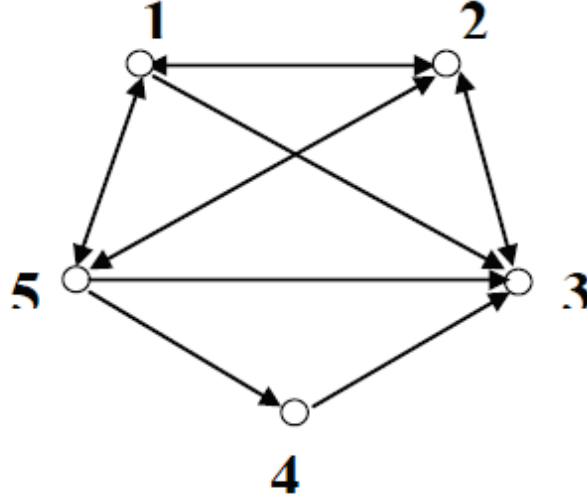


Figure 3.1: Network topology.

through a separate control loop, the model will be reduced to four states (heave, heave rate, pitch and pitch rate).

The longitudinal plane model (2.12) is

$$\begin{aligned}
 & \begin{bmatrix} m - Z_{\dot{w}} & mx_g - Z_{\dot{q}} & 0 & 0 \\ mx_g - Z_{\dot{q}} & I_y - M_{\dot{q}} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \dot{w} \\ \dot{q} \\ \dot{z} \\ \dot{\theta} \end{bmatrix} + \\
 & \begin{bmatrix} -Z_w & mu_0 - Z_q & 0 & 0 \\ -M_w & mx_G u_0 - M_q & 0 & (z_G - z_B)W \\ -1 & 0 & 0 & u_0 \\ 0 & -1 & 0 & 0 \end{bmatrix} \begin{bmatrix} w \\ q \\ z \\ \theta \end{bmatrix} = \begin{bmatrix} Z \\ M \\ 0 \\ 0 \end{bmatrix} \quad (3.2)
 \end{aligned}$$

This can be written in a conventional state-space form as

$$\dot{\mathbf{x}}_i = \begin{bmatrix} a_{11} & a_{12} & 0.0 & a_{14} \\ a_{21} & a_{22} & 0.0 & a_{24} \\ 1.0 & 0 & 0 & a_{34} \\ 0 & 1 & 0 & 0 \end{bmatrix} \mathbf{x}_i + B \mathbf{u}_i \quad (3.3)$$

where  $\mathbf{x}_i = [w \ q \ z \ \theta]^T$ ,  $\mathbf{u}_i \in \mathbb{R}^{m \times 4}$  and  $A \in \mathbb{R}^{4 \times 4}$ , and  $B \in \mathbb{R}^{4 \times m}$  for  $i = 1, \dots, N$  are the state vector, control input, state-space dynamic matrix, and control input matrix, respectively.  $m$

represents the number of control inputs which varies between 1 and 2 in different scenarios. The AUV model (3.2) has two degrees of freedom.

As previously mentioned, the INFANTE AUV [106] has a constant forward speed, which is achieved by using the propeller controller. Hence, in this thesis, the forward speed will not be considered as a control variable. This vehicle is also equipped with the stern and sail (bow) hydroplanes, which control the vehicle's pitch angle and consequently, its depth. For hydrodynamic coefficients of this INFANTE AUV, one can refer to [106].

**Remark 3.1.** *There is no concern about the collision avoidance as the consensus is defined only on  $z$  direction, and not on the agents' position.*

## 3.2 State Feedback Consensus under Healthy Scenario

Consider a group of  $N$  homogeneous multi-agent systems as

$$\begin{aligned}\dot{\mathbf{x}}_i(t) &= A\mathbf{x}_i(t) + B\mathbf{u}_i(t), \quad i = 1, 2, \dots, N \\ z_{oi}(t) &= \mathbf{x}_i(t)\end{aligned}\tag{3.4}$$

where  $\mathbf{x}_i \in \mathbb{R}^n$  and  $z_{oi} \in \mathbb{R}^n$  are the  $i$ th vehicle state and measurement, respectively.  $A \in \mathbb{R}^{n \times n}$  is a marginally stable matrix,  $B \in \mathbb{R}^{n \times m}$  and the pair  $(A, B)$  is stabilizable.

**Remark 3.2.** *This section and the entire chapter cover both single-input and multi-input systems. However, depending on the simulation scenario, either the single-input or the multi-input model may be used.*

For system (3.4), synchronization is achieved if for all  $i = 2, \dots, N$

$$\|z_{o1}(t) - z_{oi}(t)\| = \|\mathbf{x}_1(t) - \mathbf{x}_i(t)\| \rightarrow 0, \quad t \rightarrow \infty\tag{3.5}$$

Now, let us define  $\sigma_i(t) = \mathbf{x}_1(t) - \mathbf{x}_i(t)$ ,  $i = 2, 3, \dots, N$ . The synchronization achievement is guaranteed if  $\sigma_i(t) \rightarrow 0$ ,  $t \rightarrow \infty$ ,  $i = 2, 3, \dots, N$ .

To reach an agreement under the network topology  $\mathcal{G}$ , a common methodology as in the literature [31], [32], and [33] is to apply a static protocol:

$$\mathbf{u}_i(t) = -K \sum_{j \in \mathcal{N}_i} a_{ij}(\mathbf{x}_i(t) - \mathbf{x}_j(t))\tag{3.6}$$

where  $a_{ij}$ 's are the entries of the adjacency matrix, and  $K \in \mathbb{R}^{m \times n}$  is the control gain matrix.

Using the results from [34], by applying the protocol of (3.6), the augmented state-space representation of  $\sigma_i(t)$  could be written as

$$\begin{aligned}\dot{\sigma}(t) &= (I_{N-1} \otimes A - (L_r + I_{N-1} \cdot \alpha^T) \otimes BK) \sigma(t) \\ z(t) &= \sigma(t)\end{aligned}\tag{3.7}$$

where  $\sigma(t) = [\sigma_2^T(t), \sigma_3^T(t), \dots, \sigma_N^T(t)]^T$  is the augmented state-space,  $z(t)$  is the measurement in transformed coordinates, and  $\alpha$  and  $L_r$  are

$$\begin{aligned}\alpha &= (a_{12}, a_{13}, \dots, a_{1N})^T \\ L_r &= \begin{pmatrix} \deg_{in}(2) & -a_{23} & \cdots & -a_{2N} \\ -a_{32} & \deg_{in}(3) & \cdots & -a_{3N} \\ \cdots & & \cdots & \\ -a_{N2} & -a_{N3} & \cdots & \deg_{in}(N) \end{pmatrix}\end{aligned}\tag{3.8}$$

where  $\deg_{in}(i)$  represents the in-degree of node  $i$ .

Similar to [34], by using the similarity transformation of  $S = \begin{pmatrix} 1 & 0 \\ I_{N-1} & I_{N-1} \end{pmatrix}$  one can show

$$S^{-1}LS = \begin{pmatrix} 0 & -\alpha^T \\ 0 & L_r + I_{N-1} \cdot \alpha^T \end{pmatrix}\tag{3.9}$$

where  $L = [l_{ij}]$  is the Laplacian matrix of the original system with  $l_{ii} = \sum_{j \in \mathcal{N}_i} a_{ij}$ , and  $l_{ij} = -a_{ij}$  for  $i \neq j$ .

As in literature, for a connected graph  $\mathcal{G}$ , except for  $\lambda_1$ , which is zero, all other eigenvalues of  $L$  ( $\lambda_i = \sigma_i + jw_i$   $i = 2, \dots, N$ ) have positive real parts. Then, by using the properties of a similarity transformation and from (3.9), the eigenvalues of  $L_r + I_{N-1} \cdot \alpha^T$  are  $\lambda_2, \lambda_3, \dots, \lambda_N$ .

**Lemma 3.1.** *For a group of agents with dynamics (3.4) and a graph of directed spanning tree, protocol (3.6) solves the consensus problem if and only if*

$$A - \lambda_i BK < 0 \quad i = 2, \dots, N\tag{3.10}$$

*Proof.* See the proof of Lemma 2 in [34]. □

In order to obtain the control gain  $K$  in a way that guarantees the Hurwitzness of  $A - \lambda_i BK$ , the problem can be described as an LMI problem:

$$\begin{aligned} (A - \lambda_i BK)^T P + P(A - \lambda_i BK) &< 0 \quad i = 2, \dots, N \\ P &> 0 \end{aligned} \quad (3.11)$$

or equivalently,

$$\begin{aligned} (A - \lambda_i BK)Q + Q(A - \lambda_i BK)^T &< 0 \quad i = 2, \dots, N \\ Q &> 0 \end{aligned} \quad (3.12)$$

### 3.2.1 Simulation Results

In this section, a single input model (with stern as input) is employed. The linearized model about the equilibrium point,  $\mathbf{x}_0 = [w_o, q_o, z_o, \theta_o]' = [0, 0, 0, 0]'$  and  $\mathbf{u}_o = \sigma_s = 0$ , and at a forward velocity of 2.0 m/s is:

$$\dot{\mathbf{x}}_i = \begin{bmatrix} -1.400 & 2.763 & 0.0 & 0.078 \\ 2.108 & -5.419 & 0.0 & -0.312 \\ 1.0 & 0 & 0 & -2.0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \mathbf{x}_i + \begin{bmatrix} -0.201 \\ -0.809 \\ 0 \\ 0 \end{bmatrix} \sigma_s \quad (3.13)$$

The state-space model of (3.13) is controllable. Moreover, this linearized model is an under-actuated system because it has two controlled variables (2-DOF) and only one control effector.

In model (3.13), the state-space dynamic matrix  $A$  has one stable second-order mode with natural frequency of 0.121 rad/s, a real stable mode at  $-6.5$  rad/s, and a zero mode. Regardless of the forward velocity value, the dynamic matrix always has a mode at zero.

**Remark 3.3.** *In all the simulations in this section and throughout the thesis, the origin of the  $z$  axis is defined undersea (as a representative of the deployment depth). Therefore, in simulation results, positive values of depth could be observed.*

**Remark 3.4.** *In all the simulations in this section and throughout the thesis, the control inputs provided by the stern and bow actuators are limited to the range  $[-60 \ 60]$  deg, which is the maximum angle of control surface.*

**Remark 3.5.** *The model (3.13) is employed for the single-input model simulation in this thesis.*

By employing the approach in Section 3.2, the LMI control gain  $K$  of equation (3.6) is designed as

$$K = \begin{bmatrix} -0.3812 & -0.1793 & 0.3063 & -1.5953 \end{bmatrix} \quad (3.14)$$

The simulation results for this case are shown in Figures 3.2 - 3.6. As in the Figure 3.2, the control inputs are limited between  $[-60 \ 60]$ . Figures 3.3 - 3.6 show that consensus is achieved on all states.

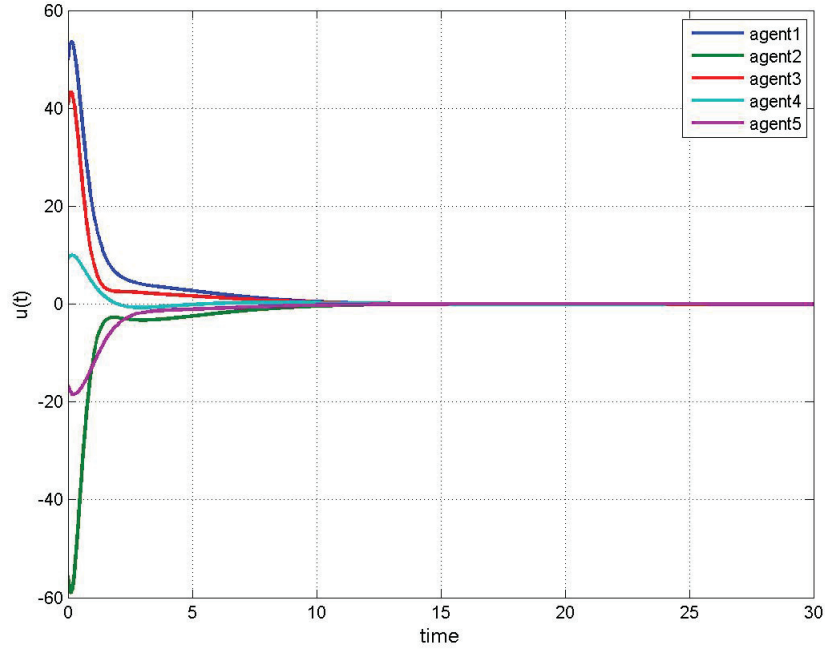


Figure 3.2: Control input signals  $u(t)$  for static state feedback protocol under healthy scenario.

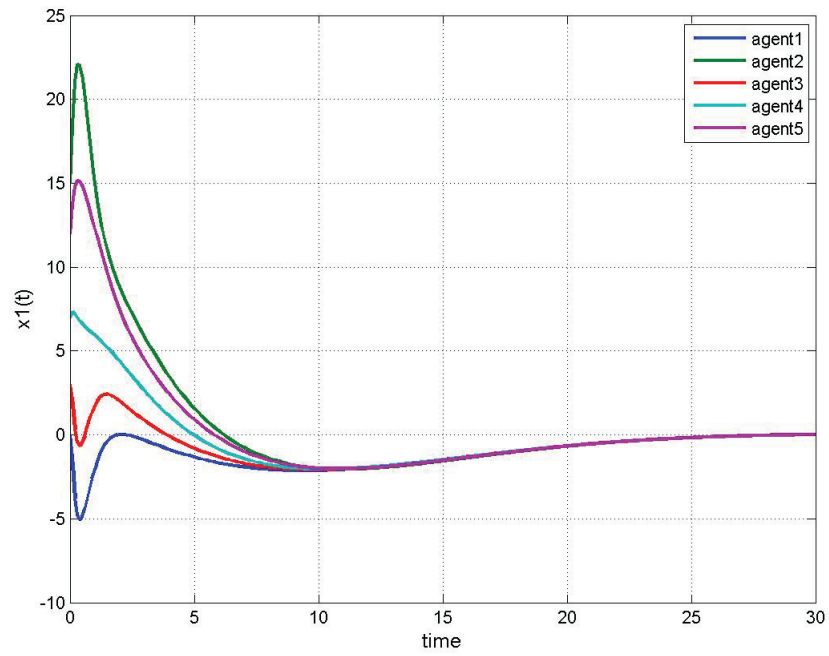


Figure 3.3: Synchronization of state 1 for static state feedback protocol under healthy scenario.

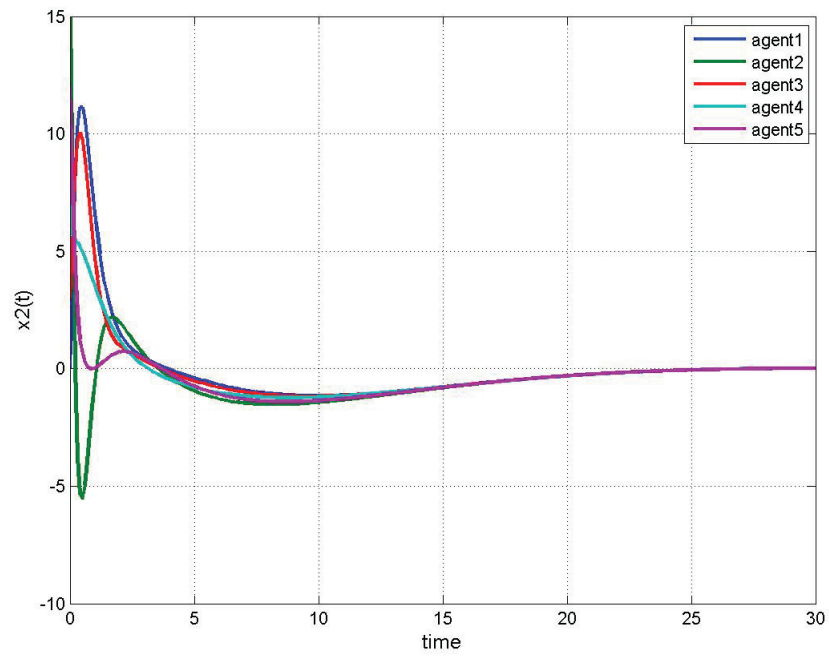


Figure 3.4: Synchronization of state 2 for static state feedback protocol under healthy scenario.

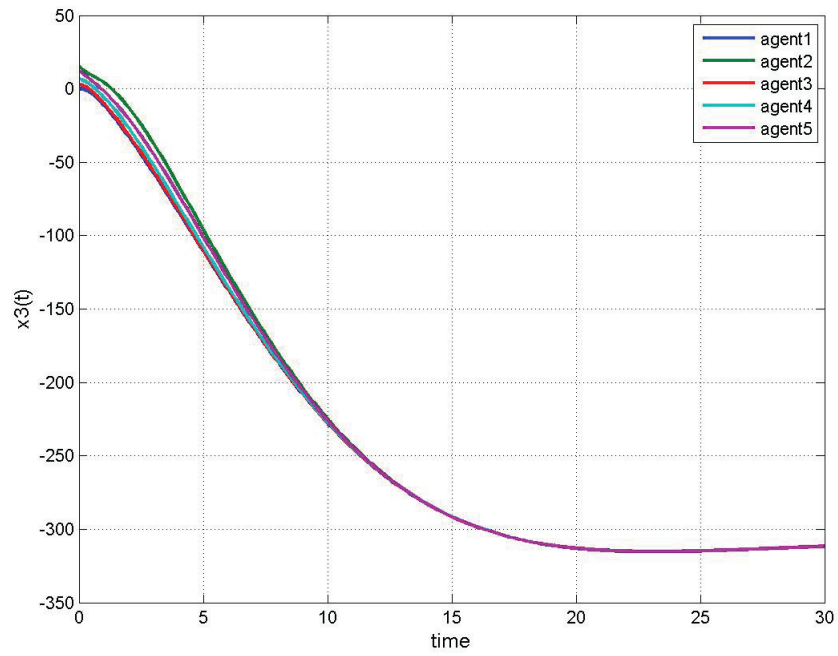


Figure 3.5: Synchronization of state 3 for static state feedback protocol under healthy scenario.

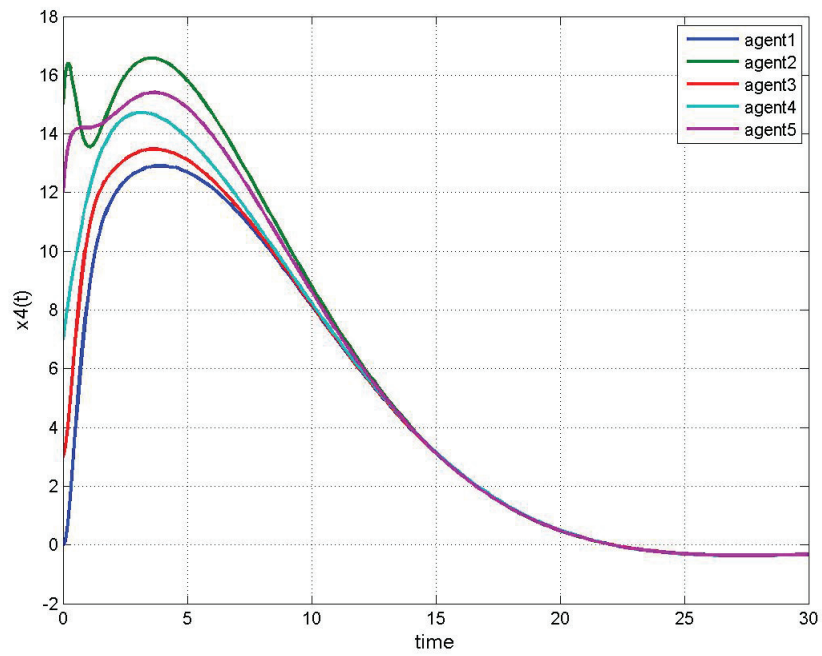


Figure 3.6: Synchronization of state 4 for static state feedback protocol under healthy scenario.

### 3.3 State Feedback Consensus Subject to the Process and Measurement Noises

In this section, as in a real-world scenario, the effect of noise on the agents' dynamics and measurements is considered. Therefore, as in section 2.1.7, the linear stochastic differential equation of the system will be

$$\begin{aligned} d\mathbf{x}_i(t) &= A\mathbf{x}_i(t)dt + B\mathbf{u}_i(t)dt + Bd\mathbf{w}_i(t), \quad i = 1, 2, \dots, N \\ dz_i(t) &= d\mathbf{x}_i(t) + dv_i(t) \end{aligned} \quad (3.15)$$

where  $d\mathbf{w}_i \in \mathbb{R}^n, dv_i \in \mathbb{R}^n$  are Wiener processes with known statistics.

The statistics of the states' initial conditions are

$$\begin{aligned} E[\mathbf{x}_i(0)] &= \hat{\mathbf{x}}_{i0} \quad i = 1, 2, \dots, N \\ E\{[\mathbf{x}_i(0) - \hat{\mathbf{x}}_{i0}][\mathbf{x}_i(0) - \hat{\mathbf{x}}_{i0}]^T\} &= P_{i0} \end{aligned} \quad (3.16)$$

The term  $\mathbf{w}_i(t)$ , which accounts for the environmental disturbances, is modeled as a zero mean Gaussian white noise process (it is the derivative of the Wiener process  $d\mathbf{w}_i(t)$ ):

$$\begin{aligned} E[\mathbf{w}_i(t)] &= 0 \quad i = 1, 2, \dots, N \\ E\{\mathbf{w}_i(t)\mathbf{w}_i(\tau)^T\} &= Q_i(t)\delta(t - \tau) \end{aligned} \quad (3.17)$$

which is specified by its spectral density matrix  $Q_i(t)$ . The measurement noise, which is modeled as a zero mean Gaussian white noise process (it is the derivative of the Wiener process  $dv_i(t)$ ), is

$$\begin{aligned} E[v_i(t)] &= 0 \quad i = 1, 2, \dots, N \\ E\{v_i(t)v_i(\tau)^T\} &= R_i(t)\delta(t - \tau) \end{aligned} \quad (3.18)$$

where the measurement uncertainty is expressed by its spectral density matrix  $R_i(t)$ . It is also assumed that the states and measurement noises are uncorrelated.

**Remark 3.6.** *The Wiener process is used to represent the integral of a white noise Gaussian process.*

**Lemma 3.2.** *Under the control protocol of (3.6), system (3.15) will synchronize in the stochastic mean-square sense.*

*Proof.* In order to prove Lemma 3.2, the problem is casted into a stochastic framework. Particularly, it will be shown that the expected values of the states will converge. In this regard, the protocol of (3.6) is applied to (3.15) and the new state-space variable  $\sigma_i(t) = \mathbf{x}_1(t) - \mathbf{x}_i(t)$  for  $i = 2, \dots, N$  is defined. For the sake of simplicity, only the process noise is considered. However, in a similar manner it could be extended to the system with measurement noise. Then, the new differential equations become

$$\begin{aligned}
d\sigma_i(t) &= d\mathbf{x}_1(t) - d\mathbf{x}_i(t) = \\
&A\mathbf{x}_1(t)dt + B\mathbf{u}_1(t)dt + Bdw_1(t) - (A\mathbf{x}_i(t)dt + B\mathbf{u}_i(t)dt + Bdw_i(t)) = \\
&A\sigma_i(t)dt - BK\left[\sum_{j \in \mathcal{N}_1} a_{1j}(z_1(t) - z_j(t)) - \sum_{j \in \mathcal{N}_i} a_{ij}(z_i(t) - z_j(t))\right]dt + dw_{1i}(t) = \\
&A\sigma_i(t)dt - BK\left[\sum_{j \in \mathcal{N}_1} a_{1j}(\mathbf{x}_1(t) - \mathbf{x}_j(t)) - \sum_{j \in \mathcal{N}_i} a_{ij}(\mathbf{x}_i(t) - \mathbf{x}_j(t))\right]dt + dw_{1i}(t) = \\
&A\sigma_i(t)dt - BK\left[\sum_{j \in \mathcal{N}_1} a_{1j}\sigma_j(t) - \sum_{j \in \mathcal{N}_i} a_{ij}(\sigma_j(t) - \sigma_i(t))\right]dt + dw_{1i}(t) \\
z_i(t) &= \sigma_i(t) \quad i = 2, 3, \dots, N
\end{aligned} \tag{3.19}$$

where  $dw_{1i}(t) = B(dw_1(t) - dw_i(t))$  is a Wiener process.

Similar to (3.7), the augmented state-space equation of (3.19) could be written as

$$\begin{aligned}
d\sigma(t) &= (I_{N-1} \otimes A - (L_r + I_{N-1} \cdot \alpha^T) \otimes BK)\sigma(t)dt + dW_1(t) \\
z(t) &= \sigma(t)
\end{aligned} \tag{3.20}$$

where  $dW_1(t)$  is defined as  $dW_1(t) = [dw_{12}(t), \dots, dw_{1N}(t)]^T$ .

Using the results from [107], if  $dW_1(t)$  has a bounded variation, the solution of (3.20) can be written as

$$\sigma(t) = \Phi(t, t_0)\sigma(t_0) + \int_{t_0}^t \Phi(t, s)dW_1(s) \tag{3.21}$$

where  $\Phi(t, t_0)$  satisfies the differential equation

$$\frac{d\Phi(t, t_0)}{dt} = (I_{N-1} \otimes A - (L_r + I_{N-1} \cdot \alpha^T) \otimes BK)\Phi(t, t_0) \tag{3.22}$$

Since  $\sigma$  is a linear function of a normal process, it is also normal, and can be completely charac-

terized by its mean and covariance. Then, its expected value will be

$$E\{\sigma(t)\} = \Phi(t, t_0)E\{\sigma(t_0)\} + E\left\{\int_{t_0}^t \Phi(t; s)dW_1(s)\right\} \quad (3.23)$$

Using the property that the operations of mathematical expectation and integration in the sense of Ito can be interchanged [107], we will have

$$E\{\sigma(t)\} = \Phi(t, t_0)E\{\sigma(t_0)\} + \int_{t_0}^t \Phi(t; s)E\{dW_1(s)\} \quad (3.24)$$

It can be easily seen that the second term on the right hand side of the above equation disappears. This results in

$$E\{\sigma(t)\} = \Phi(t, t_0)E\{\sigma(t_0)\} \quad (3.25)$$

From linear system theory and using the state transition matrix dynamics (3.22), the solution of the expected value of  $\sigma(t)$  is governed by (3.25). It can be observed that  $\Phi(t, t_0)$  has the same dynamics as (3.7). Hence, if the control gain  $K$  is chosen such that the equation (3.10) holds, the matrix  $\Phi(t, t_0)$  will also stabilize, and therefore the solution of  $E\{\sigma(t)\}$  will go to zero. This means consensus is achieved in the stochastic mean-square sense.  $\square$

In order to have a measure of how severe the noise affects the system's overall stability and performance, the following performance indices are defined:

- Control input performance index

$$PI_{\mathbf{u}} = E\left\{\frac{1}{T - T_1} \int_{T_1}^T \mathbf{u}_i^2(t)dt\right\} \quad (3.26)$$

- State performance index

$$PI_{\mathbf{x}} = E\left\{\frac{1}{T - T_1} \int_{T_1}^T \mathbf{x}_i^2(t)dt\right\} \quad (3.27)$$

- Total performance index

$$PI_{\mathbf{xu}} = E\left\{\frac{1}{T - T_1} \int_{T_1}^T (\mathbf{u}_i^2(t) + \mathbf{x}_i^2(t)) dt\right\} \quad (3.28)$$

where  $T_1$  stands for the time that the signal reaches its steady state and  $T \rightarrow \infty$ . For faulty scenarios,  $T_1$  is considered as the start of simulation time.

The above notion of performance indices are used for a single agent. The network performance indices are defined as the average of the agents' performance indices.

### 3.3.1 Simulation Results

In this section, the Monte-Carlo simulation results of equation (3.15) by employing a single-input model (3.13) are presented. For the simulations purposes, the spectral density matrix of the process and measurement noises are considered as  $Q = I_{(m \times m)}$  and  $R = 0.5 \times I_{(n \times n)}$ , respectively. The control gain  $K$  is selected as

$$K = \begin{bmatrix} 0.2790 & -0.6279 & 1.0000 & -4.0064 \end{bmatrix} \quad (3.29)$$

This control gain  $K$  is selected different from the one in Equation (3.14), as the Equation (3.10) in Lemma 3.1 does not have a unique solution of  $K$ .

The results for one of the 50 runs under Monte-Carlo simulations are shown in Figures 3.7 - 3.14.

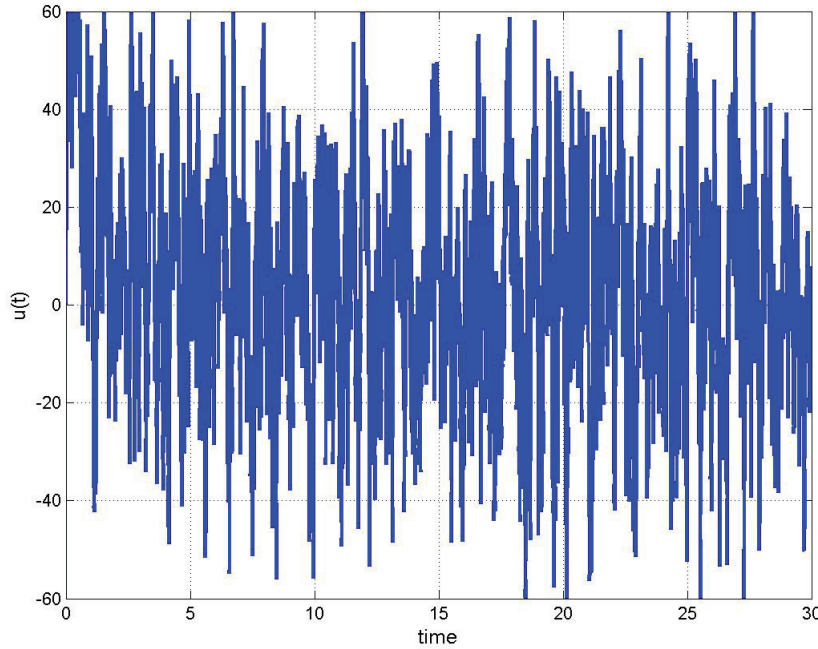


Figure 3.7: Control input signal of agent 1 for static state feedback protocol subject to noise - gain set 1.

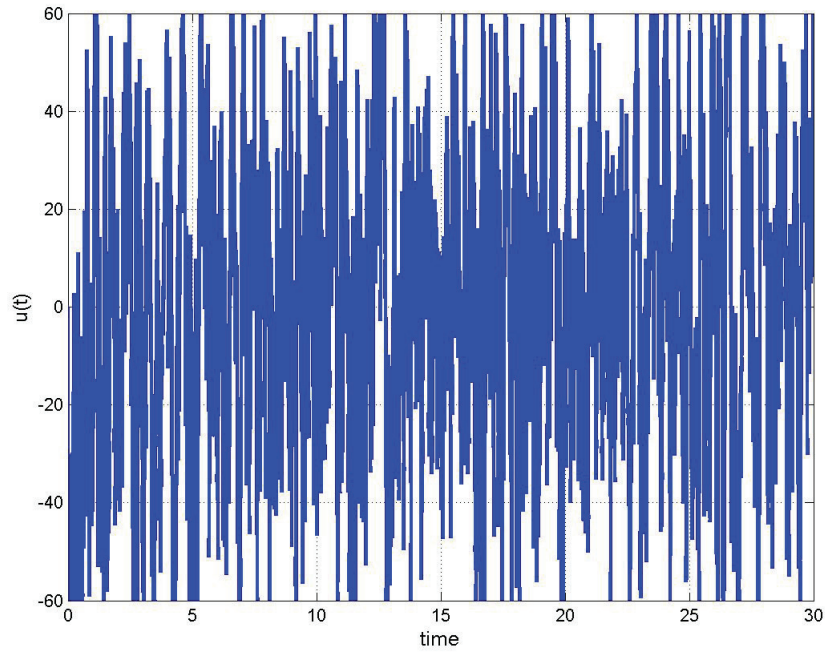


Figure 3.8: Control input signal of agent 2 for static state feedback protocol subject to noise - gain set 1.

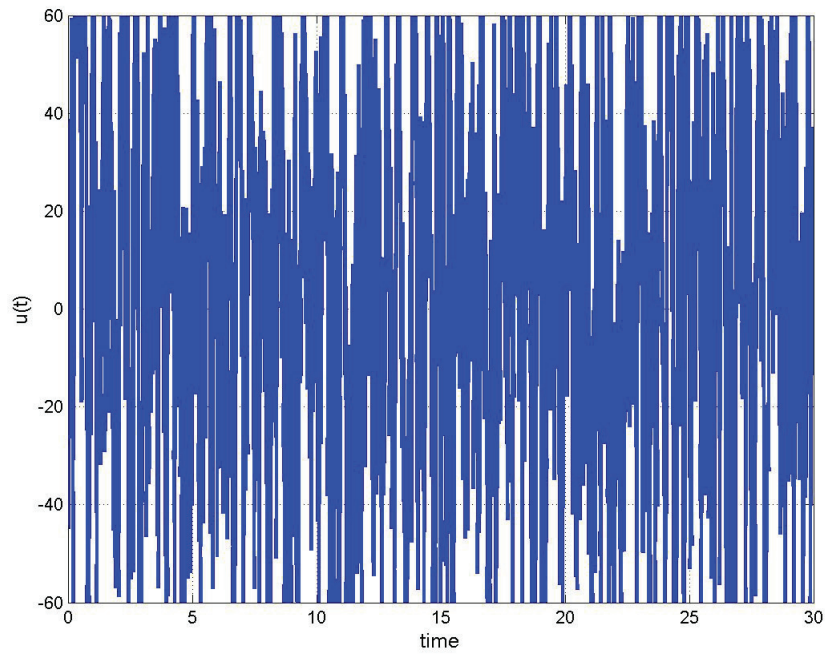


Figure 3.9: Control input signal of agent 3 for static state feedback protocol subject to noise - gain set 1.

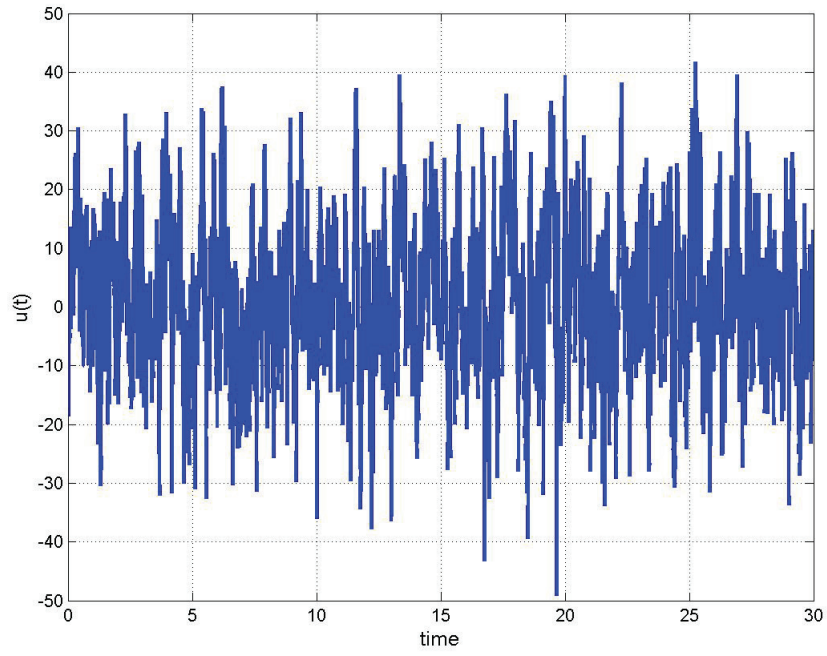


Figure 3.10: Control input signal of agent 4 for static state feedback protocol subject to noise - gain set 1.

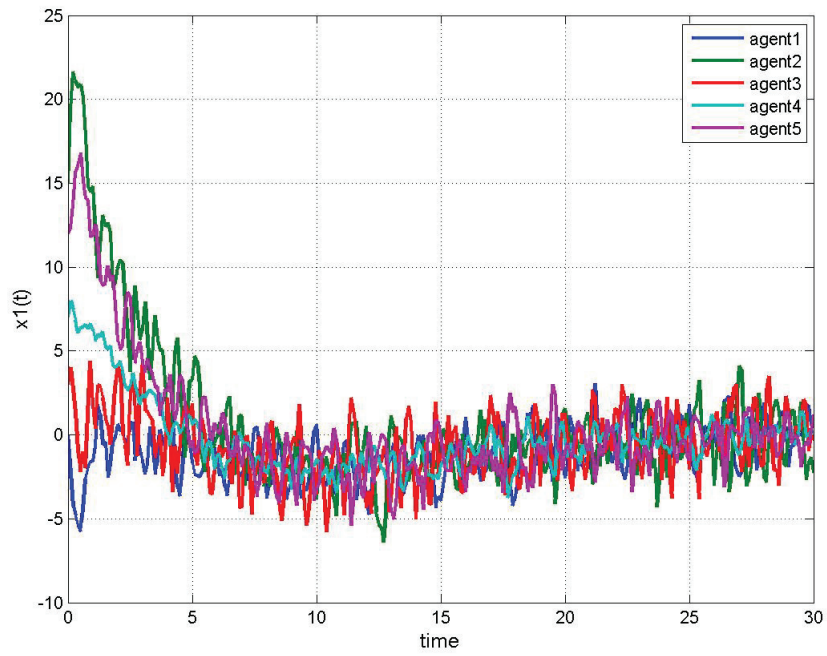


Figure 3.11: Synchronization of state 1 for static state feedback protocol subject to noise - gain set 1.

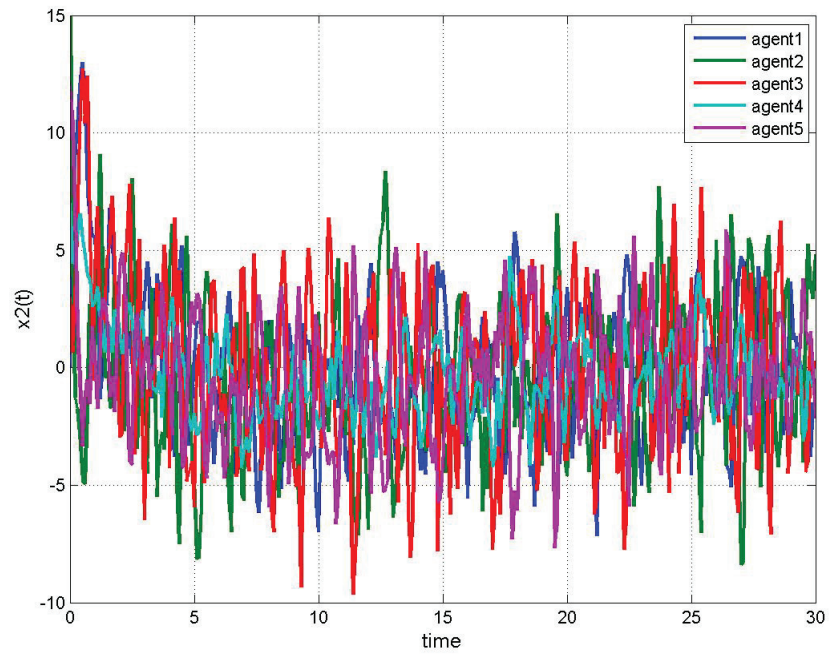


Figure 3.12: Synchronization of state 2 for static state feedback protocol subject to noise - gain set 1.

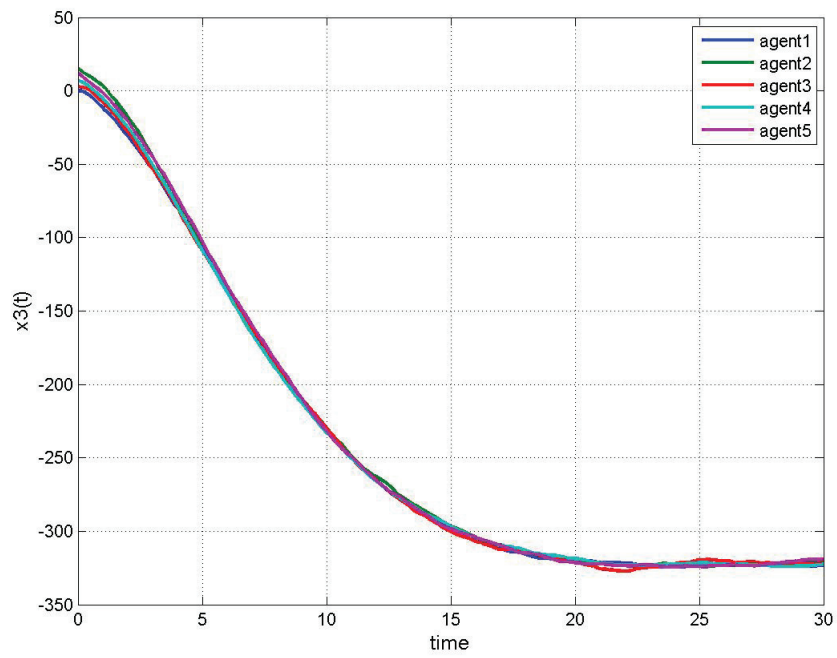


Figure 3.13: Synchronization of state 3 for static state feedback protocol subject to noise - gain set 1.

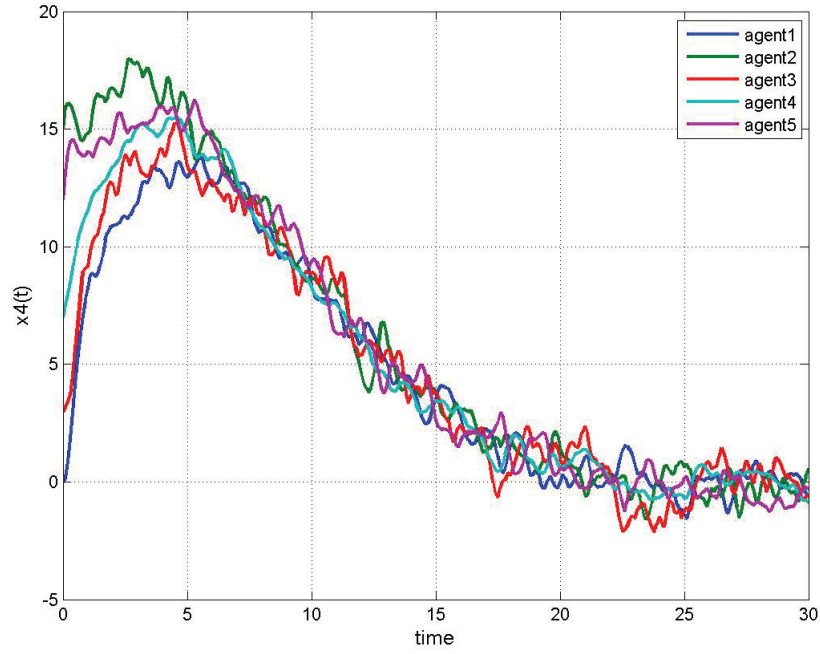


Figure 3.14: Synchronization of state 4 for static state feedback protocol subject to noise - gain set 1.

As it can be seen in Figures 3.7 - 3.10, the control inputs have high variations, which is not practical in a real application. It can be seen from Figures 3.11 - 3.14 that all states synchronize in a stochastic sense.

Table 3.1 summarizes the performance indices of 50 Monte-Carlo simulations as defined in equations (3.26) - (3.28).

Table 3.1: Monte-Carlo simulation results for performance indices of the single agents and network in the presence of noise and by employing the first set of gains. The Network refers to the averaging of the entire team performance.

	$PI_x$	$PI_u$	$PI_{xu}$
Agent 1	25	505	530
Agent 2	30	900	930
Agent 3	23	1330	1353
Agent 4	14	170	184
Agent 5	19	520	539
Network	21	705	726

In order to investigate the effect of controller gain on the performance indices, another set of gain is applied. The results for one of the runs under Monte-Carlo simulations are summarized in Figures 3.15 - 3.22. Compared to the first gain, the second gain is selected such that the contribution

of the control input performance index in the total performance index is decreased.

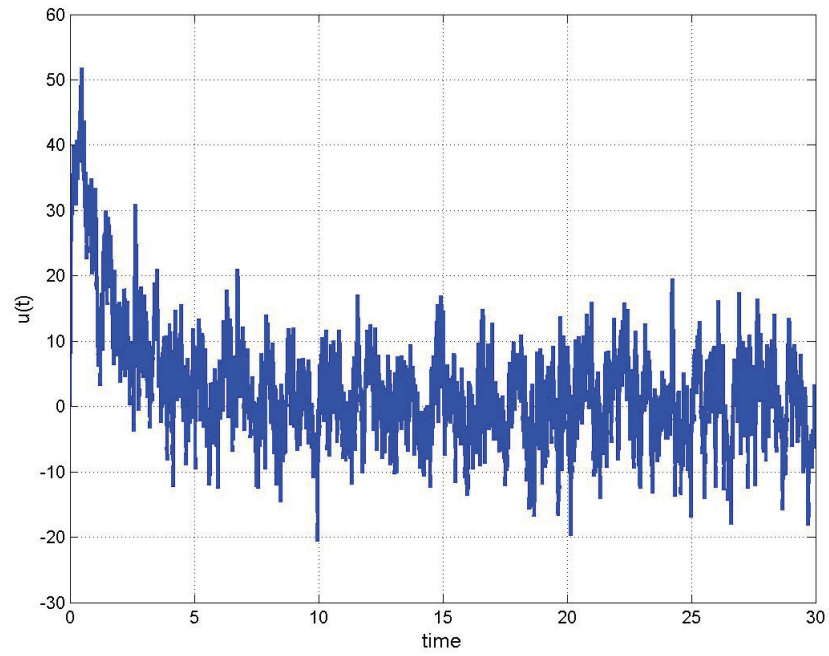


Figure 3.15: Control input signal of agent 1 for static state feedback protocol subject to noise - gain set 2.

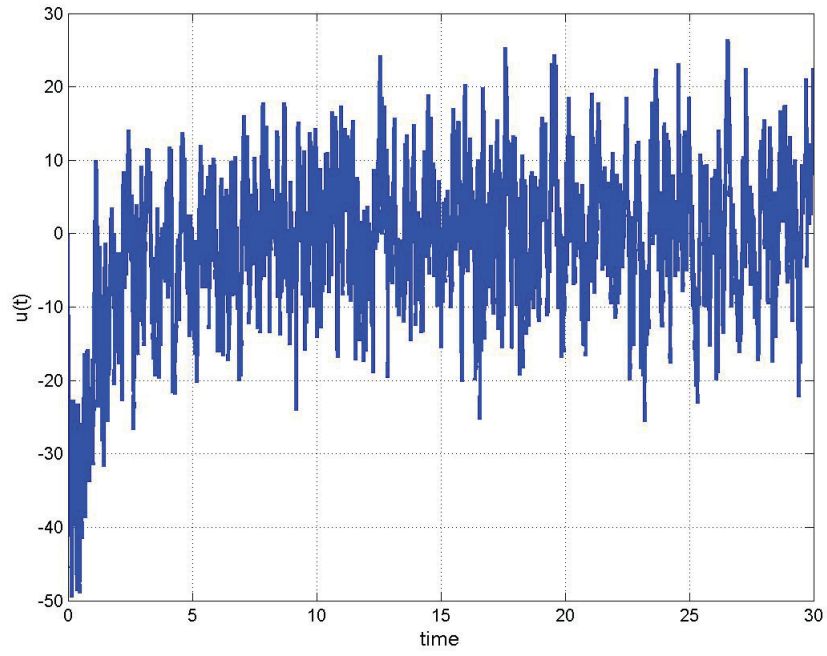


Figure 3.16: Control input signal of agent 2 for static state feedback protocol subject to noise - gain set 2.

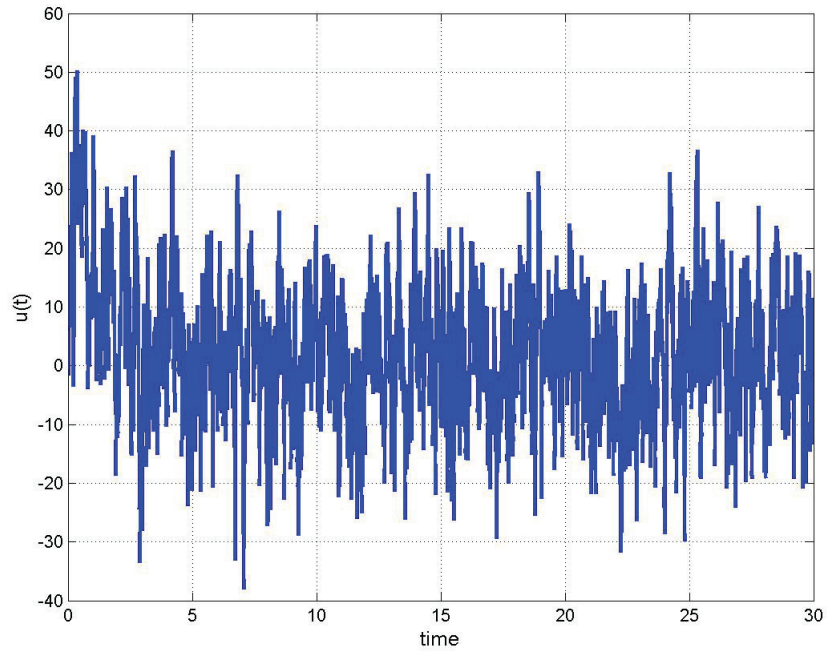


Figure 3.17: Control input signal of agent 3 for static state feedback protocol subject to noise - gain set 2.

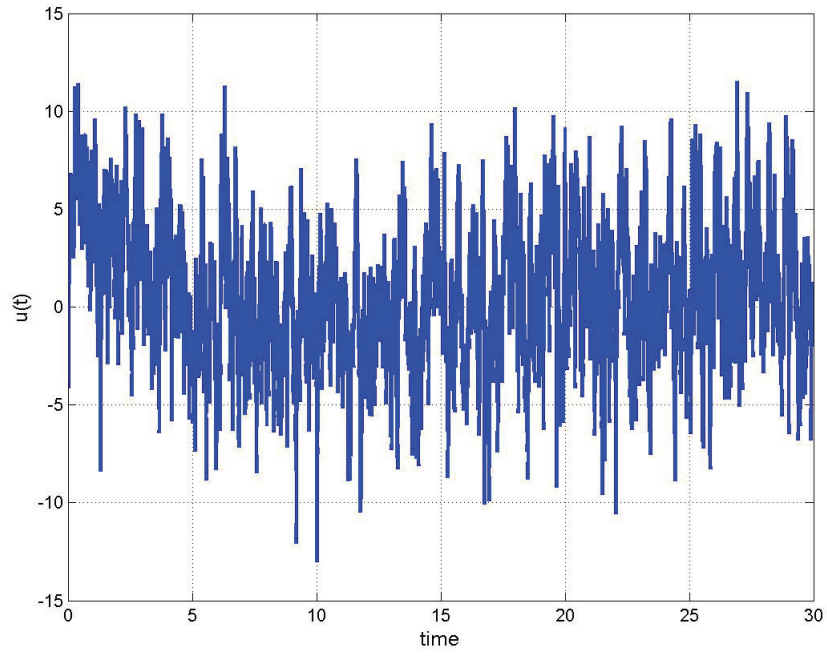


Figure 3.18: Control input signal of agent 4 for static state feedback protocol subject to noise - gain set 2.

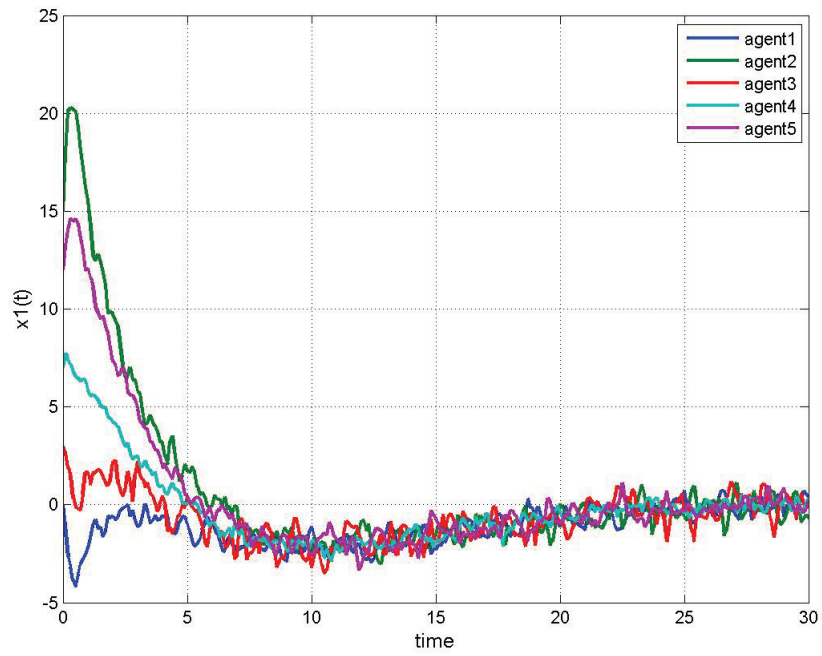


Figure 3.19: Synchronization of state 1 for static state feedback protocol subject to noise - gain set 2.

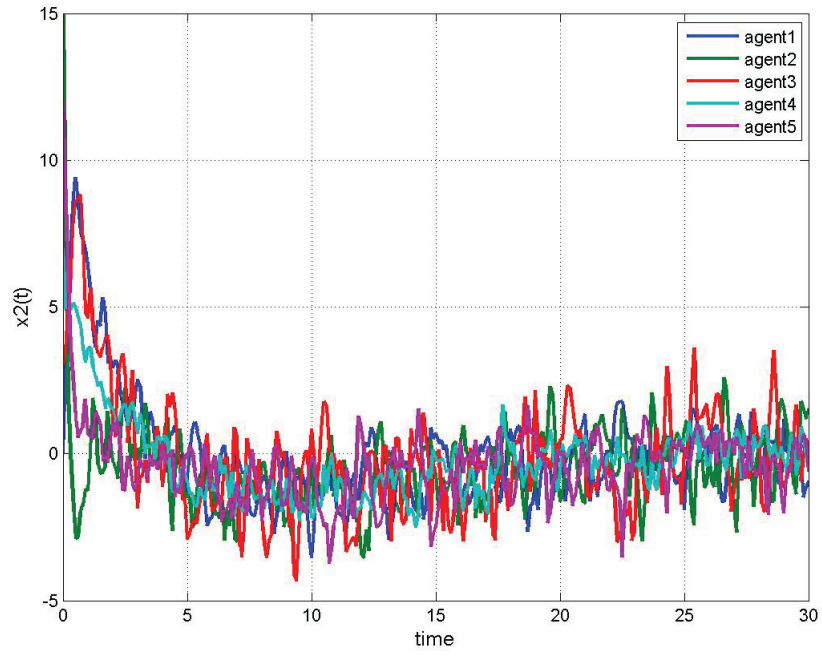


Figure 3.20: Synchronization of state 2 for static state feedback protocol subject to noise - gain set 2.

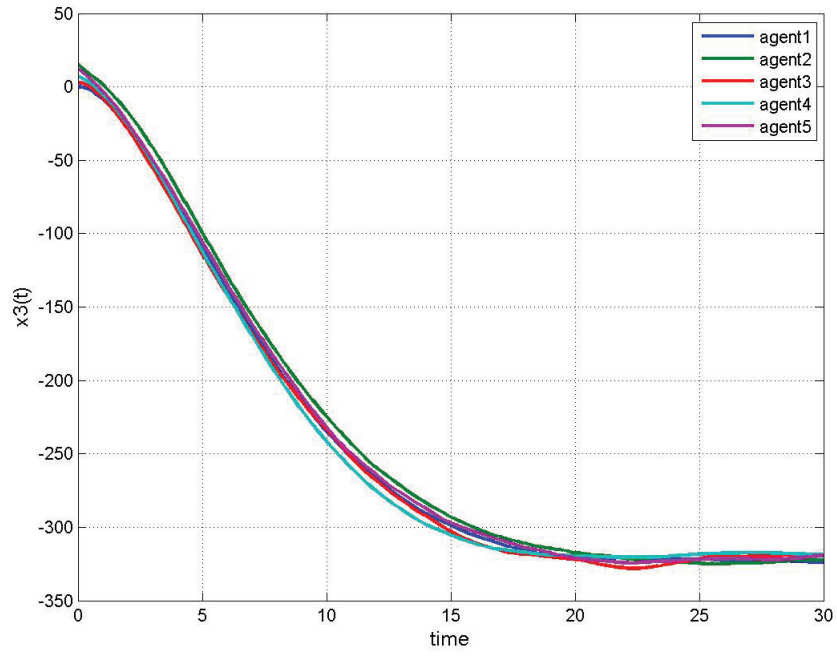


Figure 3.21: Synchronization of state 3 for static state feedback protocol subject to noise - gain set 2.

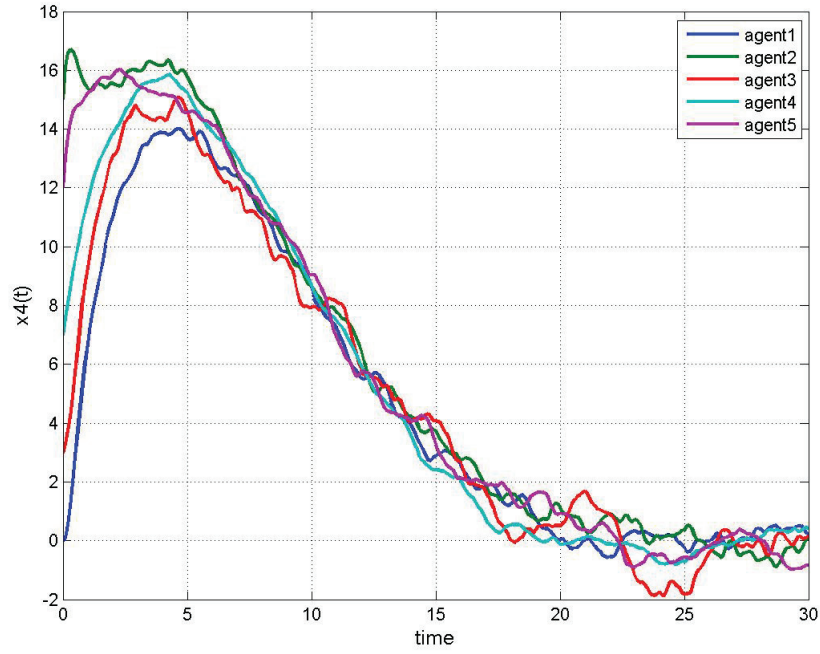


Figure 3.22: Synchronization of state 4 for static state feedback protocol subject to noise - gain set 2.

Figures 3.15 - 3.18 show that the control inputs still have high variation of amplitude and rate. From Figures 3.19 - 3.22, it can be seen that all states synchronize in a stochastic sense.

The Monte-Carlo simulation results of performance indices for 50 runs are summarized in Table 3.2.

Table 3.2: Monte-Carlo simulation results for performance indices of the single agents and network in the presence of noise and by employing the second set of gains.

	$PI_x$	$PI_u$	$PI_{xu}$
Agent 1	9	40	49
Agent 2	15	75	90
Agent 3	11	105	116
Agent 4	17	12	29
Agent 5	6	35	41
Network	10	50	60

The comparison of Tables 3.1 and 3.2 demonstrates that if the control gain is selected such that the contribution of the control input performance index in the total performance index with respect to the states performance index is decreased, it lowers the total performance index. This is expected as the measurement noise is fed back to the system through control input. By this selection of gain, even though compared to the previous case there is a significant change in the total performance

index, it is still high. Particularly, without filtering out the noise, it is not possible to achieve an acceptable result.

### 3.4 State Feedback Consensus and Kalman Filtering

As in the previous section, the process and measurement noises lead to a large covariance error of the control input and consequently, the states. To deal with this problem, one solution could be a linear quadratic Gaussian (LQG) approach, which is a generalization of LQR methodology. The LQG approach is employed when the states are not sensed directly. In this method, the controller is in the form of a feedback controller that uses the estimated states. Both the controller and the estimator gains are optimal in a sense that LQG cost function is minimized.

In this section, to deal with this problem, Kalman filtering is used for state estimation that are used in synchronization protocol. The state estimation protocol in this dissertation is not consensus based. In other words, each agent only uses its own measurement data to estimate the local states. However, there is an extensive amount of research in the literature for the consensus-based state estimation and sensor network ([108] and [109]). The employed Kalman-Bucy filter is

$$\begin{aligned}\dot{\hat{\mathbf{x}}}_i(t) &= A\hat{\mathbf{x}}_i(t) + B\mathbf{u}_i(t) + H_i(t)(z_i(t) - \hat{z}_i(t)), \quad i = 1, 2, \dots, N \\ \hat{z}_i(t) &= \hat{\mathbf{x}}_i(t)\end{aligned}\tag{3.30}$$

where

$$H_i(t) = P_i(t)R_i^{-1}(t), \quad i = 1, 2, \dots, N\tag{3.31}$$

and

$$\begin{aligned}\dot{P}_i(t) &= AP_i(t) + P_i(t)A^T + BQ_i(t)B^T - H_i(t)P_i(t) = \\ &AP_i(t) + P_i(t)A^T + BQ_i(t)B^T - P_i(t)R_i^{-1}(t)P_i(t), \quad i = 1, 2, \dots, N\end{aligned}\tag{3.32}$$

It is clear that  $H_i(t)$  cannot exist if  $R_i(t)$  is singular. Also, for a stable filter estimation,  $R_i(t)$  must be positive-definite. As in equation (3.31),  $H_i(t)$  will result in a minimum-error states' covariance matrix. Moreover, the equation (3.32) leads to a common Kalman gain  $H(t)$  for all  $i = 1, 2, \dots, N$ , since the agents are homogeneous.

### 3.4.1 Simulation Results

In this section, the Monte-Carlo simulation results of equation (3.30) by employing a single-input model (3.13) are shown. Firstly, the results of state estimation are presented. Then, the consensus achievement of the entire states will be shown. For simulation purposes, the spectral density matrix of the process and measurement noises are considered as  $Q = I_{(m \times m)}$  and  $R = 0.5 \times I_{(n \times n)}$ , respectively. Furthermore, equation (3.32) has been solved in steady state. In other words, it is obtained by solving an algebraic Riccati equation. The Kalman-Bucy gain  $H$  obtained out of this equation is still optimum in steady state. Equations (3.33) and (3.34) represent the controller and the Kalman filter gains.

$$K = \begin{bmatrix} 0.2790 & -0.6279 & 1.0000 & -4.0064 \end{bmatrix} \quad (3.33)$$

$$H = \begin{bmatrix} 0.0869 & -0.1784 & 0.0041 & -0.0106 \\ -0.1784 & 0.3767 & -0.0248 & 0.0506 \\ 0.0041 & -0.0248 & 1.0212 & -0.2782 \\ -0.0106 & 0.0506 & -0.2782 & 0.1452 \end{bmatrix} \quad (3.34)$$

The results for one of the runs under Monte-Carlo simulations are shown in Figures 3.23 - 3.31.

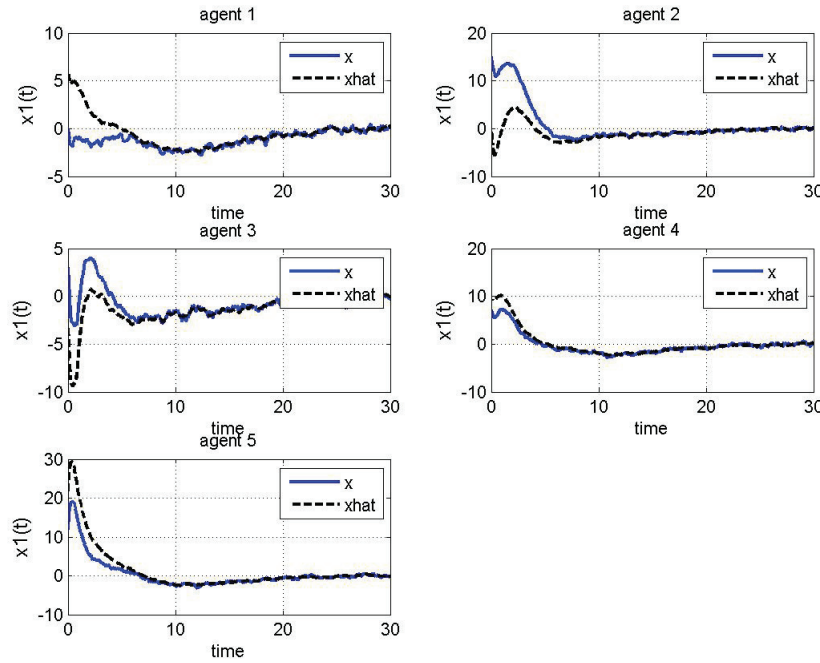


Figure 3.23: State estimation of state 1 for static state feedback protocol subject to noise and by employing Kalman filter.

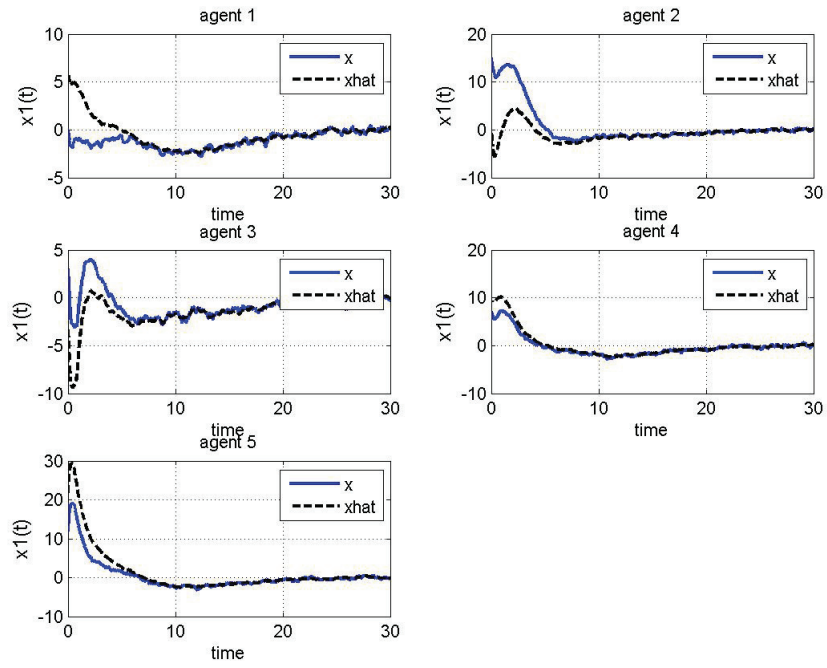


Figure 3.24: State estimation of state 2 for static state feedback protocol subject to noise and by employing Kalman filter.

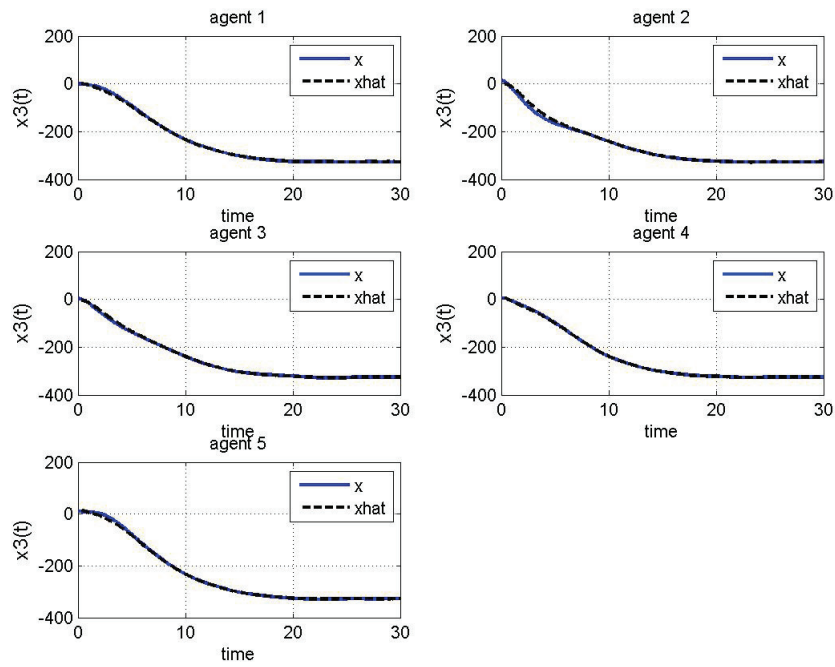


Figure 3.25: State estimation of state 3 for static state feedback protocol subject to noise and by employing Kalman filter.

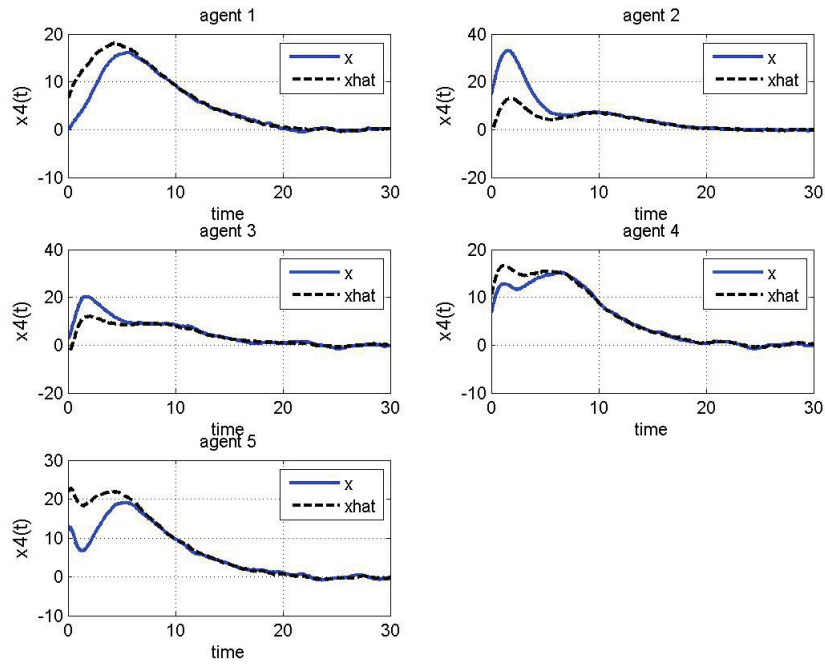


Figure 3.26: State estimation of state 4 for static state feedback protocol subject to noise and by employing Kalman filter.

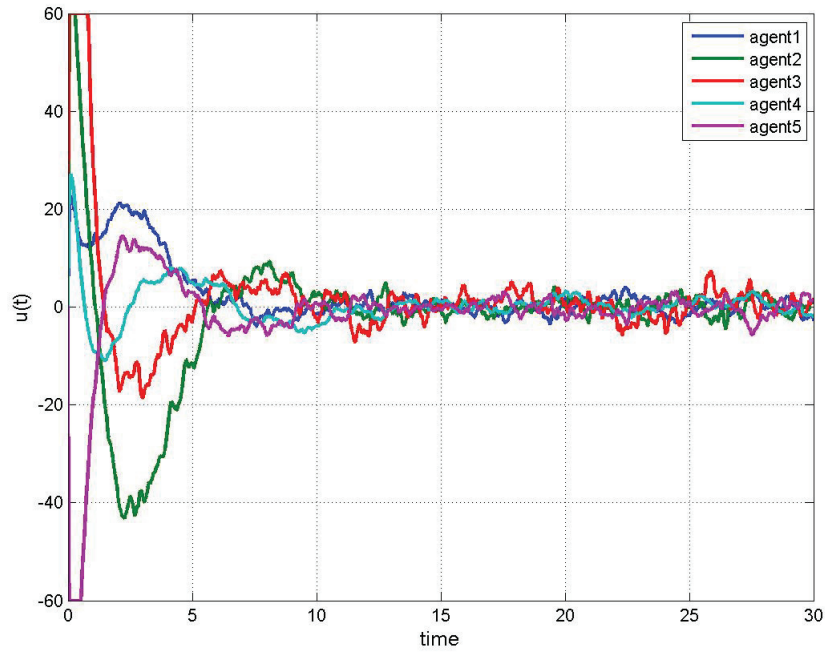


Figure 3.27: Control input signals  $u(t)$  for static state feedback protocol subject to noise and by employing Kalman filter.

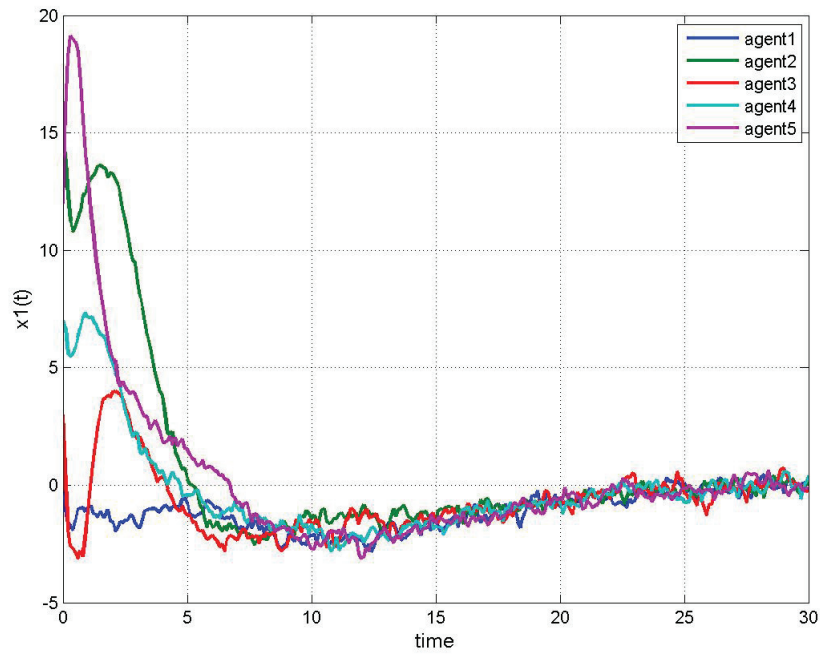


Figure 3.28: Synchronization of state 1 for static state feedback protocol subject to noise and by employing Kalman filter.

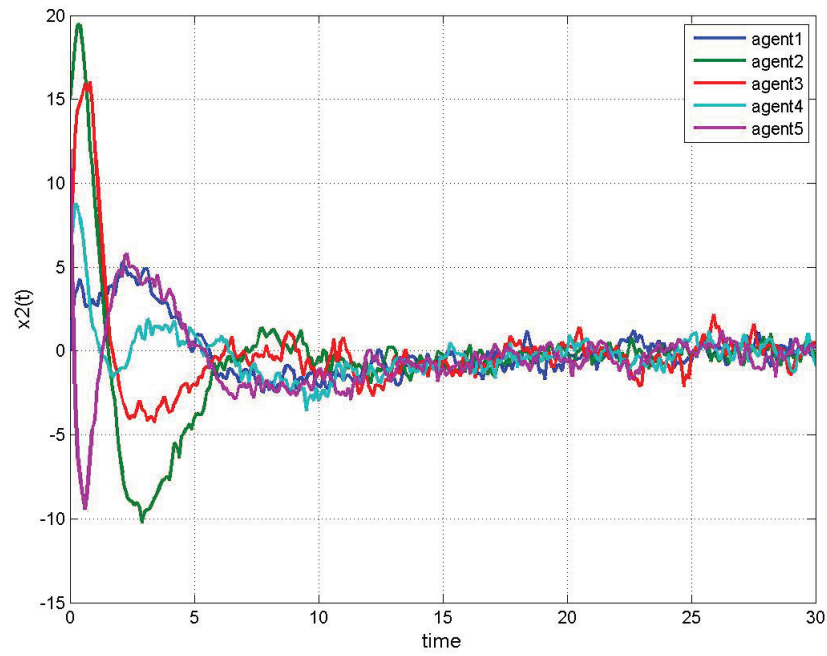


Figure 3.29: Synchronization of state 2 for static state feedback protocol subject to noise and by employing Kalman filter.

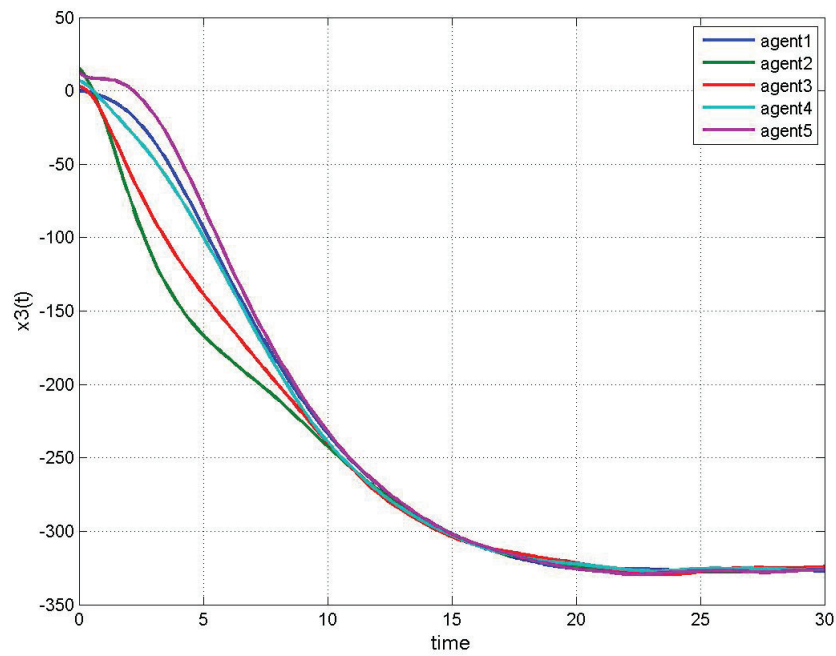


Figure 3.30: Synchronization of state 3 for static state feedback protocol subject to noise and by employing Kalman filter.

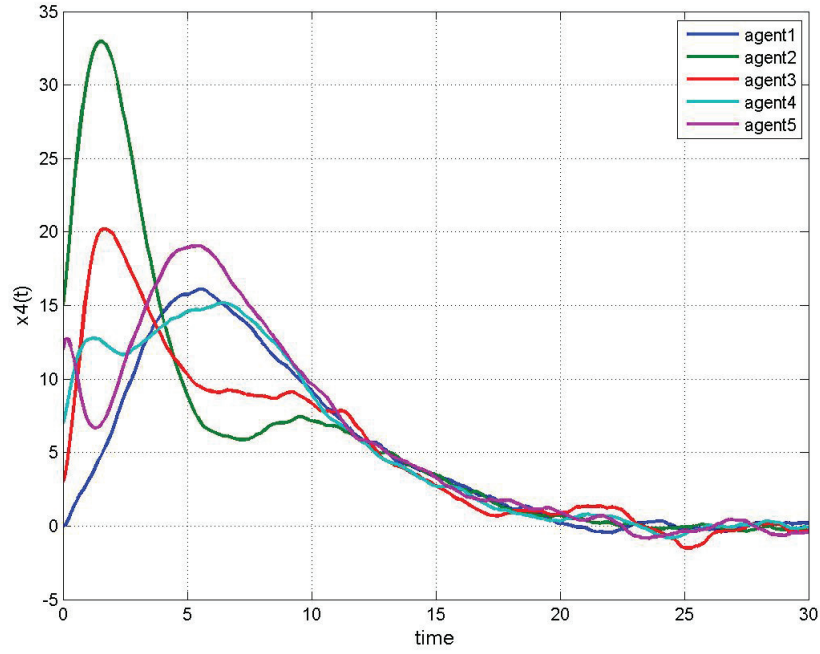


Figure 3.31: Synchronization of state 4 for static state feedback protocol subject to noise and by employing Kalman filter.

The Monte-Carlo simulation results of the performance indices for 50 runs are summarized in Table 3.3.

Table 3.3: Monte-Carlo simulation results for performance indices of the single agents and network in presence of noise and by employing Kalman filtering.

	$PI_x$	$PI_u$	$PI_{xu}$
Agent 1	1.5	20.5	21.0
Agent 2	2.0	60.0	62.0
Agent 3	3.0	15.0	18.0
Agent 4	2.0	7.0	9.0
Agent 5	2.5	12.0	14.5
Network	2.5	27.0	29.5

Comparison of the performance indices in Table 3.3 and the ones in Tables 3.1 and 3.2 shows how much the control input and the states variances at steady state are improved by employing a Kalman filter.

In addition to its steady state performance improvement, Kalman filtering considerably improves the variance of the control input and the states in transient as in Figure 3.27 - 3.31.

## 3.5 State Feedback Consensus Subject to the Agents Dynamics

### Uncertainty

In this section, the effect of agents' dynamics uncertainty on the consensus achievement will be studied. As previously discussed, the dynamics of the underwater vehicles are subject to the model uncertainties. These uncertainties could be due to the added mass terms, the hydrodynamic forces and the environmental disturbances, which are uncertain and difficult to predict. This leads to a high uncertainty in AUV model parameters.

Here, it is assumed that the uncertainty happens only in the hydrodynamic parameters of model (3.2). In a similar manner, the effect of uncertainty of the added mass terms could be analyzed. The hydrodynamic terms are modeled through  $Z_w$ ,  $Z_q$ ,  $M_w$ , and  $M_q$ . Considering the contribution of these model uncertainties, equation (3.3) becomes:

$$\dot{\mathbf{x}}_i = \begin{bmatrix} a''_{11} & a''_{12} & 0.0 & a_{14} \\ a''_{21} & a''_{22} & 0.0 & a_{24} \\ 1.0 & 0 & 0 & a_{34} \\ 0 & 1 & 0 & 0 \end{bmatrix} \mathbf{x}_i + B\mathbf{u}_i \quad (3.35)$$

In order to investigate the simulation-based sensitivity analysis of the entire control system, different levels of uncertainty are applied. This could be modeled as

$$a''_{ij} = a_{ij} + \Delta a_{ij} \quad (3.36)$$

where  $\Delta$  may vary between 0% – 100%, and  $a''_{ij}$  denotes the  $ij$ th entry of the state-space dynamics matrix  $A$  subject to the uncertainty. In each simulation scenario, the uncertainty exists only on one entry of the matrix  $A$  of agent 1. The simulation results are summarized in Figures 3.32 - 3.43.

Similar indices to (3.26)-(3.28) are also used throughout the thesis to measure the performance of the system subject to faults and dynamic uncertainties as

- Control input performance index

$$PI_u = \frac{1}{T - T_1} \int_{T_1}^T \mathbf{u}_i^2(t) dt \quad (3.37)$$

- State performance index

$$PI_x = \frac{1}{T - T_1} \int_{T_1}^T \mathbf{x}_i^2(t) dt \quad (3.38)$$

- Total performance index

$$PI_{\text{xu}} = \frac{1}{T - T_1} \int_{T_1}^T (\mathbf{u}_i^2(t) + \mathbf{x}_i^2(t)) dt \quad (3.39)$$

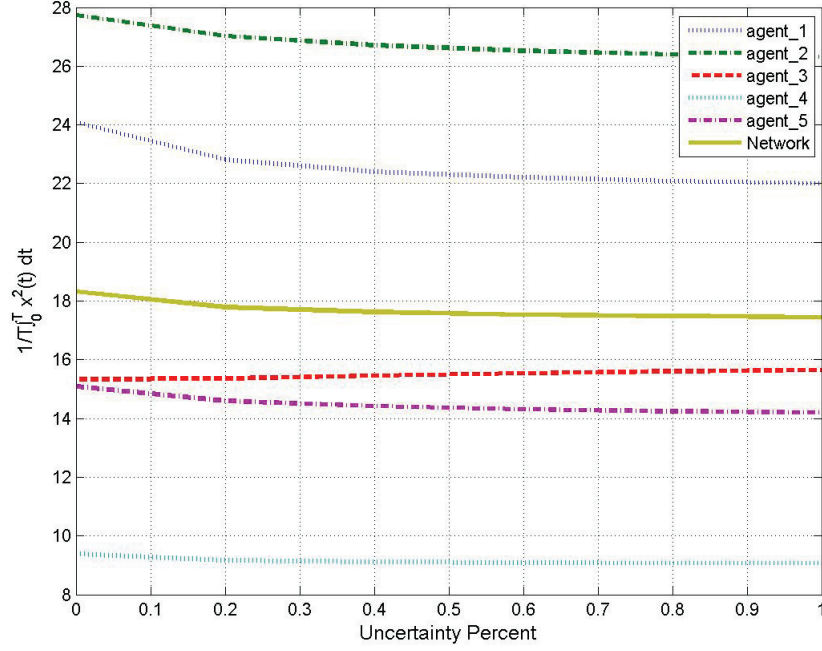


Figure 3.32: State performance index for static state feedback protocol subject to uncertainty on A(1,1).

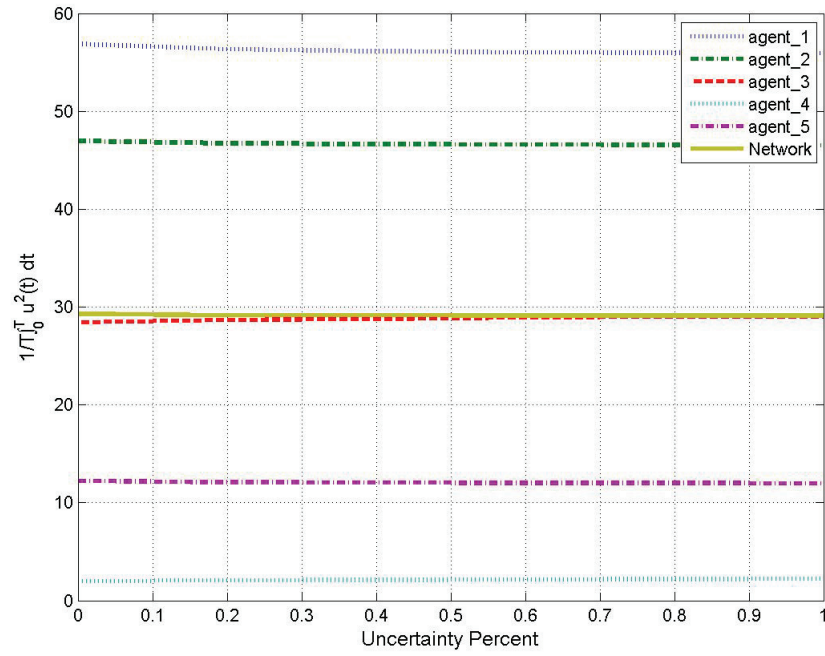


Figure 3.33: Control input performance index for static state feedback protocol subject to uncertainty on  $A(1,1)$ .

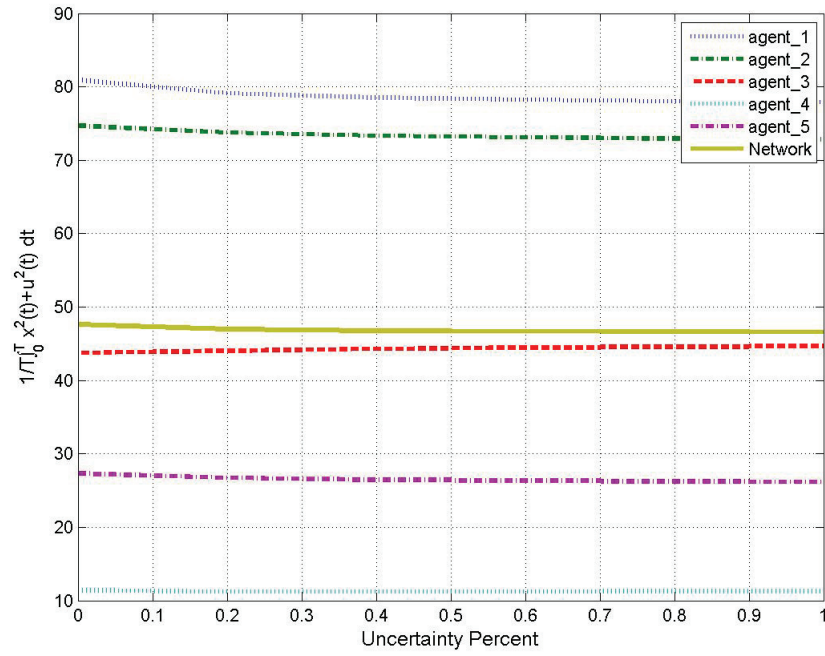


Figure 3.34: Total performance index for static state feedback protocol subject to uncertainty on  $A(1,1)$ .

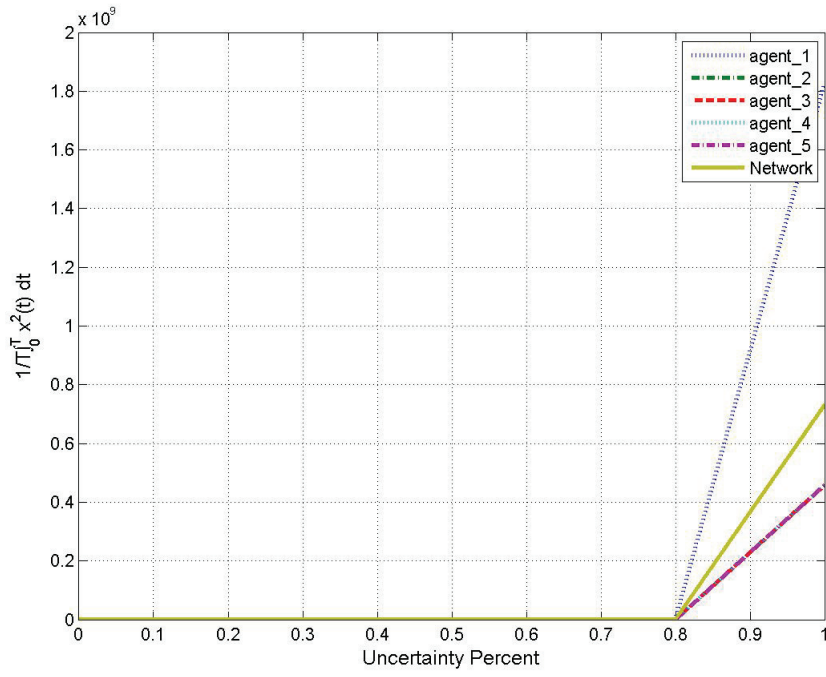


Figure 3.35: State performance index for static state feedback protocol subject to uncertainty on A(1,2).

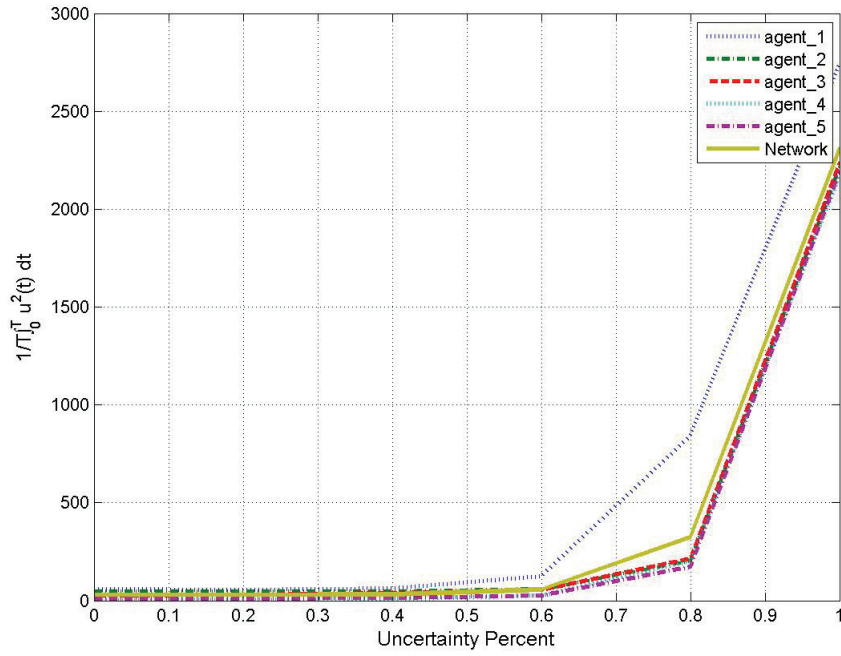


Figure 3.36: Control input performance index for static state feedback protocol subject to uncertainty on A(1,2).

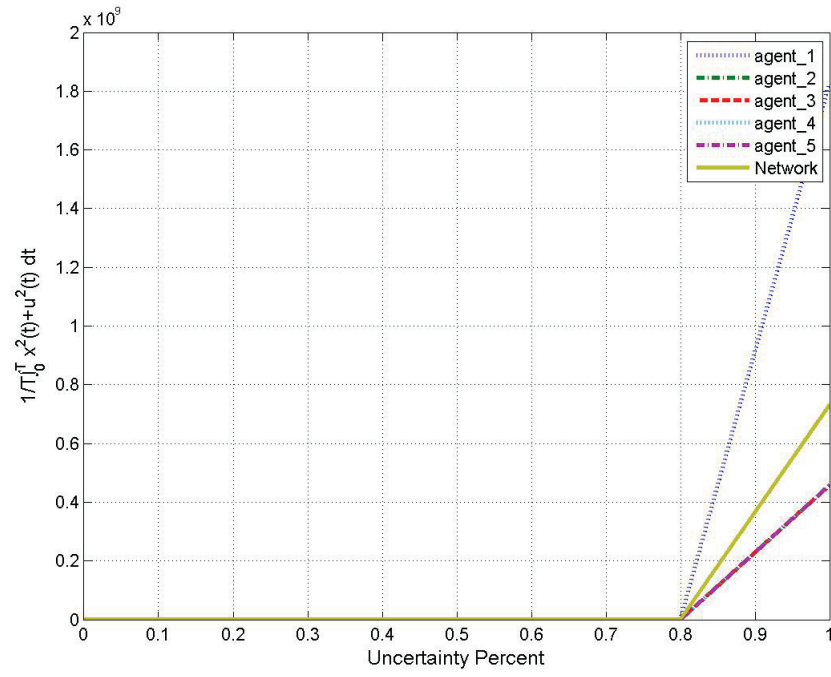


Figure 3.37: Total performance index for static state feedback protocol subject to uncertainty on A(1,2).

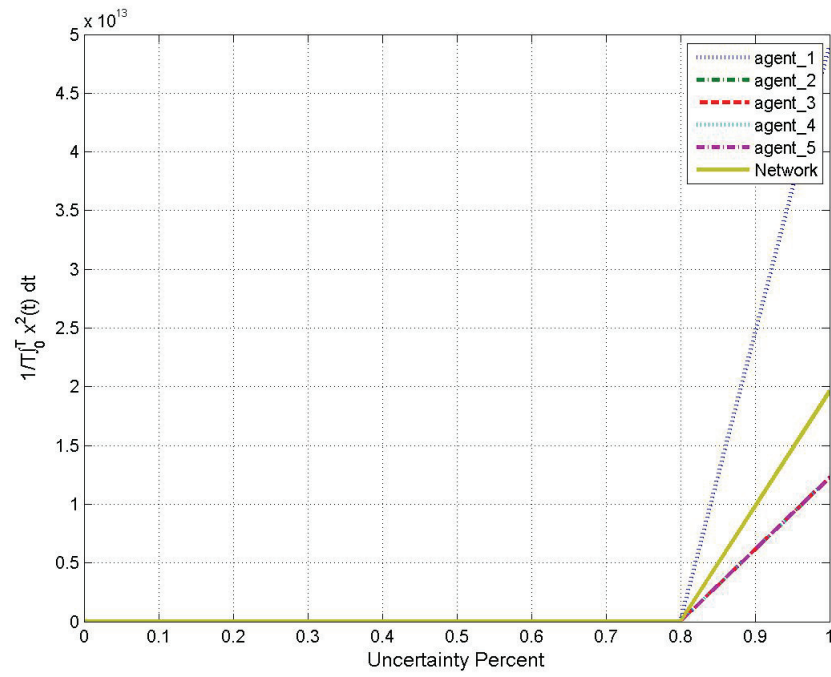


Figure 3.38: State performance index for static state feedback protocol subject to uncertainty on A(2,1).

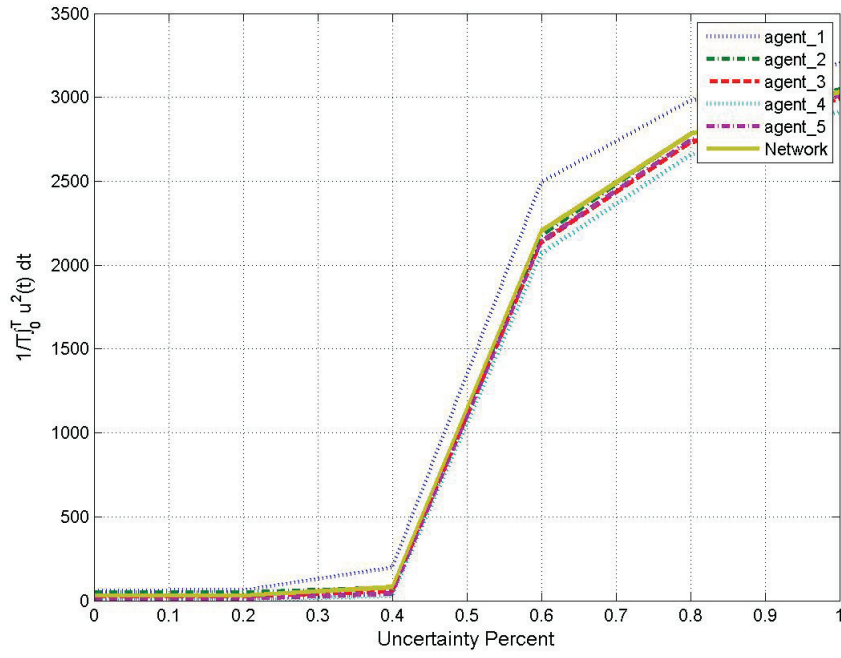


Figure 3.39: Control input performance index for static state feedback protocol subject to uncertainty on A(2,1).

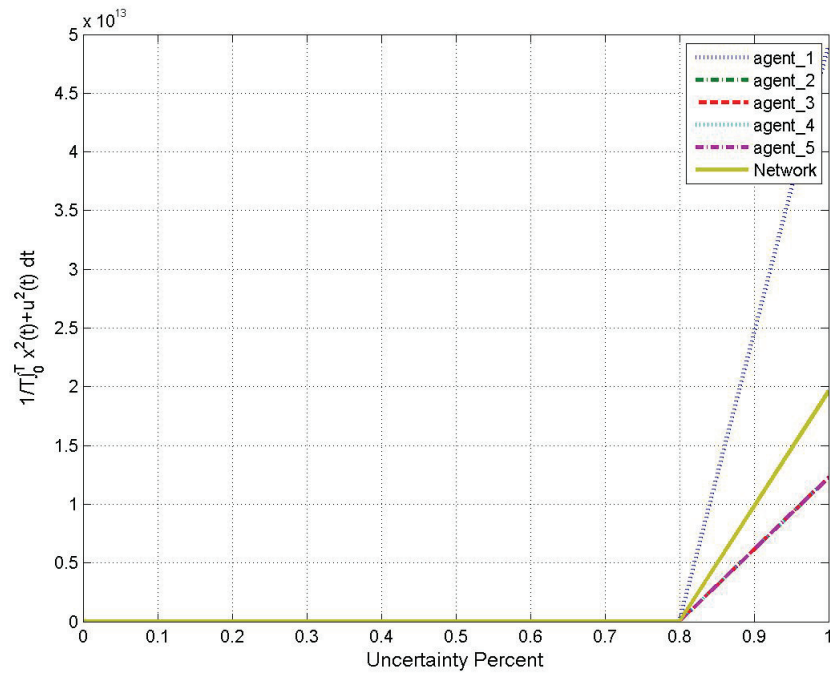


Figure 3.40: Total performance index for static state feedback protocol subject to uncertainty on A(2,1).

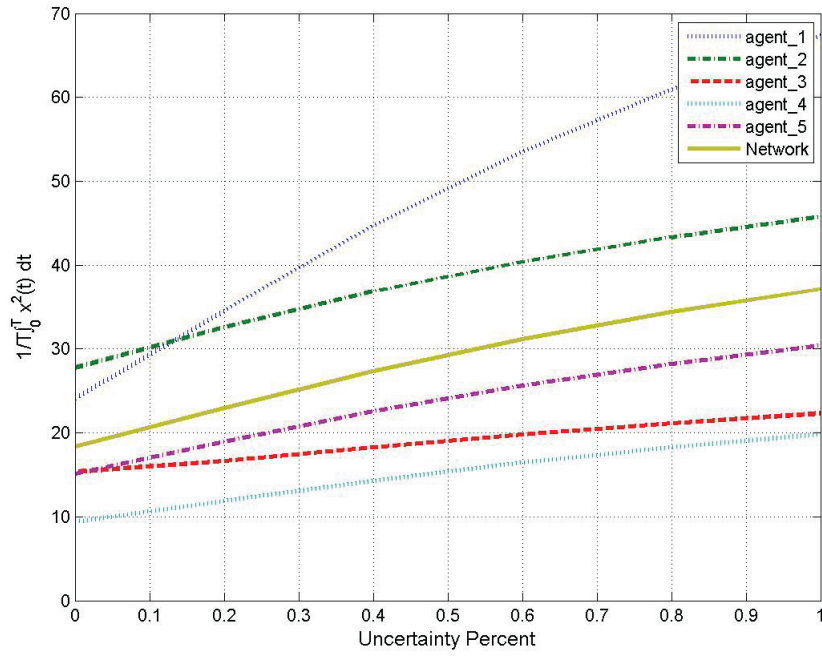


Figure 3.41: State performance index for static state feedback protocol subject to uncertainty on A(2,2).

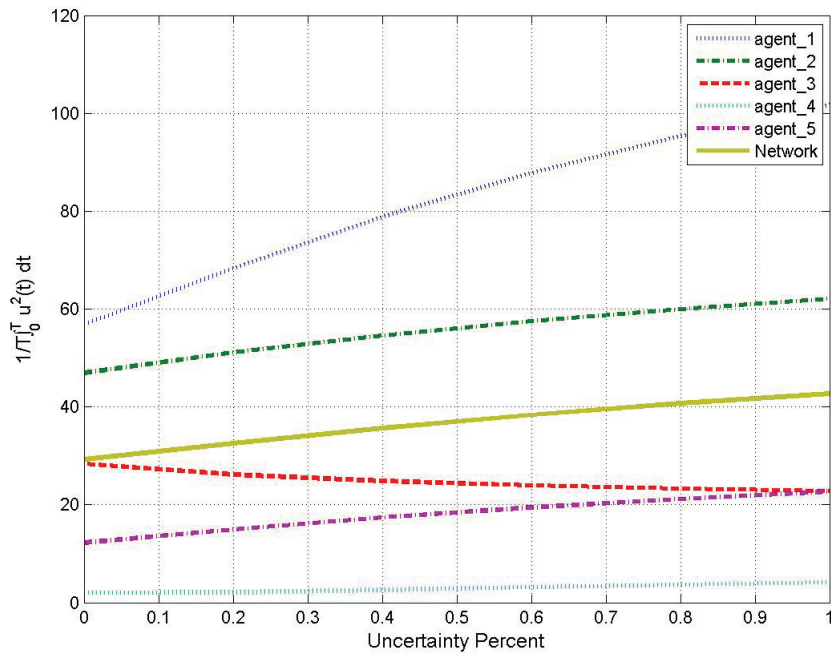


Figure 3.42: Control input performance index for static state feedback protocol subject to uncertainty on A(2,2).

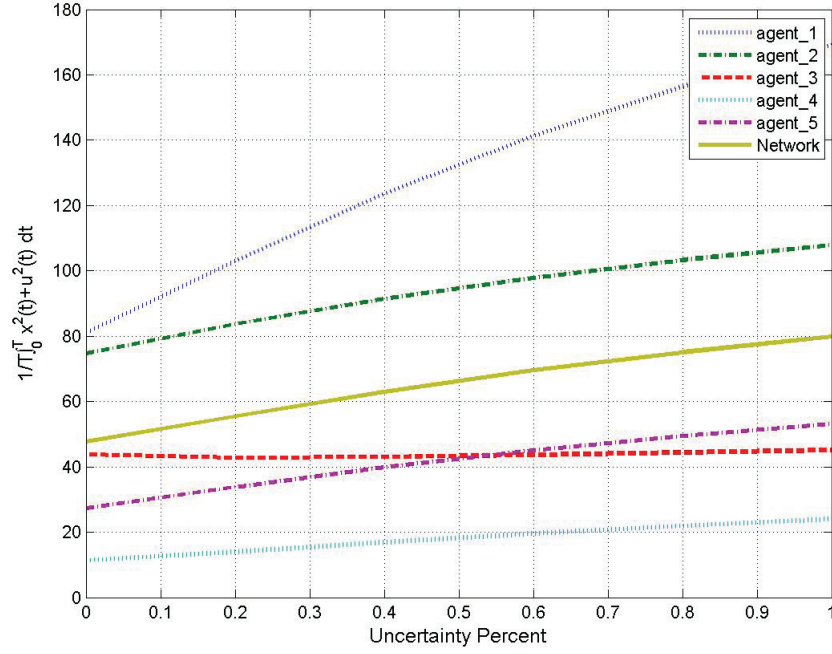


Figure 3.43: Total performance index for static state feedback protocol subject to uncertainty on  $A(2,2)$ .

The simulation results in Figures 3.32 - 3.34 and 3.41 - 3.43 illustrate that the consensus is not sensitive to the changes in  $A(1,1)$  and  $A(2,2)$ . However, Figures 3.35 - 3.37 and 3.38 - 3.40 demonstrate the sensitivity of the system's overall stability and performance to the uncertainties on  $A(1,2)$  and  $A(2,1)$ . This shows that when these uncertainties go beyond certain values, the entire system becomes unstable.

### 3.6 Fault-Tolerant Static State Feedback Synchronization Protocol Subject to the LOE Fault

In this section, the effects of actuator LOE fault in a team of multi-agent systems with marginally stable linear time-invariant dynamics, and under a fixed network topology, are addressed. It is assumed that agents are equipped with a single actuator. It is shown that by applying the static protocol of equation (3.6) and selecting a robust control gain  $K$ , the synchronization can still be met. Without loss of generality, it is assumed that the fault occurs in agent 1. From equations

(2.23) and (2.25), one knows that when there is a LOE fault, its effect can be modeled as

$$\mathbf{u}_1^f(t) = -K \sum_{j \in \mathcal{N}_1} \gamma_1 a_{1j} (\mathbf{x}_1(t) - \mathbf{x}_j(t)) \quad (3.40)$$

where  $\gamma_1$  represents the actual effectiveness factor of the faulty actuator.

From equation (3.40) it follows that the change in the effectiveness factor of the actuator in agent 1 can be modeled as a multiplication of the associated row of the original Laplacian matrix ( $L$ ) by  $\gamma_1$ . Therefore, the augmented state-space representation of the faulty system becomes

$$\dot{\sigma}(t) = (I_{N-1} \otimes A - (L_r + I \alpha_f^T) \otimes BK) \sigma(t) \quad (3.41)$$

where  $\alpha_f = \gamma_1 \alpha$ .

Since the matrix  $A$  is marginally stable, one knows that there exists a matrix  $P = P^T > 0$  such that

$$A^T P + P A \leq 0 \quad (3.42)$$

Knowing this, one possible selection for  $K$  is to choose it as

$$K = B^T P \quad (3.43)$$

where  $P$  is a possible solution of equation (3.42) and  $K$  is the agents' control gain matrix.

It will be shown that with this selection of  $K$ , even if there is an actuator LOE fault, which is modeled as a change in the Laplacian matrix ( $L$ ), the synchronization can still be met.

**Remark 3.7.** *The control gain matrix  $K$  is robust such that it does not change after fault occurrence. It is also common for all agents.*

The effect of a LOE fault in an agent's actuator is modeled by multiplying the associated row of the Laplacian matrix by a constant that represents the effectiveness of the faulty actuator. Hence,  $L' = [l'_{ij}]$  is still a zero row sum matrix, with  $l'_{ii} = \sum_{j \in \mathcal{N}_i} a'_{ij}$ , and  $l'_{ij} = -a'_{ij}$  for  $i \neq j$ . Therefore, the graph associated with  $L'$ , which is interpreted as the Laplacian matrix after fault occurrence, has a spanning tree. Using the results from Lemma 2.1,  $L'$  has a single zero eigenvalue and the rest of its eigenvalues have positive real parts.

**Theorem 3.1.** *The loss of effectiveness (LOE) fault in a team of marginally stable linear multi-agent systems, as described by equation (3.4), but with a single input channel, under a fixed*

connected interaction graph  $\mathcal{G}$ , and by using the control protocol of (3.6) does not change the synchronization achievement, if matrix  $P = P^T > 0$  in equation (3.43) is chosen such that

1. It satisfies  $A^T P + PA \leq 0$ .
2. For those values of  $y \in \mathbb{R}^n$  which makes  $y^T(A^T P + PA)y = 0$ ,  $y^T(PB(PB)^T)y \neq 0$ .

*Proof.* By applying the results obtained in Section 3.2, the system described by equation (3.41) (where  $\alpha_f$  reflects the adjusted row of Laplacian due to the effects of the fault modeling) will stabilize if  $A - \lambda'_i BK$  is Hurwitz for all  $\lambda'_i = \sigma'_i + jw'_i$  ( $i = 2, \dots, N$ ), where  $\lambda'_i$ 's are the eigenvalues of  $L_r + I.\alpha_f^T$ . Using the similarity transformation properties between  $L_r + I.\alpha_f^T$  and  $L'$  as in Section 3.2, all eigenvalues of  $L_r + I.\alpha_f^T$  have positive real parts.

Specifically, it can be shown from equation (3.10) that

$$\begin{aligned}
 & ((A - (\sigma'_i + jw'_i)BK)^H P + P(A - (\sigma'_i + jw'_i)BK)) = \\
 & ((A - (\sigma'_i + jw'_i)BB^T P)^H P + P(A - (\sigma'_i + jw'_i)BB^T P)) = \\
 & ((A - (\sigma'_i - jw'_i)BB^T P)^T P + P(A - (\sigma'_i + jw'_i)BB^T P)) = \\
 & (A^T P + PA) - 2\sigma'_i((PB)(PB)^T) < 0
 \end{aligned} \tag{3.44}$$

Since it is assumed that the underlying graph is connected,  $\sigma'_i = \text{Re}\{\lambda'_i\}$  will be positive for all  $i = 2, \dots, N$ . Moreover, given that condition 2 is satisfied, the second term on the right hand side of equation (3.44) will be negative definite. Therefore, one may easily conclude that even if there is a change in the Laplacian matrix, regardless of the value of  $\sigma'_i$  (as long as it is positive, which means as long as the graph is connected), equation (3.44) holds for all  $i = 2, \dots, N$ . Therefore, using the observations described above, if the control gain  $K$  of the healthy system is chosen as  $K = B^T P$  where  $P$  satisfies (3.42), it is guaranteed that regardless of the effectiveness of the fault and the values of  $\lambda'_i$ , the faulty system represented by the augmented states-space representation of (3.41) will still be stable.  $\square$

**Remark 3.8.** It should be noted that the protocol proposed in this section does not necessarily need to be applied to undirected networks. In other words, Laplacian matrix could be either symmetric or non-symmetric.

**Remark 3.9.** *By applying the proposed protocol even if more than one agent has LOE fault, synchronization can still be achieved. This can simply be illustrated by modeling the effect of LOE faults as multiplying the associated rows of Laplacian matrix by the effectiveness factor of faulty actuators. Then, similar to one agent's actuator fault, it can easily be shown that (3.44) still holds even though two of the agents with single actuator are faulty.*

If any of the conditions in Theorem 3.1 are not satisfied, the above fault-tolerant protocol may not be valid any more. In this case, a good alternative is to redesign the control gain  $K$  after fault occurrence. To this end, an FDD block is required to detect the fault and estimate its severity. The question is how accurate the estimated severity is. In the following section, it is assumed that there is a perfect FDD block that detects the fault and estimates its severity. Thus, using the information that is obtained through the FDD block, a reconfigurable protocol is proposed.

### 3.7 Fault-Tolerant Static State Feedback Synchronization Protocol Subject to the Float Fault

In this section, it will be shown that when there is an actuator float fault, by applying the static protocol in equation (3.6) and selecting a robust control gain  $K$  how the synchronization could still be met. From equations (2.23) and (2.25), one knows that when there is a float fault in the actuator of agent 1, the effect of this fault can be modeled as

$$\mathbf{u}_1^f(t) = 0 \quad (3.45)$$

Thus, it follows that the float fault in agent 1 can be modeled as a zero row of the original Laplacian matrix ( $L$ ). Therefore, its effect on (3.7) will be

$$\alpha = 0 \quad (3.46)$$

which makes the augmented state-space representation of the faulty system to reduce to

$$\dot{\sigma}(t) = (I_{N-1} \otimes A - L_r \otimes BK)\sigma(t) \quad (3.47)$$

Since the matrix  $A$  is marginally stable, one knows that there exists a matrix  $P = P^T > 0$  such that

$$A^T P + P A \leq 0 \quad (3.48)$$

Knowing this, one possible selection for  $K$  is to choose it as

$$K = B^T P \quad (3.49)$$

where  $P$  is one possible solution of (3.48).

**Theorem 3.2.** *The float fault in an agent of a team of marginally-stable linear multi-agent systems, as described by equation (3.4), but with a single input, under a fixed interaction graph  $\mathcal{G}$ , and by using the protocol (3.6) with control gain (3.49) does not change the synchronization achievement if*

1. *The matrix  $P = P^T > 0$  is chosen such that it satisfies  $A^T P + PA \leq 0$ .*
2. *For those values of  $y \in \mathbb{R}^n$  which makes  $y^T(A^T P + PA)y = 0$ ,  $y^T(PB(PB)^T)y \neq 0$ .*
3. *There is a spanning tree in interaction graph rooted from faulty agent.*

*Proof.* From (3.10), one knows that the system represented by (3.47) is asymptotically stable if

$$\begin{aligned} ((A - (\sigma_i^r + jw_i^r)BK)^H P + P(A - (\sigma_i^r + jw_i^r)BK)) = \\ (A^T P + PA) - 2\sigma_i^r((PB)(PB)^T) < 0 \end{aligned} \quad (3.50)$$

where  $\lambda_i^r = \sigma_i^r + jw_i^r$  ( $i = 1, \dots, N - 1$ ) are the eigenvalues of  $L_r$ .

From (3.50), it can be seen that if the non-equality is held for the Laplacian eigenvalue with smallest real part, it will also be valid for the remaining Laplacian eigenvalues. Therefore, to achieve synchronization, only the validity of (3.50) for the smallest eigenvalue will be considered.

Since there is a spanning tree rooted in the faulty agent, at least one of the row sum of matrix  $L_r$  in (3.8) is nonzero. Therefore, using the Gersgorin disk theorem one may easily conclude that the possible locations of the eigenvalues of  $L_r$  will be on the open right-half plan ( $\sigma_i^r = \text{Re}\{\lambda_i^r\} > 0$ ,  $i = 2, \dots, N$ ). Thus, knowing  $A^T P + PA \leq 0$ , and given that condition 2 is satisfied, one may conclude that the faulty system represented by (3.47) will still be asymptotically stable. Hence, synchronization will be achieved.  $\square$

**Remark 3.10.** *It is clear that if more than one agent is subject to the float fault, synchronization cannot be achieved.*

**Remark 3.11.** *It should be noted that the proposed protocol in this section does not necessarily need to be applied to an undirected network. In other words, the Laplacian matrix could be either symmetric or non-symmetric. For a non-symmetric Laplacian matrix, the eigenvalues in equation (3.50) are complex values  $(\sigma_i^r + jw_i^r)$ .*

When a  $P$  matrix does not exist that also satisfies equation (3.48), this methodology cannot be applied. In this case, an alternative approach is to redesign  $K$  such that equation (3.47) is stable. From (3.8), one knows that  $L_r$  is a positive definite matrix with all known elements. Therefore, its eigenvalues could easily be obtained, and the simultaneous LMI of (3.12) is applied to redesign the control gain  $K$ .

**Remark 3.12.** *It is worth nothing that if the original Laplacian matrix  $L$  is symmetric,  $L_r$  will be symmetric as well.*

**Remark 3.13.** *The matrix  $L_r$  stands for the Laplacian matrix of the network by removing the incoming link of the faulty agent. For each faulty scenario (the float fault in different agents), the control gain matrix  $K$  could be calculated off-line. The advantage of using the similarity transformation and consequently defining  $L_r$  as (3.8) is that the effect of fault on Laplacian matrix  $L$  can be isolated as a change in  $\alpha$ . In this way,  $L_r$  is not affected.*

**Remark 3.14.** *In this section, the control gain matrix  $K$  is common for all agents. The control structure is robust such that its gain does not change after fault occurrence.*

When a float fault happens in agent 1, its control effort becomes zero. Therefore, its final value can be obtained as

$$\lim_{t \rightarrow \infty} \mathbf{x}_1(t) = (\lim_{t \rightarrow \infty} \exp^{-At}) \mathbf{x}_1(t_f) \quad (3.51)$$

where  $t_f$  represents the fault occurrence time.

Finally, if  $K$  is designed such that (3.47) is stable, the consensus will be reached and we will have

$$\lim_{t \rightarrow \infty} \mathbf{x}_i(t) = \lim_{t \rightarrow \infty} \mathbf{x}_j(t) = \lim_{t \rightarrow \infty} \mathbf{x}_1(t) = (\lim_{t \rightarrow \infty} \exp^{-At}) \mathbf{x}_1(t_f) \quad (3.52)$$

Note that  $A$  has a zero eigenvalue which means  $\lim_{t \rightarrow \infty} \exp^{-At} \neq 0$ .

## 3.8 Simulation Results for the Float and LOE Faults

In this section, the simulation results of Sections 3.6 and 3.7 for a change of the effectiveness factor of the actuator between 0 and 1 will be shown.

### 3.8.1 AUV with single-input channel

For simulation purposes, the model (3.13) and the control gain designed in (3.14) are used. In this case, the change in the effectiveness factor of the single actuator could be modeled as a change in the row of the Laplacian matrix associated with the faulty agent. As it can be seen, there is no pattern for the change of the Laplacian matrix eigenvalues.

Table 3.4: Variation of the real part of Laplacian eigenvalues with change of the actuator effectiveness in agent 1.

$\gamma$	0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1
$\sigma_1$	0.737	0.845	0.927	0.985	1.025	1.054	1.075	1.091	1.103	1.113	1.121
$\sigma_2$	1.484	1.577	1.697	1.840	2.000	2.171	2.349	2.530	2.708	2.872	3.000
$\sigma_3$	3.183	3.184	3.186	3.189	3.192	3.197	3.203	3.214	3.232	3.269	3.347
$\sigma_4$	4.596	4.593	4.590	4.587	4.583	4.578	4.572	4.565	4.557	4.546	4.532
Consensus	YES	YES	YES	YES	YES	YES	YES	YES	YES	YES	YES

Table 3.5: Variation of the real part of Laplacian eigenvalues with change of the actuator effectiveness in agent 2.

$\gamma$	0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1
$\sigma_1$	1.000	1.144	1.160	1.137	1.130	1.127	1.125	1.123	1.122	1.121	1.121
$\sigma_2$	1.000	1.144	1.413	1.717	2.000	2.273	2.533	2.778	3.000	3.000	3.000
$\sigma_3$	3.000	3.000	3.000	3.000	3.000	3.000	3.000	3.000	3.000	3.192	3.347
$\sigma_4$	4.000	4.012	4.027	4.046	4.070	4.101	4.142	4.199	4.278	4.387	4.532
Consensus	YES	YES	YES	YES	YES	YES	YES	YES	YES	YES	YES

Table 3.6: Variation of the real part of Laplacian eigenvalues with change of the actuator effectiveness in agent 3.

$\gamma$	0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1
$\sigma_1$	0.268	0.829	1.023	1.215	1.262	1.177	1.152	1.138	1.130	1.125	1.121
$\sigma_2$	1.000	0.829	1.023	1.215	1.546	2.000	2.378	2.716	3.000	3.000	3.000
$\sigma_3$	3.000	3.000	3.000	3.000	3.000	3.000	3.000	3.000	3.000	3.211	3.347
$\sigma_4$	3.732	3.742	3.754	3.770	3.792	3.823	3.870	3.946	4.070	4.264	4.532
Consensus	YES	YES	YES	YES	YES	YES	YES	YES	YES	YES	YES

Table 3.7: Variation of the real part of Laplacian eigenvalues with change of the actuator effectiveness in agent 4.

$\gamma$	<b>0</b>	<b>0.1</b>	<b>0.2</b>	<b>0.3</b>	<b>0.4</b>	<b>0.5</b>	<b>0.6</b>	<b>0.7</b>	<b>0.8</b>	<b>0.9</b>	<b>1</b>
$\sigma_1$	0.065	0.168	0.272	0.376	0.481	0.586	0.691	0.798	0.904	1.012	1.121
$\sigma_2$	3.000	3.000	3.000	3.000	3.000	3.000	3.000	3.000	3.000	3.000	3.000
$\sigma_3$	3.463	3.454	3.445	3.435	3.425	3.414	3.403	3.390	3.377	3.363	3.347
$\sigma_4$	4.473	4.478	4.483	4.489	4.494	4.500	4.506	4.512	4.519	4.525	4.532
Consensus	YES	YES	YES	YES	YES	YES	YES	YES	YES	YES	YES

Table 3.8: Variation of the real part of Laplacian eigenvalues with change of the actuator effectiveness in agent 5.

$\gamma$	<b>0</b>	<b>0.1</b>	<b>0.2</b>	<b>0.3</b>	<b>0.4</b>	<b>0.5</b>	<b>0.6</b>	<b>0.7</b>	<b>0.8</b>	<b>0.9</b>	<b>1</b>
$\sigma_1$	1.000	1.121	1.121	1.121	1.121	1.121	1.121	1.121	1.121	1.121	1.121
$\sigma_2$	1.121	1.200	1.400	1.600	1.800	2.000	2.200	2.400	2.600	2.800	3.000
$\sigma_3$	3.347	3.347	3.347	3.347	3.347	3.347	3.347	3.347	3.347	3.347	3.347
$\sigma_4$	4.532	4.532	4.532	4.532	4.532	4.532	4.532	4.532	4.532	4.532	4.532
Consensus	YES	YES	YES	YES	YES	YES	YES	YES	YES	YES	YES

As an initial conjecture, by decreasing the effectiveness factor of the single actuator of an agent, and modeling it as a change of the associated row of the Laplacian matrix, the smallest real part of none-zero eigenvalue of Laplacian matrix will become smaller. However, this is not the case as it can be seen from the Tables 3.4 - 3.8.

Tables 3.4 - 3.8 show that the controller structure and its gain is robust with respect to LOE and the float faults.

### 3.8.2 AUV with Multi-input Channel

In this section, a multi-input model (with stern and bow as inputs) is employed. The linearized model about the equilibrium point,  $\mathbf{x}_0 = [w_o, q_o, z_o, \theta_o]' = [0, 0, 0, 0]'$  and  $\mathbf{u}_o = \sigma_s = 0$ , and at a forward velocity of 2.0 m/s is

$$\dot{\mathbf{x}}_i = \begin{bmatrix} -1.400 & 2.763 & 0.0 & 0.078 \\ 2.108 & -5.419 & 0.0 & -0.312 \\ 1.0 & 0 & 0 & -2.0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \mathbf{x}_i + \begin{bmatrix} -0.797 & -0.201 \\ 1.588 & -0.809 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \sigma_b \\ \sigma_s \end{bmatrix} \quad (3.53)$$

The state-space model of (3.13) is controllable. Moreover, this linearized model is neither under-actuated nor over-actuated because it has two controlled variables (2-DOF) and two control effectors.

In model (3.13), the state-space dynamic matrix  $A$  has one stable second-order mode with natural frequency of 0.121 rad/s, a real stable mode at  $-6.5$  rad/s, and a zero mode. Regardless of the forward velocity value, the dynamic matrix always has a mode at zero.

**Remark 3.15.** *The model (3.53) is employed for the multi-input model simulation in the thesis.*

It is assumed that fault occurs in stern actuator. By employing the approach in Section 3.2, three different sets of control gain  $K$  in equation (3.6) are designed.

1. In this case, the control gain is selected such that the contribution of the states in total performance index is small.

$$K = \begin{bmatrix} 0.3320 & -0.5650 & 1.0000 & -3.7872 \\ 0.7908 & 0.6427 & 0.0061 & 1.1843 \end{bmatrix} \quad (3.54)$$

Here, the results for selected gain (3.54) are shown in Figures 3.44 - 3.46.

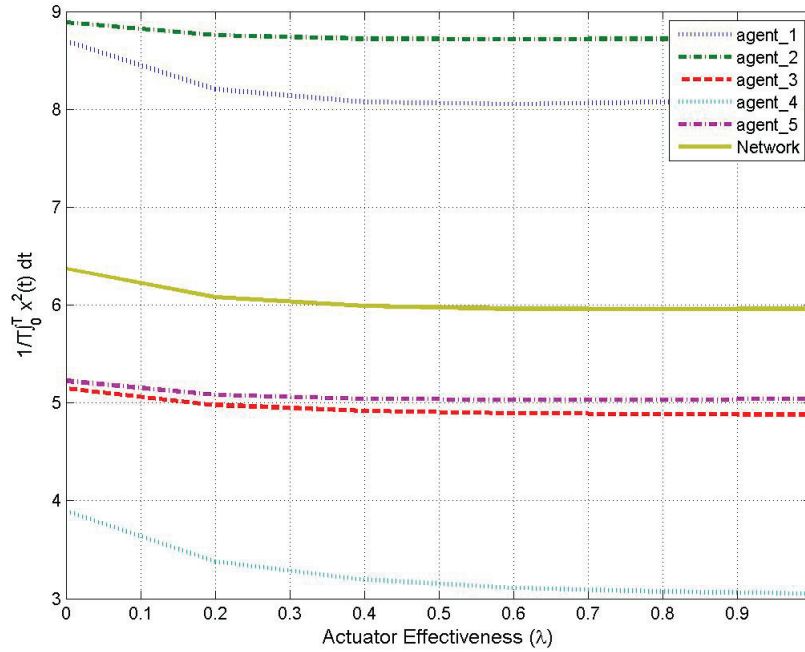


Figure 3.44: State performance index for static state feedback protocol without fault recovery - gain set 1.

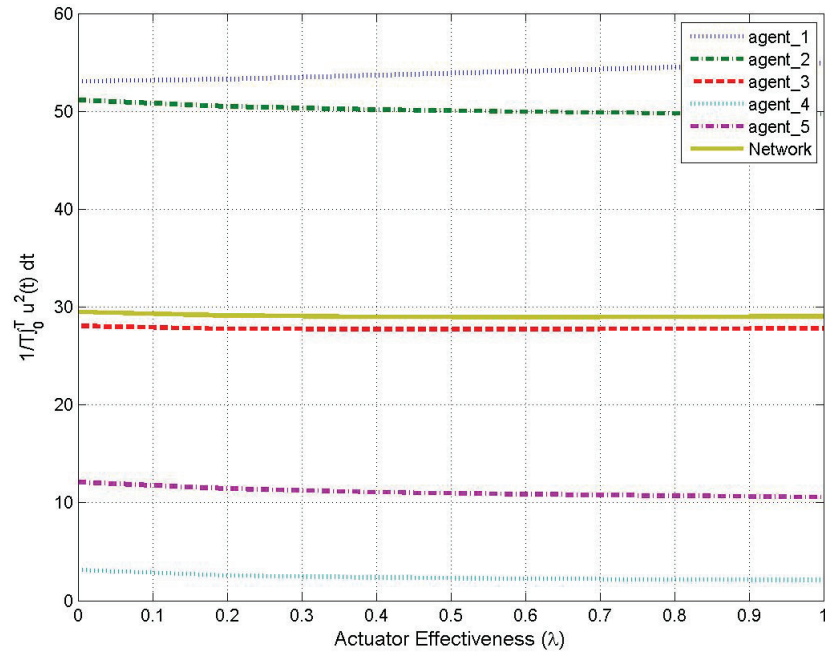


Figure 3.45: Control input performance index for static state feedback protocol without fault recovery - gain set 1.

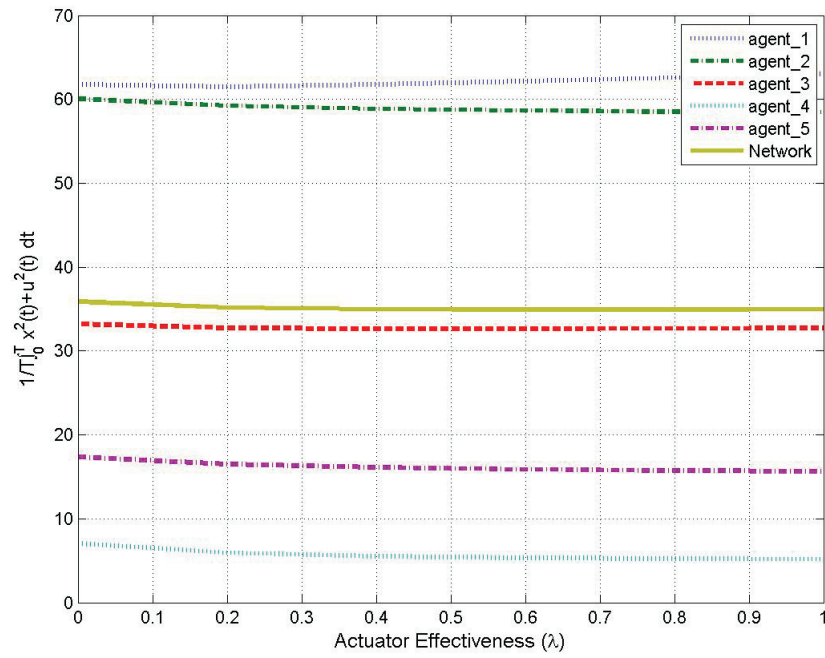


Figure 3.46: Total performance index for static state feedback protocol without fault recovery - gain set 1.

2. In this case, the control gain is selected such that compared to the previous case, the contribution of states in the total performance index is increased.

$$K = \begin{bmatrix} -0.1299 & -0.3506 & 0.2971 & -1.6673 \\ 0.4171 & 0.4044 & -0.1084 & 1.1199 \end{bmatrix} \quad (3.55)$$

Here, the results for the selected gain (3.55) are shown in Figures 3.47 - 3.49.

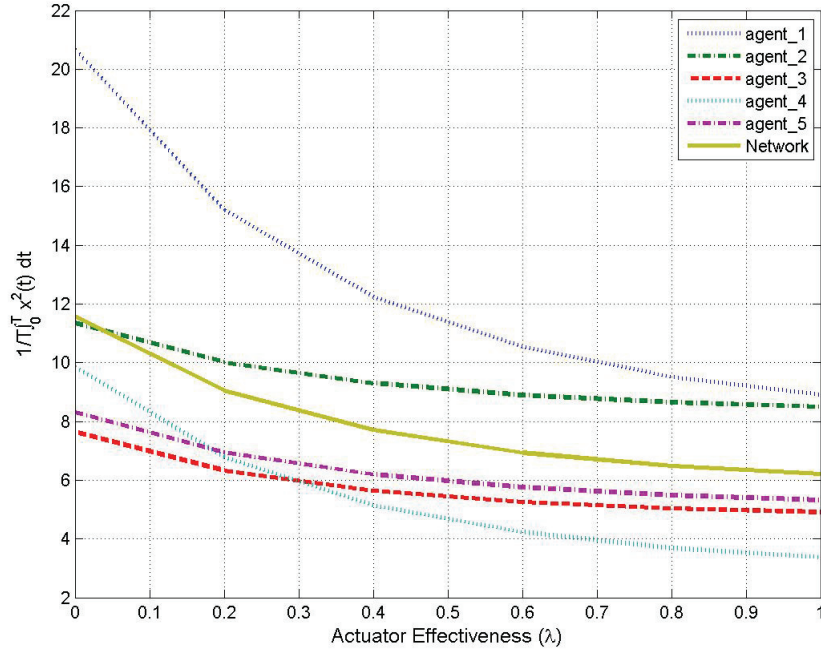


Figure 3.47: State performance index for static state feedback protocol without fault recovery - gain set 2.

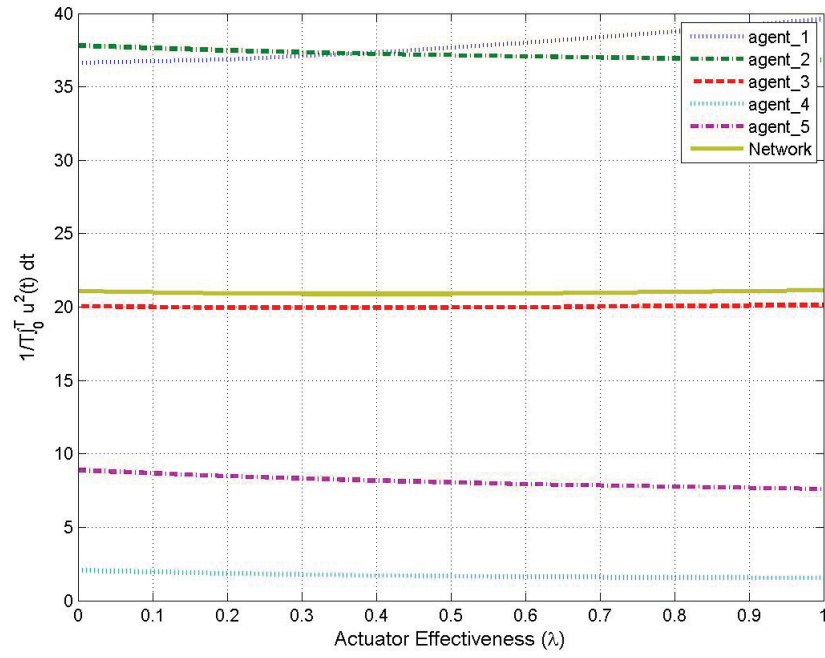


Figure 3.48: Control input performance index for static state feedback protocol without fault recovery - gain set 2.

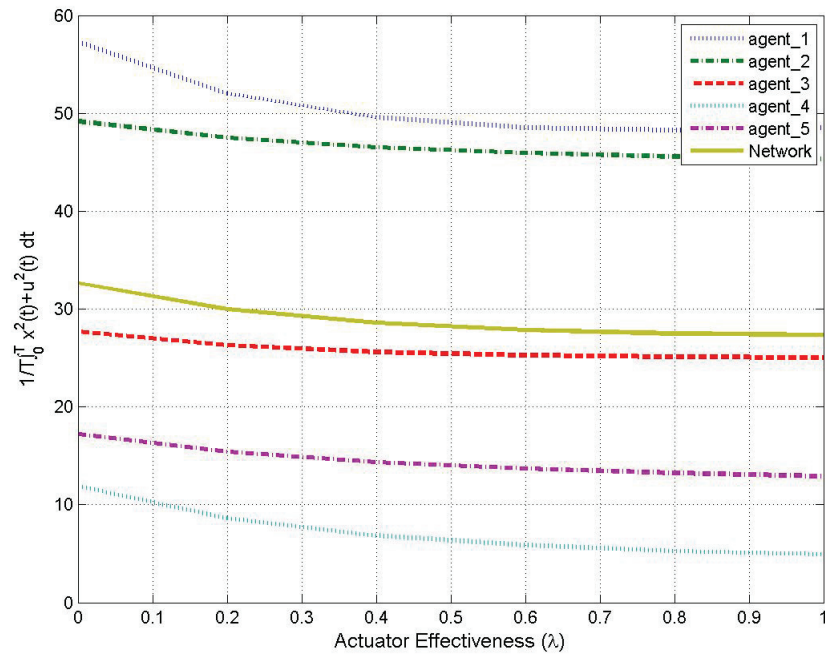


Figure 3.49: Total performance index for static state feedback protocol without fault recovery - gain set 2.

3. In this case, the control gain is selected such that the contribution of the states in the total performance index is the highest among these three cases.

$$K = \begin{bmatrix} -0.1741 & -0.1922 & 0.0770 & -0.6129 \\ 0.3189 & 0.2800 & -0.0638 & 0.6867 \end{bmatrix} \quad (3.56)$$

Here the results for the selected gain (3.56) are shown in Figures 3.50 - 3.52.

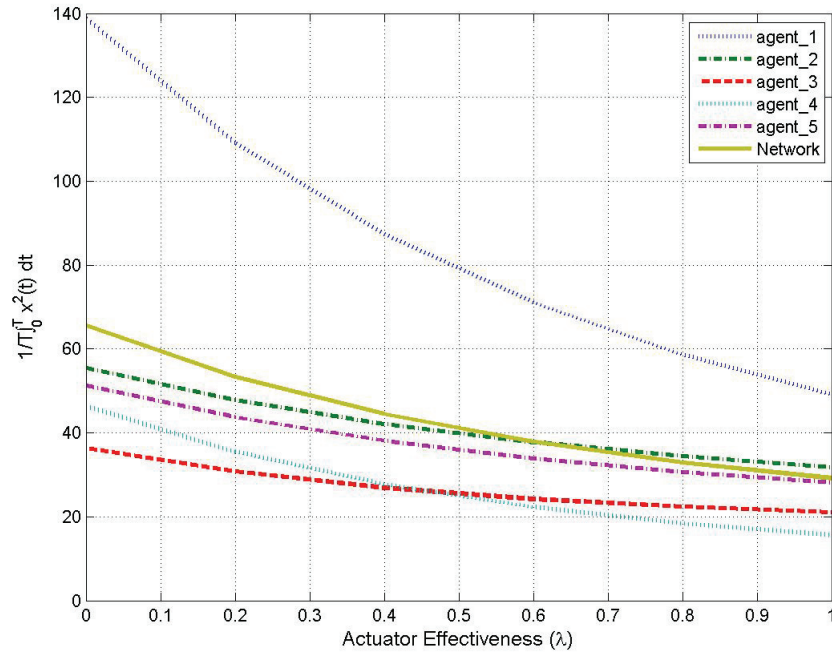


Figure 3.50: State performance index for static state feedback protocol without fault recovery - gain set 3.

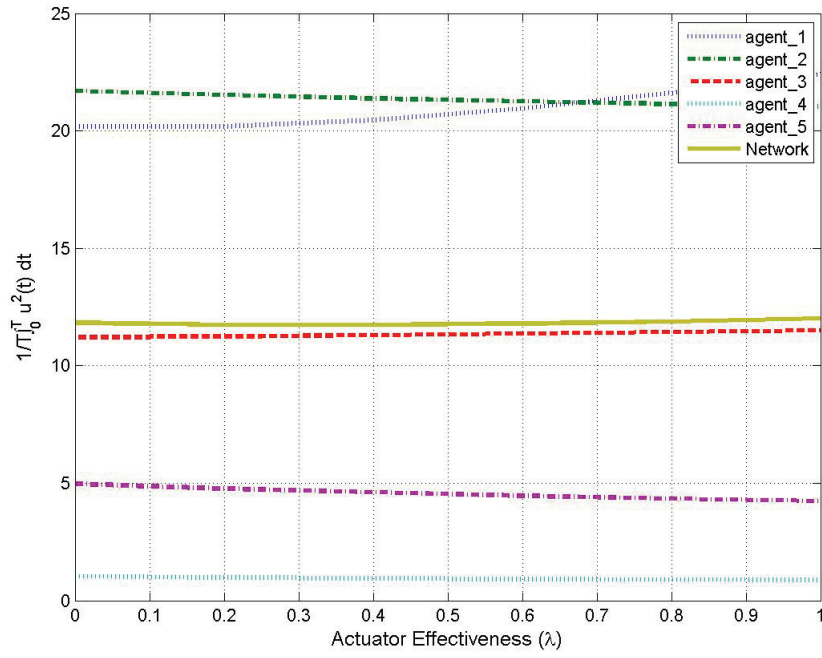


Figure 3.51: Control input performance index for static state feedback protocol without fault recovery - gain set 3.

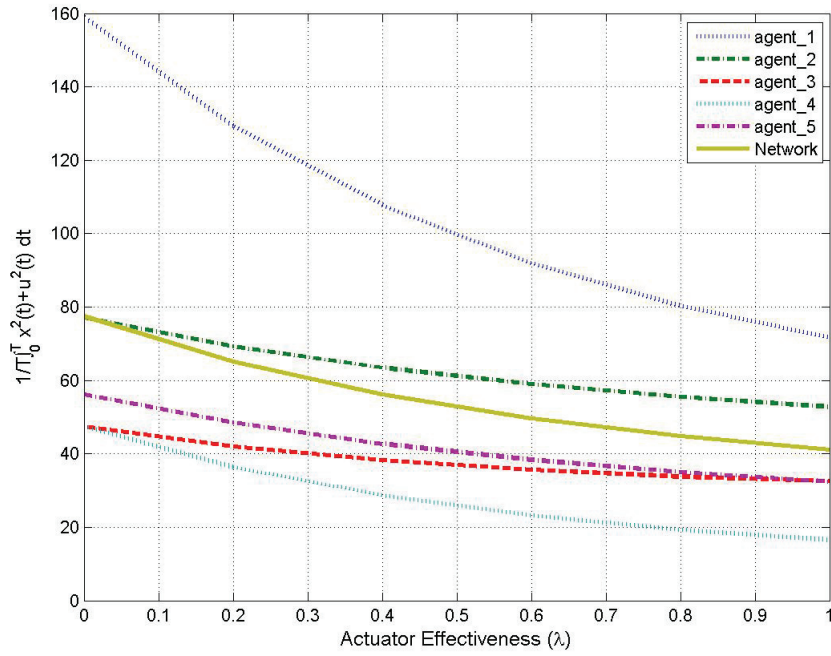


Figure 3.52: Total performance index for static state feedback protocol without fault recovery - gain set 3.

It is worth mentioning that the control input signal in Figures 3.44 - 3.52 is the control signal applied to the system and not the one calculated by the controller and therefore the effect of fault is compensated. This makes a difference in faulty case where these two signals are not identical.

Figures 3.44 - 3.52 demonstrate that as in the design of the control gain less weight is assigned to the states than the control input, the input performance index becomes more sensitive with respect to the fault.

**Remark 3.16.** *In the simulations conducted in this section for LOE and float faults, there is no change in control structure or control gain after fault occurrence. In other words, the control gain that is designed for the healthy system is used for the post fault, as well.*

The main conclusion of these simulations results is that the static approach is robust with respect to the fault. However, this is a limited approach and as in Section 3.3, the results will be degraded in the presence of noise.

## 3.9 Fault-Tolerant Static State Feedback Synchronization Protocol of Integrator Systems Subject to the LOE/Float Fault

In this section, integrator systems, as a special class of multi-agent systems with single input channel, are considered and the effects of LOE faults are investigated. Using the Gersgorin disk theorem [99], the analysis shows that loss of effectiveness fault in actuator(s) of single and double integrator systems does not affect consensus. Therefore, consensus achievement of integrator systems is fault-tolerant and there is no need for a fault detection and identification module to be utilized.

### 3.9.1 Single Integrator Systems

Consider a team of single integrators as

$$\dot{\mathbf{x}}_i(t) = \mathbf{u}_i(t) \quad i = 1, \dots, N$$

where  $\mathbf{x}_i \in \mathbb{R}$  and  $\mathbf{u}_i \in \mathbb{R}$ .

Now, assume that some agent(s) fail to produce the team control command. Specifically, due

to a loss of effectiveness fault in the actuator of agent 1, its input signal is changed to

$$\mathbf{u}_1^f(t) = - \sum_{j \in \mathcal{N}_1} \gamma_1 a_{1j} (\mathbf{x}_1(t) - \mathbf{x}_j(t)) \quad (3.57)$$

where  $0 < \gamma_1 < 1$  represents the actual effectiveness factor of the faulty actuator.

Therefore, the augmented state-space equation of the entire multi-agent system becomes

$$\dot{\mathbf{X}}(t) = -L' \mathbf{X}(t) \quad (3.58)$$

where  $\mathbf{X}(t) = [\mathbf{x}_1(t), \mathbf{x}_2(t), \dots, \mathbf{x}_N(t)]^T$ , and  $L'$  is the same as  $L$  except for the row associated with the faulty agent, which becomes

$$l'_{1j} = \gamma_1 l_{1j} \quad j = 1, \dots, N$$

**Lemma 3.3.** *The LOE fault in actuators of a team of single integrator systems, under the linear first order consensus protocol (3.57) and with a connected network topology, does not change the consensus achievement.*

*Proof.* The effect of a LOE fault in an agent's actuator is modeled by multiplying the associated row of the Laplacian matrix by a constant that represents the effectiveness of the faulty actuator.  $L' = [l'_{ij}]$  is still a zero row sum matrix, with  $l'_{ii} = \sum_{j \in \mathcal{N}_i} a'_{ij}$ , and  $l'_{ij} = -a'_{ij}$  for  $i \neq j$ . Therefore, the graph associated with  $L'$ , which is interpreted as the Laplacian matrix after fault occurrence, has a spanning tree. Using the results from Lemma 2.1,  $L'$  has a single zero eigenvalue and the rest of its eigenvalues have positive real parts. Hence, the agents dynamics will remain stable and consensus will still be reached.  $\square$

**Lemma 3.4.** *A float fault in one of the actuators of a team of single integrator systems, under the linear first order consensus protocol (3.57) and with a connected network topology, does not change the consensus achievement, if by removing the incoming links of the faulty agent, the graph associated with  $L'$  still has a spanning tree.*

*Proof.* See the proof of Lemma 3.3.  $\square$

### 3.9.2 Double Integrator System

Consider a team of second-order integrator multi-agent systems as

$$\begin{aligned}\dot{\mathbf{x}}_i(t) &= v_i(t) \\ \dot{v}_i(t) &= \mathbf{u}_i(t) \quad i = 1, \dots, N\end{aligned}$$

where  $\mathbf{x}_i \in \mathbb{R}$ ,  $v_i \in \mathbb{R}$  and  $\mathbf{u}_i \in \mathbb{R}$ .

Let us assume that agent 1 fails to produce the team control command and instead generates

$$\mathbf{u}_1^f(t) = - \sum_{j \in \mathcal{N}_1} \gamma_1 a_{1j} [\mathbf{x}_1(t) - \mathbf{x}_j(t) + \eta(v_1(t) - v_j(t))] \quad (3.59)$$

where  $0 < \gamma_1 < 1$  represents the actual effectiveness factor of the faulty actuator. Similar to the single integrator case, it is assumed that the fault occurs in one agent; however, it can easily be shown that the same analysis holds even if the fault happens in more than one agent.

Therefore, the augmented state-space equation of the entire multi-agent team is given by

$$\dot{\mathbf{X}}(t) = \begin{bmatrix} 0_{(N \times N)} & I_N \\ -L' & -\eta L' \end{bmatrix} \mathbf{X}(t) = \Gamma' \mathbf{X}(t) \quad (3.60)$$

where  $\mathbf{X}(t) = [\mathbf{x}_1^T(t), \mathbf{x}_2^T(t), \dots, \mathbf{x}_N^T(t)]^T$ , and  $L'$  is defined in (3.58).

**Lemma 3.5.** *The LOE fault in actuators of a team of double integrator systems with consensus protocol (3.59) and a connected network topology does not change the consensus achievement.*

*Proof.* From [110] one knows that the eigenvalues of  $\Gamma'$  are obtained as

$$\det \begin{bmatrix} \lambda I_N & -I_N \\ L' & \lambda I_N + \eta L' \end{bmatrix} = \det [\lambda^2 I_N + (1 + \eta \lambda) L'] \quad (3.61)$$

Knowing that  $\det(\lambda I_N + L') = \prod_{i=1}^N (\lambda - \mu_i)$  where  $\mu_i$  is the  $i$ th eigenvalue of  $-L'$ , the solutions of (3.61) will be

$$\pm \lambda_i = \frac{\eta \mu_i \pm \sqrt{\eta^2 \mu_i^2 + 4 \mu_i}}{2} \quad (3.62)$$

From Equation (3.62), it can be seen that for a  $L'$  with positive eigenvalues and a single zero eigenvalue, the  $\Gamma'$  will have two zero eigenvalues and the rest of its eigenvalues will be negative.

This concludes the proof.  $\square$

**Lemma 3.6.** *A float fault in one of the actuators of a team of double integrator systems, under the linear first order consensus protocol (3.59) and with a connected network topology, does not change the consensus achievement, if by removing the incoming links of the faulty agent, the graph associated with  $L'$  still has a spanning tree.*

*Proof.* The proof is similar to Lemma 3.5 and therefore is omitted.  $\square$

## 3.10 Reconfigurable State Feedback Synchronization Protocol Subject to the LIP Fault

In this section, it will be shown how applying a PI control protocol affects the synchronization in the presence of actuator LIP fault. Without a loss of generality, it is assumed that the fault occurs in agent 1. From (2.23) and (2.25), one knows that when there is a LIP fault in the actuator of agent 1, the effect of this fault in faulty actuator can be modeled as

$$\mathbf{u}_1^f(t) = \gamma_{1o}\beta_1^f \quad (3.63)$$

where  $0 \leq \gamma_{1o} \leq 1$ , and  $\beta_1^f$  is the actuator maximum capacity.

Therefore, the state-space representation of agent 1 will be

$$\begin{aligned} \dot{\mathbf{x}}_1(t) &= A\mathbf{x}_1(t) + B\mathbf{u}_1^f(t) \\ \mathbf{u}_1^f(t) &= \gamma_{1o}\beta_1^f \end{aligned} \quad (3.64)$$

**Lemma 3.7.** *If LIP fault occurs in the actuator of one of the agents in a group of multi-agent systems, synchronization could still be achieved by employing a PI protocol.*

*Proof.* Let us define new state variables as  $\sigma_i(t) = \mathbf{x}_1(t) - \mathbf{x}_i(t)$ , for  $i = 2, 3, \dots, N$ . This leads to the state-space representation as follows:

$$\begin{aligned} \dot{\sigma}_i(t) &= A\sigma_i(t) + B(\mathbf{u}_1^f(t) - \mathbf{u}_i(t)), \quad i = 2, 3, \dots, N \\ z_i(t) &= \sigma_i(t) \end{aligned} \quad (3.65)$$

As it is clear from (3.63) and (3.65), the effect of LIP fault shows itself as a constant disturbance in the new state-space equation of  $\sigma_i$ . Therefore, from internal model principle, in order to have steady state disturbance rejection the following two conditions should be satisfied.

1. Closed loop system must be stable.
2.  $C(s)G(s)$  must include the unstable modes of  $\mathbf{u}_1^f(s)$ , when  $G(s) = (sI_n - A)^{-1}B$  and  $C(s) = (K_p + \frac{K_I}{s})$  represent the transfer functions of agent's dynamics and controller dynamics, respectively.

The effect of LIP fault is modeled as a step disturbance on  $\sigma_i$  dynamics. Step disturbance has an unstable mode at  $s = 0$ . Therefore, to satisfy condition 2 the transfer function  $C(s)G(s)$  must have a pole at  $s = 0$ . This means, one needs to design a PI controller as

$$\mathbf{u}_i(t) = -K_p \sum_{j \in \mathcal{N}_i} a_{ij}(\mathbf{x}_i(t) - \mathbf{x}_j(t)) - K_I \int \sum_{j \in \mathcal{N}_i} a_{ij}(\mathbf{x}_i(t) - \mathbf{x}_j(t))dt \quad (3.66)$$

where  $K_p$  and  $K_I$  are both matrices of dimension  $1 \times n$ .

Now, it will be shown that by applying the control protocol of (3.66), the dynamics of  $\sigma_i$  will be stable and therefore consensus is achieved.

$$\begin{aligned} \dot{\sigma}_i(t) &= \dot{\mathbf{x}}_1(t) - \dot{\mathbf{x}}_i(t) = \\ &A\mathbf{x}_1(t) + B\gamma_{1o}\beta_1^f - (A\mathbf{x}_i(t) + B\mathbf{u}_i) = \\ &A(\mathbf{x}_1 - \mathbf{x}_i) - B\mathbf{u}_i + B\gamma_{1o}\beta_1^f = \\ &A\sigma_i(t) - B[-K_p \sum_{j \in \mathcal{N}_i} a_{ij}(\mathbf{x}_i(t) - \mathbf{x}_j(t)) - K_I \int \sum_{j \in \mathcal{N}_i} a_{ij}(\mathbf{x}_i(t) - \mathbf{x}_j(t))dt] + B\gamma_{1o}\beta_1^f = \\ &A\sigma_i(t) - BK_p \sum_{j \in \mathcal{N}_i} a_{ij}(\sigma_i(t) - \sigma_j(t)) - BK_I \int \sum_{j \in \mathcal{N}_i} a_{ij}(\sigma_i(t) - \sigma_j(t))dt + B\gamma_{1o}\beta_1^f \\ &z_i(t) = \sigma_i(t) \quad i = 2, 3, \dots, N \end{aligned} \quad (3.67)$$

Then, the augmented dynamic equation of the entire agents will be

$$\begin{aligned} \dot{\sigma}(t) &= (I_{N-1} \otimes A - L_r \otimes BK_p)\sigma(t) - (L_r \otimes BK_I) \int \sigma(t)dt + (I_{N-1} \otimes B)w(t) \\ z(t) &= \sigma(t) \end{aligned} \quad (3.68)$$

where  $w(t) = 1_{N-1} \otimes \gamma_{1o}\beta_1^f s(t)$  ( $s(t)$  is the step function), and the rest of the variables are defined in (3.7).

It follows that LIP fault in agent 1 can be modeled by setting  $\alpha = 0$ , and adding a constant disturbance to the augmented state-space model. From (3.68), the synchronization of multi-agents with single input channel under LIP fault is formulated as a simultaneous stabilization problem subject to a constant disturbance. For the sake of simplicity, it has been assumed that fault occurs at time zero.

For any network topology, one knows that there exist an orthogonal matrix  $T$  such that  $J = T^T L_r T$  is a triangular matrix, whose principal diagonal elements consist of  $\lambda_i$ ,  $i = 2, 3, \dots, N$  [34]. Knowing this and by applying the change of variables as  $\varepsilon = (T^T \otimes I_n)\sigma$ , one may have

$$\begin{aligned} \dot{\varepsilon}(t) &= (I_{N-1} \otimes A - J \otimes BK_P)\varepsilon(t) - (J \otimes BK_I) \int \varepsilon(t)dt + w'(t) \\ z'(t) &= (T^T \otimes I_n)\sigma(t) = \varepsilon(t) \end{aligned} \quad (3.69)$$

where  $w'(t) = (T^T \otimes I_n)(I_{N-1} \otimes B)w(t)$ .

**Remark 3.17.** Knowing that matrix  $(T^T \otimes I_n)$  is non-singular the zero solution of  $\varepsilon(t)$  will lead to  $\sigma(t) = 0$ .

In order to check the second condition of the internal model principle, we will transform the closed loop equations into s-domain, which is more convenient. Thus, by taking the Laplace transform of equation (3.69), we will have

$$\begin{aligned} s\varepsilon(s) &= (I_{N-1} \otimes A - J \otimes BK_P)\varepsilon(s) - \frac{1}{s}(J \otimes BK_I)\varepsilon(s) + w'(s) \\ z'(s) &= (T^T \otimes I_n)\sigma(s) = \varepsilon(s) \end{aligned} \quad (3.70)$$

where by defining  $w'(t) = (T^T \otimes I_n)(I_{N-1} \otimes B)(1_{N-1} \otimes \gamma_{1o}\beta_1^f)s(t)$ , and taking the Laplace transform, we will have  $w'(s) = \frac{1}{s}(T^T \otimes I_n)(I_{N-1} \otimes B)(1_{N-1} \otimes \gamma_{1o}\beta_1^f) = \frac{1}{s}W'$ . It can be seen that  $(T^T \otimes I_n)(I_{N-1} \otimes B)(1_{N-1} \otimes \gamma_{1o}\beta_1^f)$  is defined as  $W'$ , which is a constant vector of size  $n \times (N-1)$ .

Therefore, the output vector  $z'(s)$  will be

$$z'(s) = [(s^2 I_{(N-1) \times n} - s(I_{N-1} \otimes A - J \otimes BK_P) + (J \otimes BK_I))]^{-1} W' \quad (3.71)$$

By using matrix inverse properties for triangular matrices in (2.17) and (2.18) and knowing that the eigenvalues of  $L_r$  appear as diagonal elements of  $J$ , one may write (3.71) as follows:

$$\begin{aligned}
 z'(s) &= \begin{bmatrix} (s^2 I_n - s(A - \lambda_2 B K_P) + \lambda_2 B K_I) & & \times \\ & \ddots & \\ 0 & & (s^2 I_n - s(A - \lambda_N B K_P) + \lambda_N B K_I) \end{bmatrix}^{-1} W' \\
 &= \begin{bmatrix} (s^2 I_n - s(A - \lambda_2 B K_P) + \lambda_2 B K_I)^{-1} & & \times \\ & \ddots & \\ 0 & & (s^2 I_n - s(A - \lambda_N B K_P) + \lambda_N B K_I)^{-1} \end{bmatrix} W'
 \end{aligned} \tag{3.72}$$

In the above equation, the notation  $\times$  means that we do not care about the upper diagonal elements.

**Remark 3.18.** For a diagonalizable Laplacian matrix (e. g. networks with two-way communication links), (3.72) will reduce to a diagonal matrix.

The system of (3.72) is comprised of  $N - 1$  subsystems. In order to simplify the stability analysis of  $z'(s)$ , only its last subsystem is taken into account and analyzed. This can simply be extended to the rest of the subsystems.

In this regard, the vector  $z'(s)$  has been split as  $z' = [z'_1, z'_2, \dots, z'_{N-1}]$ . Therefore, the last subsystem of (3.72) will be

$$\begin{aligned}
 z'_{N-1}(s) &= (s^2 I_n - s(A - \lambda_N B K_P) + \lambda_N B K_I)^{-1} W'_{N-1} = \\
 &= (s(s I_n - (A - \lambda_N B K_P)) + \lambda_N B K_I)^{-1} W'_{N-1} = \\
 &= s^{-1}((s I_n - (A - \lambda_N B K_P)) + \frac{\lambda_N B K_I}{s})^{-1} W'_{N-1} = \\
 &= s^{-1}((s I_n - A) + \lambda_N B(K_P + \frac{K_I}{s}))^{-1} W'_{N-1} = \\
 &= s^{-1}((s I_n - A) + \lambda_N(s I_n - A)(s I_n - A)^{-1} B(K_P + \frac{K_I}{s}))^{-1} W'_{N-1} = \\
 &= s^{-1}((s I_n - A) + \lambda_N(s I_n - A)G(s)C(s))^{-1} W'_{N-1} = \\
 &= s^{-1}(I_n + \lambda_N G(s)C(s))^{-1}(s I_n - A)^{-1} W'_{N-1}
 \end{aligned} \tag{3.73}$$

Using,

$$W' = (T^T \otimes I_n)(I_{N-1} \otimes B)(1_{N-1} \otimes \gamma_{1o}\beta_1^f) = \gamma_{1o}\beta_1^f \begin{bmatrix} t_{11}B & \cdots & t_{1(N-1)}B \\ \vdots & \ddots & \vdots \\ t_{(N-1)1}B & \cdots & t_{(N-1)(N-1)}B \end{bmatrix} \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix} \quad (3.74)$$

$W'_{N-1}$  could be written as

$$W'_{N-1} = \gamma_{1o}\beta_1^f t'_{N-1} B \quad (3.75)$$

where  $t'_{N-1} = t_{(N-1)1} + \cdots + t_{(N-1)(N-1)}$  and  $t_{ij}$ s are the elements of  $T^T$ .

Finally,  $z'_{N-1}(s)$  could be written as follows:

$$\begin{aligned} z'_{N-1}(s) &= (\gamma_{1o}\beta_1^f t'_{N-1})s^{-1}(I_n + \lambda_N G(s)C(s))^{-1}(sI_n - A)^{-1}B = \\ &= (\gamma_{1o}\beta_1^f t'_{N-1})s^{-1}(I_n + \lambda_N G(s)C(s))^{-1}G(s) \end{aligned} \quad (3.76)$$

Here, it will be shown that for a marginally stable system regardless of the location of poles on imaginary axis (a single pole at  $s = 0$ , or complex conjugated poles on the imaginary axis), the states described by (3.76) will converge to zero. In this regard, from the final value theorem, one knows that  $\lim_{t \rightarrow \infty} z'_{N-1}(t) = \lim_{s \rightarrow 0} s z'_{N-1}(s)$  if

1. All roots in the denominator of  $z'_{N-1}(s)$  have negative real parts.
2.  $z'_{N-1}(s)$  does not have more than one pole at the origin.

As in condition 1 of final value theorem,  $K_p$  and  $K_I$  must be designed in a way that  $z'_{N-1}(s)$  has no eigenvalues with positive real parts. Since many control design approaches have been developed in time domain, the closed loop equations of the system will be transformed back to time domain. Thus, time-domain equivalent of (3.76) is obtained as follows:

$$\begin{aligned} \dot{z}'_{N-1}(t) &= A z'_{N-1}(t) + \lambda_N B u_{N-1}(t) + B w_{N-1}(t) \\ w_{N-1}(t) &= (\gamma_{1o}\beta_1^f t'_{N-1}) \end{aligned} \quad (3.77)$$

By applying Laplace transform to equation (3.77), we will have

$$\begin{aligned}
sz'_{N-1}(s) &= Az'_{N-1}(s) + \lambda_N BC(s)z'_{N-1}(s) + B(\gamma_{1o}\beta_1^f t'_{N-1})s^{-1} \implies \\
(sI_n - A)z'_{N-1}(s) &= \lambda_N BC(s)z'_{N-1}(s) + (\gamma_{1o}\beta_1^f t'_{N-1})s^{-1}B \implies \\
z'_{N-1}(s) &= \lambda_N (sI_n - A)^{-1} BC(s)z'_{N-1}(s) + (sI_n - A)^{-1} Bw(s) = \\
&\lambda_N G(s)C(s)z'_{N-1}(s) + G(s)(\gamma_{1o}\beta_1^f t'_{N-1})s^{-1} \implies \\
z'_{N-1}(s) &= (\gamma_{1o}\beta_1^f t'_{N-1})s^{-1}(I_n - \lambda_N G(s)C(s))^{-1}G(s) \tag{3.78}
\end{aligned}$$

which is the same as (3.76). Therefore, one can use any conventional or modern control approach either in time domain or Laplace domain to design  $K_P$  and  $K_I$ .

Now, given that  $K_P$  and  $K_I$  are selected such that condition 1 of the final value theorem is satisfied, the second condition will be investigated. Knowing that  $C(s)$  is a PI controller and regardless of whether  $G(s)$  has a pole at  $s = 0$  or not, the condition 2 is also met, and we will have

$$\lim_{t \rightarrow \infty} z'_{N-1}(t) = \lim_{s \rightarrow 0} s(\gamma_{1o}\beta_1^f t'_{N-1})s^{-1}(I_n + \lambda_N G(s)C(s))^{-1}G(s) = 0 \tag{3.79}$$

In order to explain more how the equation (3.79) is satisfied, in the following, two possible cases for the location of poles in a  $n$ th order system (system with single input and  $n$  output) are considered. In case 1, it is assumed that only the first input-output (state) transfer function has a pole at  $s = 0$ . In a similar way, it can be extended to the case where more than one of the input-output transfer functions have a pole at  $s = 0$ .

- $G(s)$  has one pole at  $s = 0$ . For the sake of simplicity, the  $n$ th order transfer function  $G(s)$  and  $C(s)$  have been split as  $G(s) = [g_{11}s^{-1}, g_{21}, \dots, g_{n1}]$  and  $C(s) = [c_{11}s^{-1}, c_{21}s^{-1}, \dots, c_{n1}s^{-1}]$ , respectively. It is to be noted that  $g_{ts}$  and  $c_{ts}$  for  $t, s = 1, \dots, n$  are functions of  $s$  (with non-zero values at  $s = 0$ ); however, for the sake of simplicity it has not been shown. Knowing

these, (3.79) could be written as

$$\begin{aligned}
& \lim_{t \rightarrow \infty} z'_{N-1}(t) = \\
& (\gamma_{1o} \beta_1^f t'_{N-1}) \lim_{s \rightarrow 0} \left[ \begin{array}{cccc} 1 + g_{11}c_{11}s^{-2} & g_{11}c_{12}s^{-2} & \cdots & g_{11}c_{1n}s^{-2} \\ g_{21}c_{11}s^{-1} & 1 + g_{21}c_{12}s^{-1} & \cdots & g_{21}c_{1n}s^{-1} \\ \vdots & \vdots & \ddots & \vdots \\ g_{n1}c_{11}s^{-1} & g_{n1}c_{12}s^{-1} & \cdots & 1 + g_{n1}c_{1n}s^{-1} \end{array} \right]^{-1} \left[ \begin{array}{c} g_{11}s^{-1} \\ g_{21} \\ \vdots \\ g_{n1} \end{array} \right] \cong \\
& \lim_{s \rightarrow 0} s^4 \left[ \begin{array}{cccc} 1 + gc_{11}s^{-2} & gc_{12}s^{-3} & \cdots & gc_{1n}s^{-3} \\ gc_{21}s^{-2} & 1 + gc_{22}s^{-3} & \cdots & gc_{2n}s^{-3} \\ \vdots & \vdots & \ddots & \vdots \\ gc_{n1}s^{-2} & gc_{n2}s^{-3} & \cdots & 1 + gc_{nn}s^{-3} \end{array} \right] \left[ \begin{array}{c} s^{-1} \\ 1 \\ \vdots \\ 1 \end{array} \right] \\
& \lim_{s \rightarrow 0} \left[ \begin{array}{c} s \\ s \\ \vdots \\ s \end{array} \right] = \left[ \begin{array}{c} 0 \\ 0 \\ \vdots \\ 0 \end{array} \right] \tag{3.80}
\end{aligned}$$

Note that  $g_{tk}c_{ks}$  is defined as a new function as  $gc_{ts}$  for  $t, s = 1, \dots, n$ .

- $G(s)$  does not have any pole at  $s = 0$ . In this case,  $G(s) = [g_{11}, g_{21}, \dots, g_{n1}]$  and  $C(s) = [c_{11}s^{-1}, c_{21}s^{-1}, \dots, c_{n1}s^{-1}]$ . Therefore, we will have

$$\begin{aligned}
& \lim_{t \rightarrow \infty} z''_{N-1}(t) = \\
& (\gamma_{1o} \beta_1^f t'_{N-1}) \lim_{s \rightarrow 0} \left[ \begin{array}{cccc} 1 + g_{11}c_{11}s^{-1} & g_{11}c_{12}s^{-1} & \cdots & g_{11}c_{1n}s^{-1} \\ g_{21}c_{11}s^{-1} & 1 + g_{21}c_{12}s^{-1} & \cdots & g_{21}c_{1n}s^{-1} \\ \vdots & \vdots & \ddots & \vdots \\ g_{n1}c_{11}s^{-1} & g_{n1}c_{12}s^{-1} & \cdots & 1 + g_{n1}c_{1n}s^{-1} \end{array} \right]^{-1} \left[ \begin{array}{c} g_{11} \\ g_{21} \\ \vdots \\ g_{n1} \end{array} \right] \cong
\end{aligned}$$

$$\begin{aligned}
& \lim_{s \rightarrow 0} s^3 \begin{bmatrix} 1 + gc_{11}s^{-2} & gc_{12}s^{-2} & \cdots & gc_{1n}s^{-2} \\ gc_{21}s^{-2} & 1 + gc_{22}s^{-2} & \cdots & gc_{2n}s^{-2} \\ \vdots & \vdots & \ddots & \vdots \\ gc_{n1}s^{-2} & gc_{n2}s^{-2} & \cdots & 1 + gc_{nn}s^{-2} \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} \\
& \lim_{s \rightarrow 0} \begin{bmatrix} s \\ s \\ \vdots \\ s \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \tag{3.81}
\end{aligned}$$

This means that condition 2 of the final value theorem is satisfied. This along with condition 1 shows that closed loop system is stable.  $\square$

When consensus is reached, we will have

$$\begin{aligned}
\lim_{t \rightarrow \infty} \mathbf{x}_i(t) &= \lim_{t \rightarrow \infty} \mathbf{x}_j(t) = \lim_{t \rightarrow \infty} \mathbf{x}_1(t) = \\
\lim_{s \rightarrow 0} s \mathbf{x}_1(s) &= \lim_{s \rightarrow 0} s(sI_n - A)^{-1} \mathbf{x}_1(t_f) + (\gamma_{1o} \beta_1^f) s(sI_n - A)^{-1} B s^{-1} = \\
\lim_{s \rightarrow 0} s(sI_n - A)^{-1} \mathbf{x}_1(t_f) &+ (\gamma_{1o} \beta_1^f) G(s)
\end{aligned} \tag{3.82}$$

where  $t_f$  represents the fault occurrence time.

The final consensus value in (3.82) can be either of the followings:

1.  $G(s)$  has a pole at  $s = 0$ : In this case, regardless of the initial value, the final value of states goes to  $\infty$ . Therefore, for AUV system, which is marginally stable, the final values of states goes to  $\infty$ .
2.  $G(s)$  has no pole at origin: In this case, regardless of the initial value, the final value of states goes to  $G(0)(\gamma_{1o} \beta_1^f)$ .

**Remark 3.19.** *Given that there is a spanning tree rooted from the faulty agent, neither the states' value at fault occurrence time ( $\mathbf{x}_i(t_f)$ ), nor network structure affect the final consensus value.*

**Remark 3.20.** *In the case of finite consensus value, this value is proportional to the severity of LIP fault.*

**Remark 3.21.** *In the case of LIP fault, the controller is restructured from a simple proportional controller to a proportional-integral one.*

### 3.11 Reconfigurable State Feedback Dynamic Synchronization Protocol Subject to the LOE/Float Fault

In this section a team of  $N$  multi-agent systems with  $m$  input channels are considered as

$$\dot{\mathbf{x}}_i(t) = A\mathbf{x}_i(t) + B\mathbf{u}_i(t) \quad i = 1, 2, \dots, N \quad (3.83)$$

where  $A \in \mathbb{R}^{n \times n}$  is a marginally stable matrix,  $B \in \mathbb{R}^{n \times m}$ ,  $\mathbf{x}_i \in \mathbb{R}^n$ ,  $\mathbf{u}_i \in \mathbb{R}^{m \times 1}$ , and the pair  $(A, B)$  is stabilizable.

Let us assume that there is a fault in the first actuator of agent 1. Therefore, after fault occurrence, the control input matrix becomes

$$\begin{aligned} B_1 &= B\Gamma_1, \quad \Gamma_1 = \text{diag}(\gamma_1^1, 1, \dots, 1) \\ B_2 &= B_3 = \dots = B_N = B \end{aligned} \quad (3.84)$$

where  $0 \leq \gamma_1^1 < 1$  indicates the effectiveness factor of the faulty actuator of agent 1.

By applying the static consensus protocol (3.6) and equations (3.84) to (3.83), due to the fact that the agents are not identical any longer ( $B_1 = B\Gamma_1$ ), the augmented state-space representation of (3.7) is no longer valid. Particularly, the Kronecker product notation cannot be employed any longer. Therefore, the static protocol methodology proposed in Section 3.6 cannot guarantee the post-fault recovery.

Given that the accurate fault information are available through a fault detection and identification module, a dynamic protocol is proposed. However, this methodology can be used as a passive approach, without control gain redesign, if post-fault results meet the stability and performance criteria.

**Lemma 3.8.** *The dynamic protocol*

$$\begin{aligned} \dot{\zeta}_i(t) &= (A + B_i'K_i')\zeta_i(t) + \sum_{j \in \mathcal{N}_i} (\zeta_j(t) - \zeta_i(t) + \mathbf{x}_i(t) - \mathbf{x}_j(t)) \\ \mathbf{u}_i(t) &= K_i'\zeta_i(t) \quad i = 1, \dots, N \end{aligned} \quad (3.85)$$

will synchronize the system (3.83) and (3.84) when the accurate fault information is available.

where  $\zeta_i \in \mathbb{R}^n$ ,  $B_i' \in \mathbb{R}^{n \times m}$  and  $K_i' \in \mathbb{R}^{m \times n}$  are respectively the controller state, estimated control input matrix, and reselected control gain associated with the agent  $i$ . Note that assuming the

exact fault estimation information are available, we will have  $B'_1 = B_1 = B\Gamma_1$ ,  $B'_i = B_i = B$  for  $i = 2, \dots, N$  and  $K'_i = K_i = K$  for  $i = 2, \dots, N$ .

*Proof.* In order to prove that the control protocol of (3.85) leads to consensus, the synchronization state variables are defined as  $s_i(t) = \mathbf{x}_i(t) - \zeta_i(t)$ . In other words, synchronization state is defined as difference between the controller state and the plant state. At steady state, when the controller state goes to zero ( $\zeta_i(t) \rightarrow 0 \forall i$ ) if  $s_i = s_j$  for  $\forall i, j$ , then synchronization will be achieved ( $\mathbf{x}_i = \mathbf{x}_j$  for  $\forall i, j$ ).

Thus,  $s_i$  dynamics will be as

$$\begin{aligned} \dot{s}_i(t) &= \dot{\mathbf{x}}_i(t) - \dot{\zeta}_i(t) = \\ &= A\mathbf{x}_i(t) + B'_i K'_i \zeta_i(t) - ((A + B'_i K'_i)\zeta_i(t) + \sum_{j \in \mathcal{N}_i} (\zeta_j(t) - \zeta_i(t) + \mathbf{x}_i(t) - \mathbf{x}_j(t))) = \\ &= As_i(t) - \sum_{j \in \mathcal{N}_i} (s_i(t) - s_j(t)) \quad i = 1, 2, \dots, N \end{aligned} \quad (3.86)$$

Therefore, the augmented state-space system (including the control state and the synchronization state variables) will become

$$\dot{\zeta}(t) = (\tilde{A}_N + B'_d K'_d)\zeta(t) + Ls(t) \quad (3.87a)$$

$$\dot{s}(t) = \tilde{A}_N s(t) - Ls(t) \quad (3.87b)$$

where  $L = [l_{ij}]$  is the Laplacian matrix of the network with  $l_{ii} = \sum_{j \in \mathcal{N}_i} a_{ij}$  and  $l_{ij} = -a_{ij}$  for  $i \neq j$ ,  $\tilde{A}_N = I_N \otimes A$ ,  $B'_d = \text{block diag}(B'_1, B, \dots, B)$ ,  $K'_d = \text{block diag}(K'_1, K, \dots, K)$ ,  $\zeta = [\zeta_1^T, \zeta_2^T, \dots, \zeta_N^T]^T$ ,  $s = [s_1^T, s_2^T, \dots, s_N^T]^T$ , and  $K'_1$  is the control gain matrix of the faulty agent 1.

From equation (3.87b), it can be seen that the consensus dynamics does not depend on the dynamics of the controller. Therefore, using the result from Lemma 2.2, it is guaranteed that  $s_i(t)$  in (3.87b) will reach consensus asymptotically and the consensus trajectory will be governed by  $\dot{s}_c(t) = As_c(t)$ . This is obtained by using the fact that the Laplacian is a zero row sum matrix.

The assumption that  $A$  is a marginally stable matrix does not violate the generality of the

lemma. This is a common assumption in synchronization problems. However, the eigenvalues with negative real parts can be ignored as their effect will diminish in steady state.

Hence, when the consensus trajectory of (3.87b) reaches its steady state (and therefore  $Ls(t)$  reaches zero), the augmented dynamics of the controller in equation (3.87a) will reduce to

$$\dot{\zeta}(t) = (\tilde{A}_N + B'_d K'_d) \zeta(t) \quad (3.88)$$

This implies if the control gain  $K'_d$  is redesigned to make  $\tilde{A}_N + B'_d K'_d$  Hurwitz, the dynamics of (3.87a) will be exponentially stable and will reach zero asymptotically. The control gain  $K'_d$  can be designed using any conventional control approach.

Finally,  $\mathbf{x}_i(t)$  is obtained by  $\mathbf{x}_i(t) = s_i(t) + \zeta_i(t)$  and will converge to  $s_i(t)$ . Therefore, one can conclude that all agents will reach to a common solution of  $\dot{s}_c(t) = A s_c(t)$ .  $\square$

**Remark 3.22.** *In this section, the communication graph is assumed to be connected. This means it is uniformly connected as well. The time dependence of  $a_{ij}(t)$  in Lemma 2.2 shows that this lemma is valid even though the graph changes with time.*

**Remark 3.23.** *The proposed dynamic protocol (3.85) is adopted from [36]. In that paper, the authors consider a group of homogeneous agents. However, the problem in this section models the post-fault system as a group of heterogeneous agents which try to synchronize.*

**Remark 3.24.** *It is worth mentioning that as in equation (3.87b), the fault does not change the consensus dynamics.*

**Remark 3.25.** *It is worth noting that after fault occurrence only the control gain of the faulty agent 1 ( $K'_1$ ) changes, and the control gains for the rest of the agents remain the same as the fault-free case.*

### 3.11.1 Simulation Results

In this section, a multi-input model of AUV (with stern and bow as inputs) as in equation (3.53) and by having access to full state-space information is considered.

It is assumed that LOE/float fault occurs in stern actuator at time 2 Sec. By employing the approach in Section 3.11 in a passive manner, without control gain redesign, three different sets of control gain  $K$  of equation (3.85) are applied.

1. In this case, the control gain is selected such that the contribution of the states in total performance index is small.

$$K = \begin{bmatrix} 0.3320 & -0.5650 & 1.0000 & -3.7872 \\ 0.7908 & 0.6427 & 0.0061 & 1.1843 \end{bmatrix} \quad (3.89)$$

The simulation results for the selected gain (3.89) are shown in Figures 3.53 - 3.55.

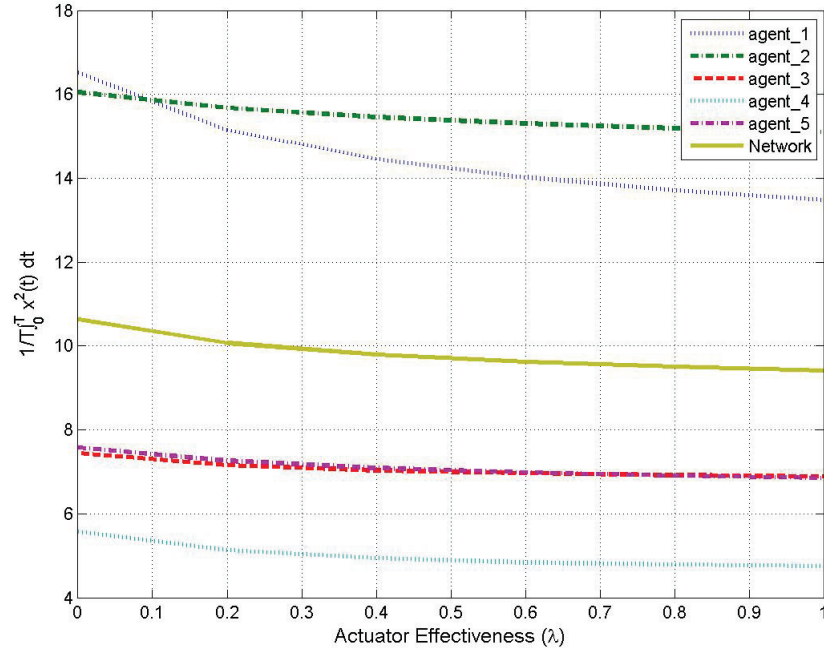


Figure 3.53: State performance index for dynamic state feedback protocol without fault recovery - gain set 1.

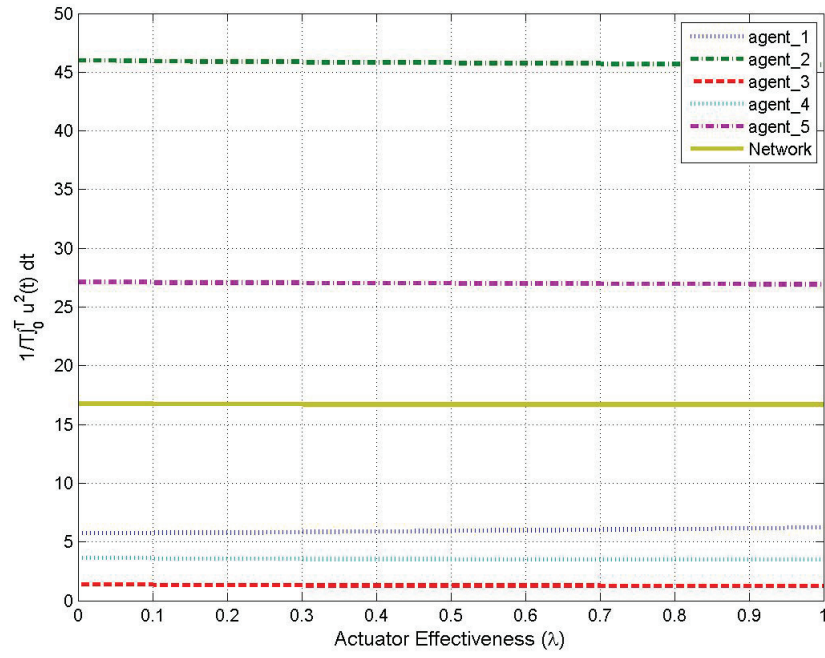


Figure 3.54: Control input performance index for dynamic state feedback protocol without fault recovery - gain set 1.

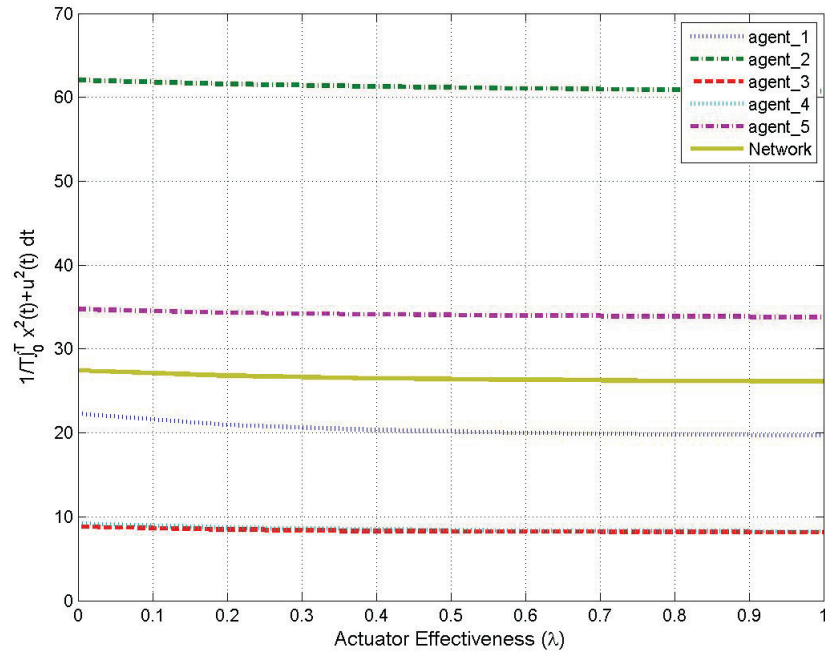


Figure 3.55: Total performance index for dynamic state feedback protocol without fault recovery - gain set 1.

2. In this case, the control gain is selected such that compared to the previous case, the contribution of states in the total performance index is increased.

$$K = \begin{bmatrix} -0.1299 & -0.3506 & 0.2971 & -1.6673 \\ 0.4171 & 0.4044 & -0.1084 & 1.1199 \end{bmatrix} \quad (3.90)$$

The simulation results for the selected gain (3.90) are shown in Figures 3.56 - 3.58.

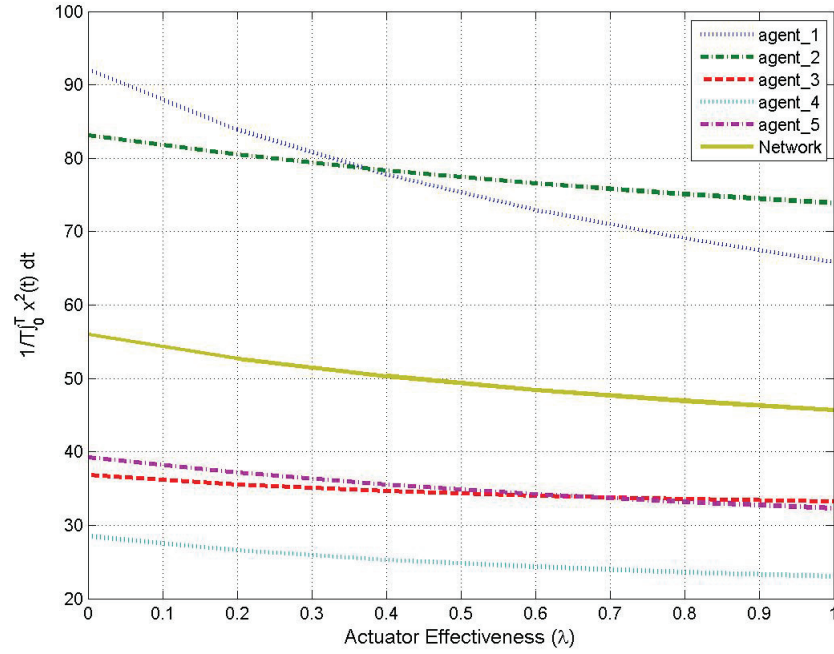


Figure 3.56: State performance index for dynamic state feedback protocol without fault recovery - gain set 2.

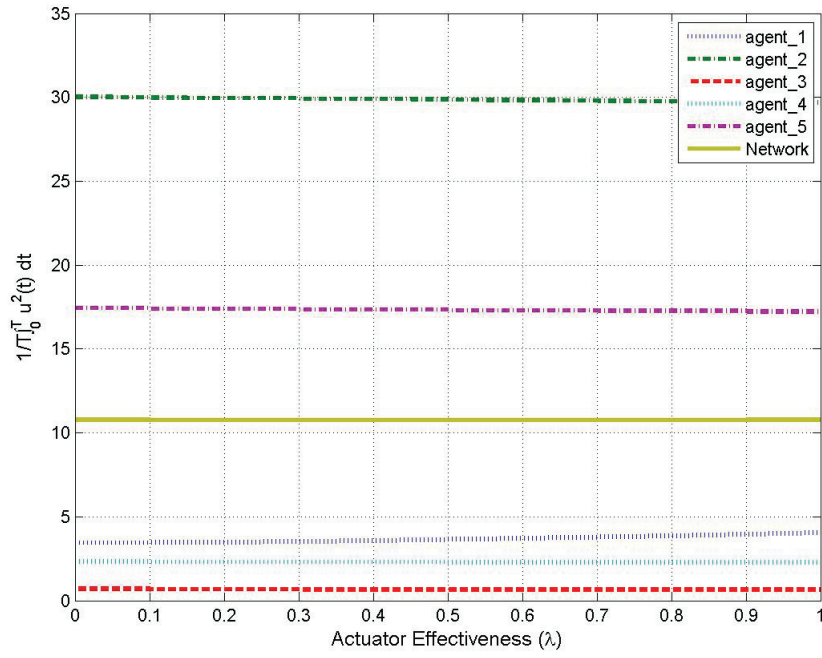


Figure 3.57: Control input performance index for dynamic state feedback protocol without fault recovery - gain set 2.

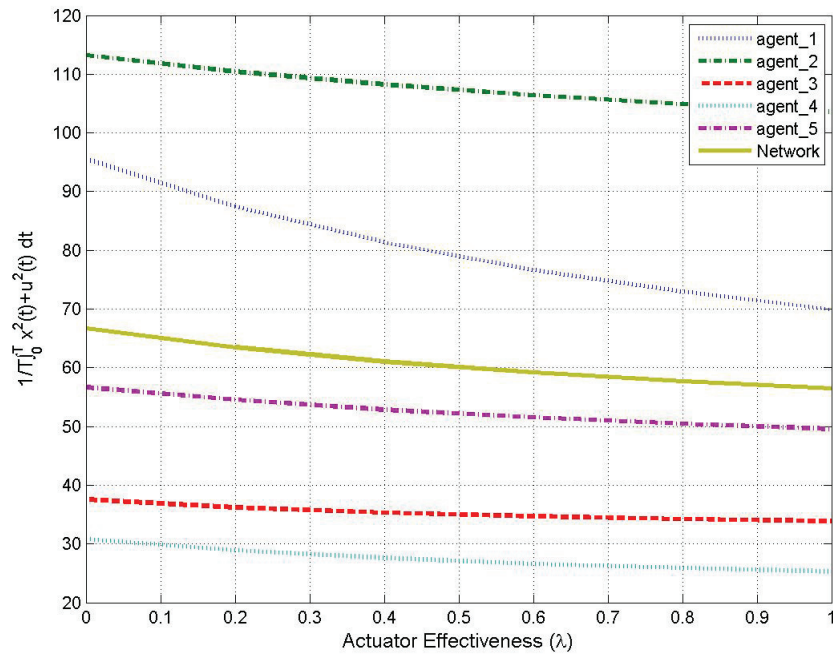


Figure 3.58: Total performance index for dynamic state feedback protocol without fault recovery - gain set 2.

3. In this case, the control gain is selected such that the contribution of the states in the total performance index is the highest among these three cases.

$$K = \begin{bmatrix} -0.1741 & -0.1922 & 0.0770 & -0.6129 \\ 0.3189 & 0.2800 & -0.0638 & 0.6867 \end{bmatrix} \quad (3.91)$$

The simulation results for the selected gain (3.91) are shown in Figures 3.59 - 3.61.

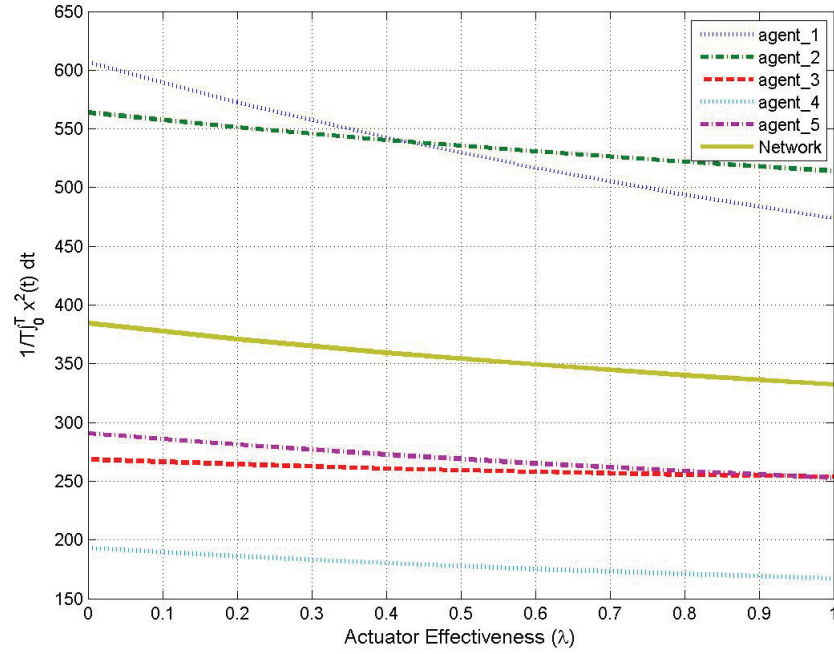


Figure 3.59: States performance index for dynamic state feedback protocol without fault recovery - gain set 3.

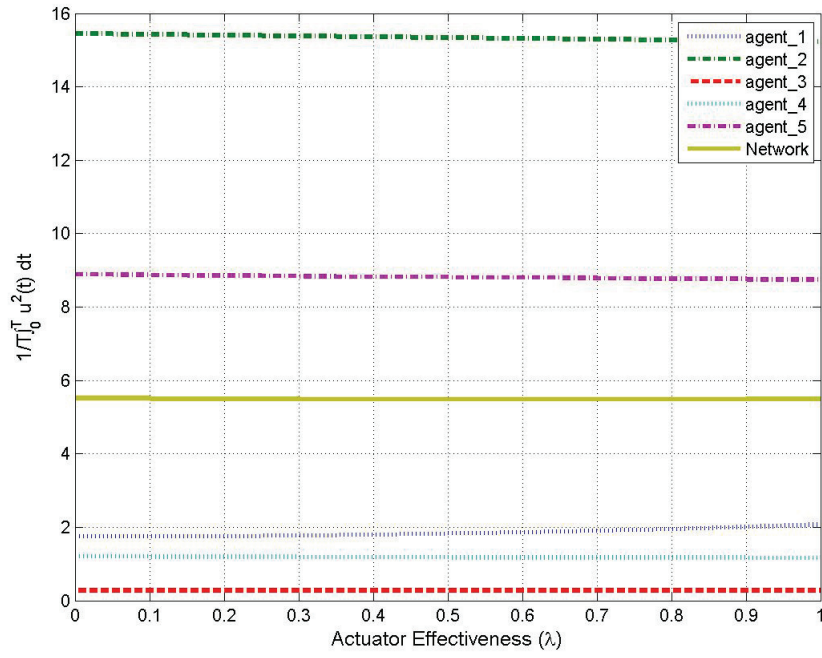


Figure 3.60: Control inputs performance index for dynamic state feedback protocol without fault recovery - gain set 3.

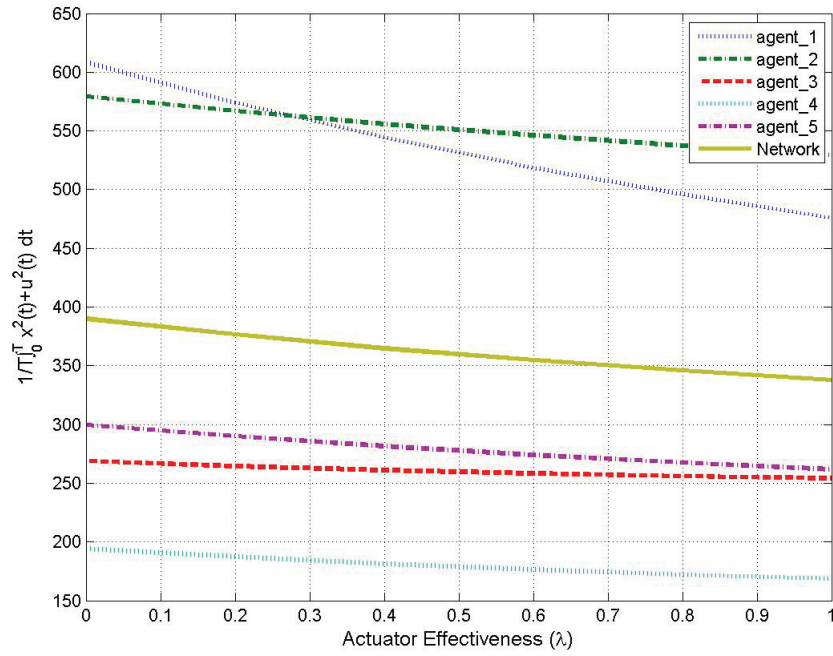


Figure 3.61: Total performance index for dynamic state feedback protocol without fault recovery - gain set 3.

In the above simulations, the contribution of the control input in the performance index is calculated using the control signal applied to the system and not the one calculated by the controller. This makes a difference in faulty case, where these two signals are not the same.

From Figures 3.53 - 3.61, it can be seen that as the control gain is selected such that the contribution of the states in the performance index decreases, the states performance index becomes less sensitive with respect to the fault. However, the less sensitivity of the performance index costs more control input.

**Remark 3.26.** *In the simulation results, the methodology of Section 3.11 is applied as a non-reconfigurable method, since for this system even after fault occurrence the performance indexes do not change drastically. However, if there is a significant change in the performance indexes, the approach should be applied in an active manner.*

## 3.12 Summary

In the first part of this chapter, which deals with healthy case, the synchronization of multi-agent systems under the following scenarios have been considered:

- Consensus subject to noise in Sections 3.3
- Consensus by employing a Kalman filter for state estimation in Section 3.4
- Consensus subject to model parameters uncertainties in Section 3.5

Among these, consensus subject to noise is one of the main contributions, where the consensus achievement in a stochastic mean square sense (MSS) has been shown. In the second part, multiple faulty scenarios as LIP, float and LOE faults have been considered. These scenarios are as follows:

- Consensus subject to LOE and float faults in Sections 3.6, 3.7, and 3.8
- Consensus subject to LIP fault by employing PI controller in Section 3.10
- Consensus achievement of integrator systems subject to LOE and float faults in Sections 3.9
- Reconfigurable dynamic synchronization protocol for a multi-input model subject to LOE and float faults in Section 3.11

Among these, the reconfigurable synchronization protocol subject to LIP fault and the reconfigurable dynamic synchronization protocol for a multi-input model subject LOE/float faults are among the main contributions.

# Chapter 4

## Output Feedback Synchronization

In this chapter, the problem of output feedback synchronization will be considered. Specifically, it is assumed that state information are not available anymore, and we only have access to partial state information. Similar to the previous chapter, the output feedback synchronization problem will be tackled subject to the both healthy and faulty scenarios.

First, the healthy scenario is considered under a perfect communication network. Then, the effect of noise on consensus achievement will be addressed. The analysis of the effect of uncertainty is skipped, due to its excessive similarity to state feedback. Then, the faulty scenarios will be studied where a static as well as two different dynamic protocols are developed. The first dynamic methodology exploits a dynamic observer as well as a dynamic controller; however, the second approach only employs a dynamic controller.

### 4.1 Output Feedback Synchronization and Luenberger Observer

Consider a group of  $N$  homogeneous multi-agent systems as

$$\begin{aligned}\dot{\mathbf{x}}_i(t) &= A\mathbf{x}_i(t) + B\mathbf{u}_i(t) \\ y_i(t) &= C\mathbf{x}_i(t) \quad i = 1, 2, \dots, N\end{aligned}\tag{4.1}$$

where  $A \in \mathbb{R}^{n \times n}$  is a marginally stable matrix,  $B \in \mathbb{R}^{n \times m}$ ,  $C \in \mathbb{R}^{p \times n}$  and system  $(A, B, C)$  is stabilizable using a static output feedback.

**Remark 4.1.** *Since the system  $(A, B, C)$  is stabilizable using static output feedback, one may conclude that the pair  $(A, B)$  is stabilizable and  $(A, C)$  is detectable [111].*

**Remark 4.2.** *The formulation of this section is general, and it covers both single-input and multi-input systems.*

Here to deal with synchronization problem, a dynamic protocol is employed in Lemma 4.1. This method is adopted from the literature [36] with required modifications.

**Lemma 4.1.** *Under the following dynamic controller*

$$\begin{aligned}\dot{\zeta}_i(t) &= (A + BK)\zeta_i(t) + \sum_{j \in \mathcal{N}_i} a_{ij}(\zeta_j(t) - \zeta_i(t) + \hat{\mathbf{x}}_i(t) - \hat{\mathbf{x}}_j(t)) \\ \dot{\hat{\mathbf{x}}}_i &= A\hat{\mathbf{x}}_i + B\mathbf{u}_i + H(\hat{y}_i - y_i) \\ \mathbf{u}_i(t) &= K\zeta_i(t) \quad i = 1, \dots, N\end{aligned}\tag{4.2}$$

the system (4.1) will synchronize.

where  $\zeta_i \in \mathbb{R}^n$ ,  $B \in \mathbb{R}^{n \times m}$  and  $K \in \mathbb{R}^{m \times n}$  are respectively the controller state, control input matrix, and the control gain associated with the agent  $i$ .

*Proof.* By defining the new state variables as  $s_i(t) = \hat{\mathbf{x}}_i(t) - \zeta_i(t)$  and  $e_i(t) = \mathbf{x}_i(t) - \hat{\mathbf{x}}_i(t)$ , the augmented state-space system is

$$\dot{\zeta}_i(t) = (\tilde{A}_N + \tilde{B}_N \tilde{K}_N)\zeta_i(t) + Ls(t) \tag{4.3a}$$

$$\dot{s}(t) = \tilde{A}_N s(t) - Ls(t) + \tilde{H}\tilde{C}e(t) \tag{4.3b}$$

$$\dot{e}(t) = (\tilde{A}_N + \tilde{H}\tilde{C})e(t) \tag{4.3c}$$

This could be rewritten as

$$\begin{bmatrix} \dot{\zeta}(t) \\ \dot{s}(t) \\ \dot{e}(t) \end{bmatrix} = \begin{bmatrix} (\tilde{A}_N + \tilde{B}_N \tilde{K}_N) & L & 0 \\ 0 & \tilde{A}_N - L & \tilde{H}\tilde{C} \\ 0 & 0 & (\tilde{A}_N + \tilde{H}\tilde{C}) \end{bmatrix} \begin{bmatrix} \zeta(t) \\ s(t) \\ e(t) \end{bmatrix} \tag{4.4}$$

where  $L = [l_{ij}]$  is the Laplacian matrix of the system, with  $l_{ii} = \sum_{j \in \mathcal{N}_i} a_{ij}$  and  $l_{ij} = -a_{ij}$  for  $i \neq j$ ,  $\tilde{A}_N = I_N \otimes A$ ,  $\tilde{C}_N = I_N \otimes C$ ,  $\tilde{H}_N = I_N \otimes H$ ,  $\tilde{B}_N = I_N \otimes B$ ,  $\tilde{K}_N = I_N \otimes K$ ,  $\zeta = [\zeta_1^T, \zeta_2^T, \dots, \zeta_N^T]^T$ ,  $s = [s_1^T, s_2^T, \dots, s_N^T]^T$ ,  $e = [e_1^T, e_2^T, \dots, e_N^T]^T$ .

Knowing that the pair  $(A, C)$  is observable, there exists a matrix  $H$  that makes the equation (4.3c) stabilize. Moreover, using the result of Lemma 2.2, it is guaranteed that  $s_i(t)$  in equation (4.3b) asymptotically synchronizes and the synchronization trajectory will be governed by  $\dot{s}_c(t) = As_c(t)$ . This is concluded by using the fact that Laplacian is a zero row sum matrix.

Hence, when the synchronization trajectory (4.3b) reaches its steady state and therefore  $Ls(t)$  reaches zero, the augmented dynamics of the controller (4.3a) reduces to

$$\dot{\zeta} = (\tilde{A}_N + \tilde{B}_N \tilde{K}_N) \zeta \quad (4.5)$$

This implies when the control gain  $\tilde{K}_N$  is designed to make  $\tilde{A}_N + \tilde{B}_N \tilde{K}_N$  Hurwitz, the dynamics of (4.3) will be exponentially stable and will asymptotically reach zero.

Finally,  $x_i(t)$ , which is obtained by  $x_i(t) = e_i(t) + s_i(t) + \zeta_i(t)$ , converges to  $s_i(t)$ . Therefore, one can conclude that the state of all agents will reach to a common solution of  $\dot{s}_c(t) = As_c(t)$ .  $\square$

### 4.1.1 Simulation Results

In this section, the synchronization of autonomous underwater vehicles (AUVs) in the longitudinal plane by having partial state information, using the measurement matrix in (2.13), is investigated. For the sake of simulation, the single-input model (3.13) is considered.

For simulations purposes, the control gain  $K$  and the Luenberger observer gain  $H$  are designed as

$$K = \begin{bmatrix} -0.3812 & -0.1793 & 0.3063 & -1.5953 \end{bmatrix}$$

$$H = \begin{bmatrix} 3.1046 & 0.2406 & 0.0780 \\ -2.0866 & 0.5573 & -0.3120 \\ 0.4008 & 2.7676 & -2.0000 \\ 1.0000 & 0 & 1.0000 \end{bmatrix} \quad (4.6)$$

In the following, first the convergence of state estimation to states are presented in Figures 4.1 - 4.4. Then, as it can be seen in Figures 4.5 - 4.14, the controllers' as well as the agents' states reach consensus. Finally, Figures 4.9 - 4.10 show that control input is limited between  $[-60 \ 60]$ .

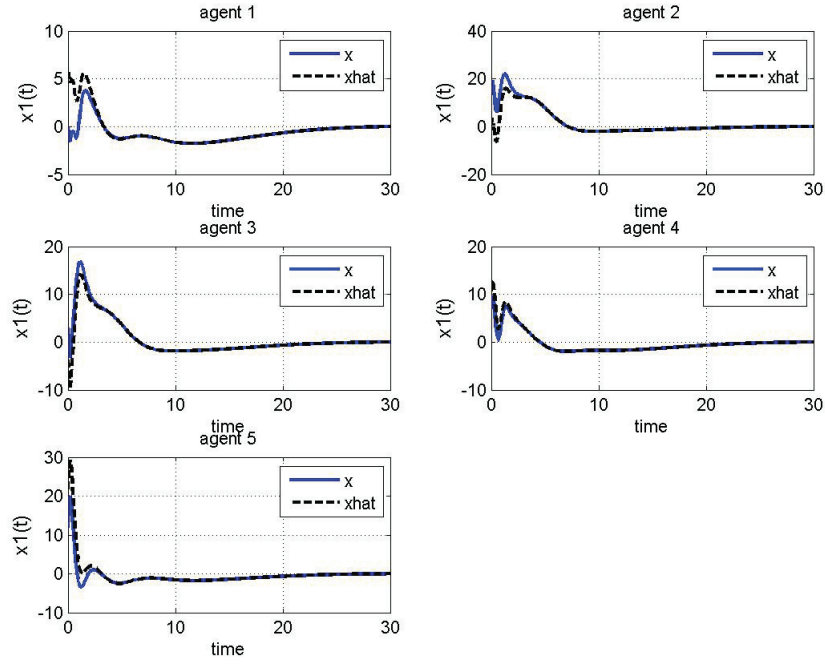


Figure 4.1: State estimation of state 1 for dynamic output feedback protocol under healthy scenario.

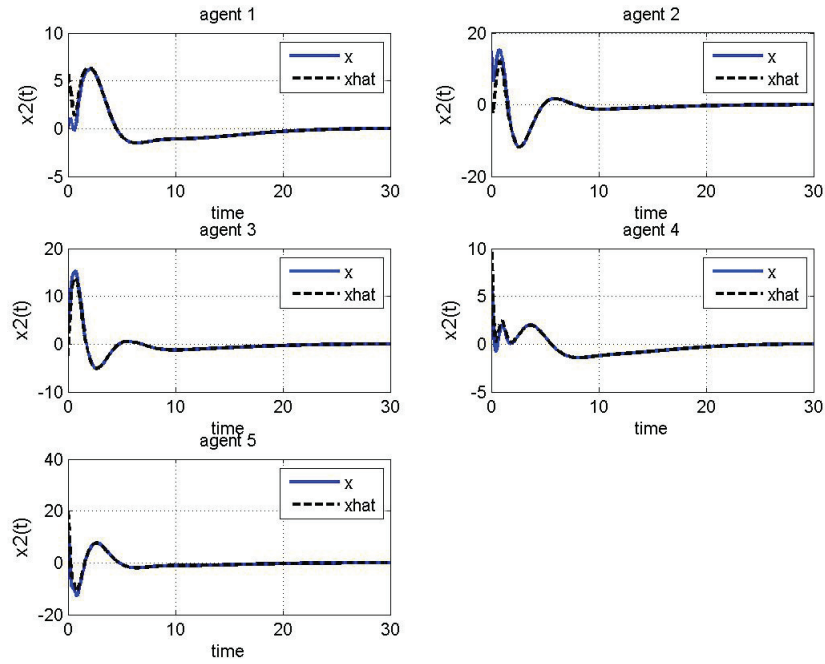


Figure 4.2: State estimation of state 2 for dynamic output feedback protocol under healthy scenario.

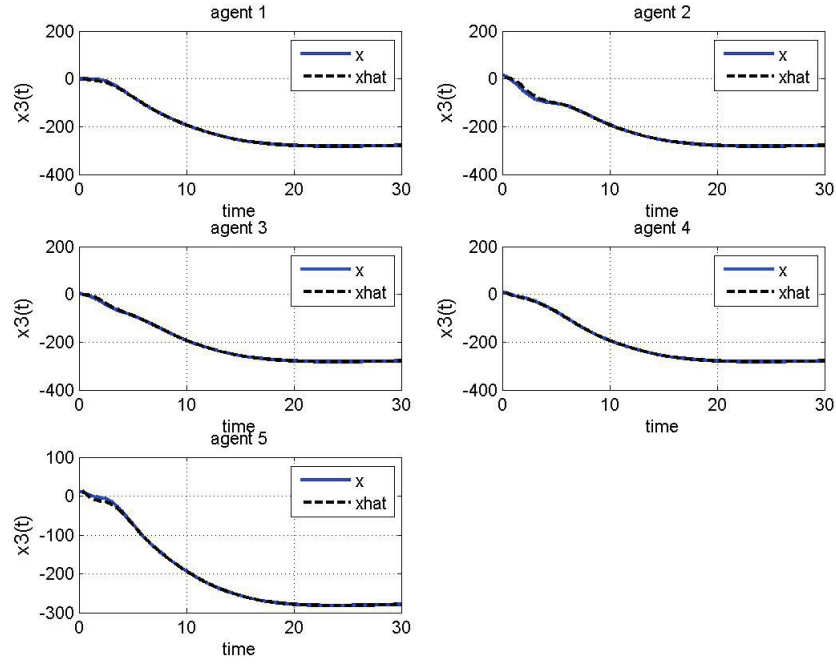


Figure 4.3: State estimation of state 3 for dynamic output feedback protocol under healthy scenario.

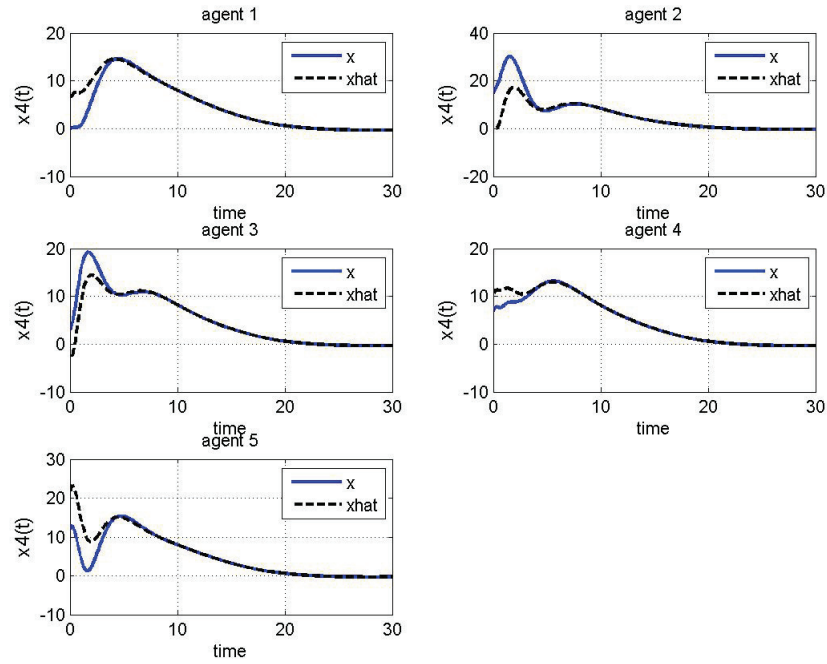


Figure 4.4: State estimation of state 4 for dynamic output feedback protocol under healthy scenario.

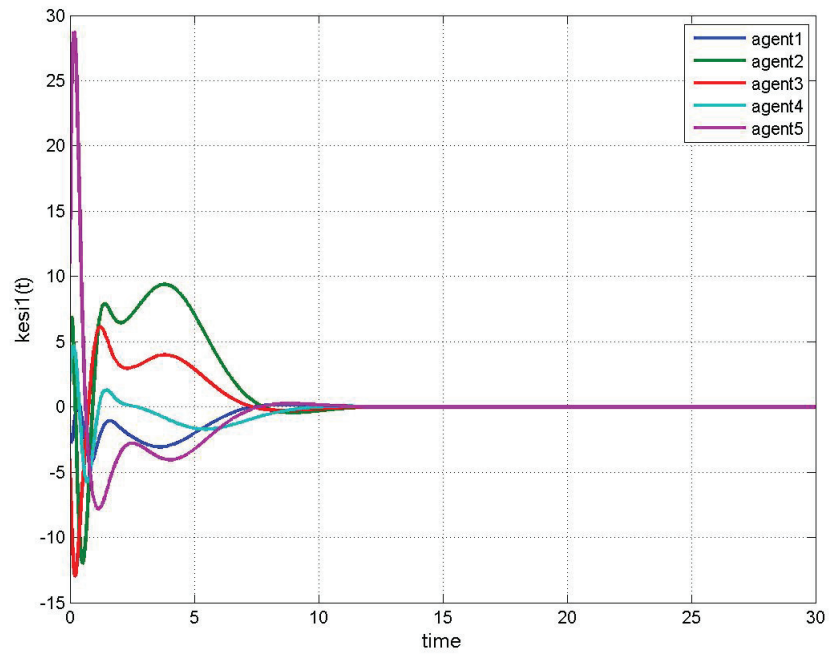


Figure 4.5: Controller state 1 for dynamic output feedback protocol under healthy scenario.

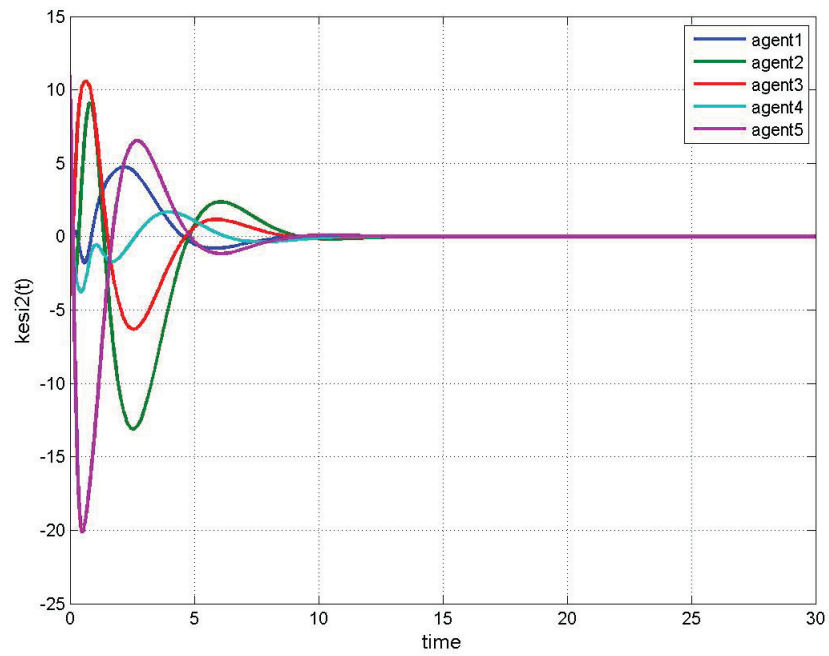


Figure 4.6: Controller state 2 for dynamic output feedback protocol under healthy scenario.

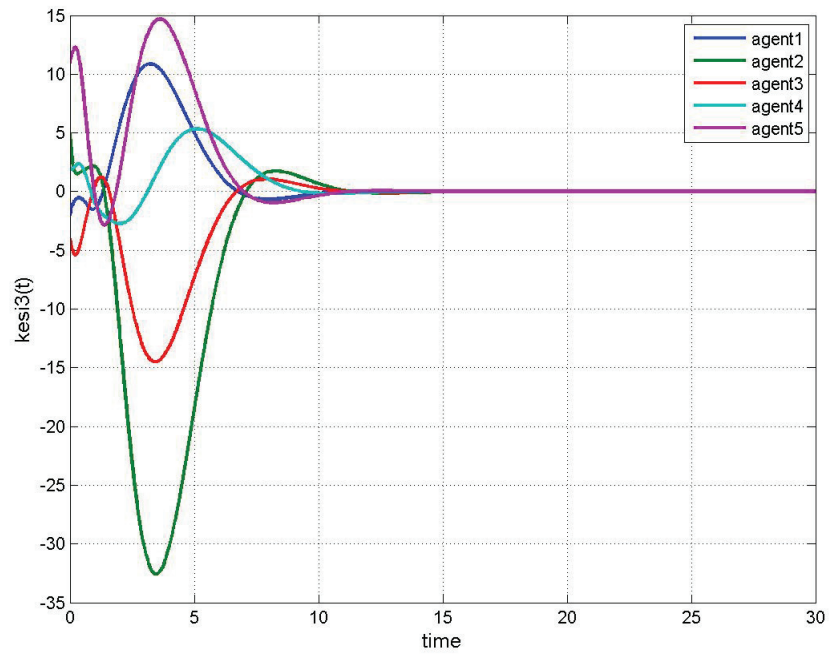


Figure 4.7: Controller state 3 for dynamic output feedback protocol under healthy scenario.

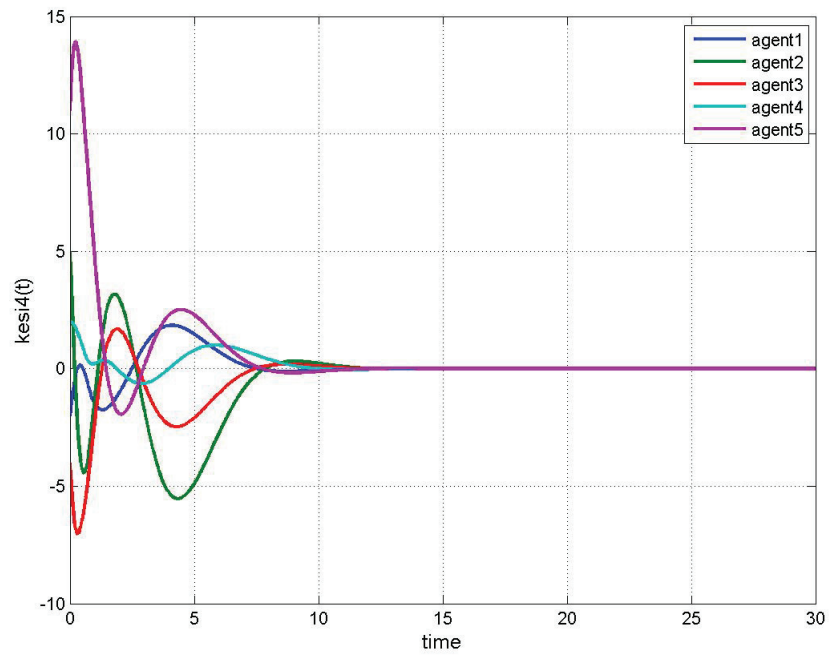


Figure 4.8: Controller state 4 for dynamic output feedback protocol under healthy scenario.

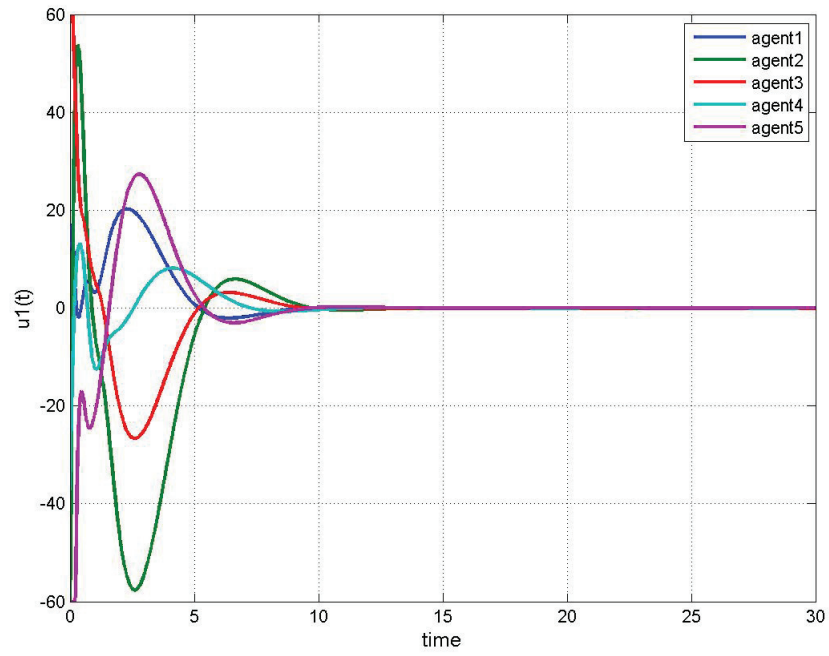


Figure 4.9: Control Input signal  $u_1(t)$  for dynamic output feedback protocol under healthy scenario.

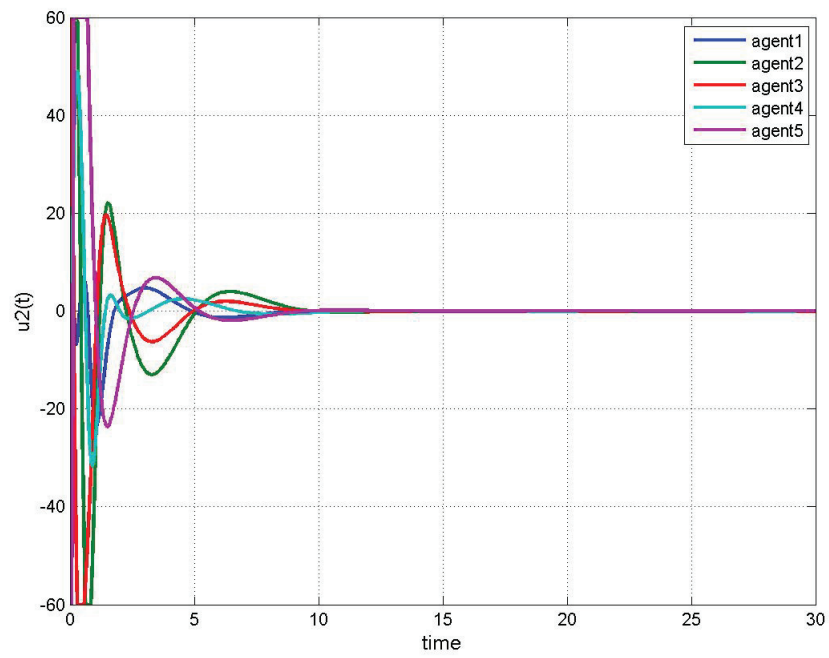


Figure 4.10: Control input signal  $u_2(t)$  for dynamic output feedback protocol under healthy scenario.

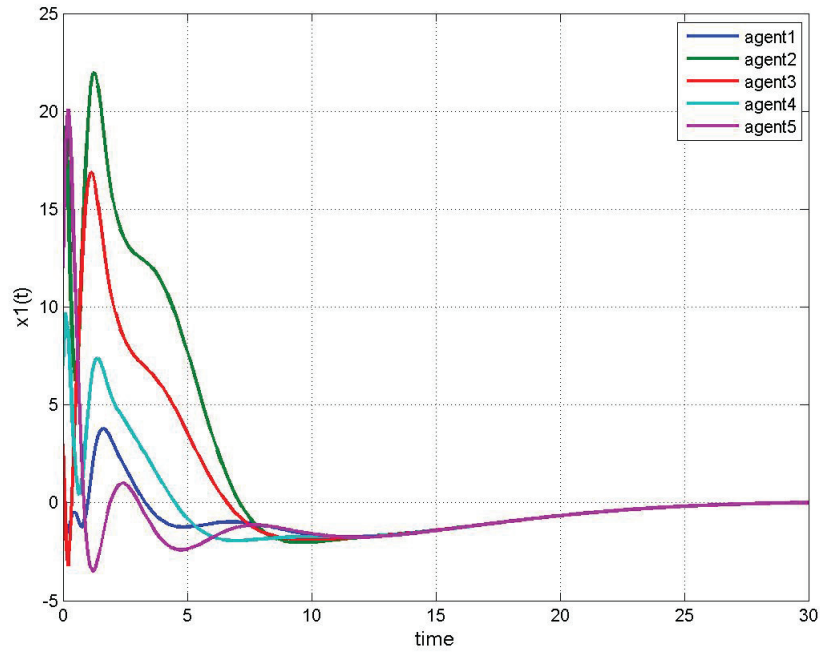


Figure 4.11: Synchronization of state 1 for dynamic output feedback protocol under healthy scenario.

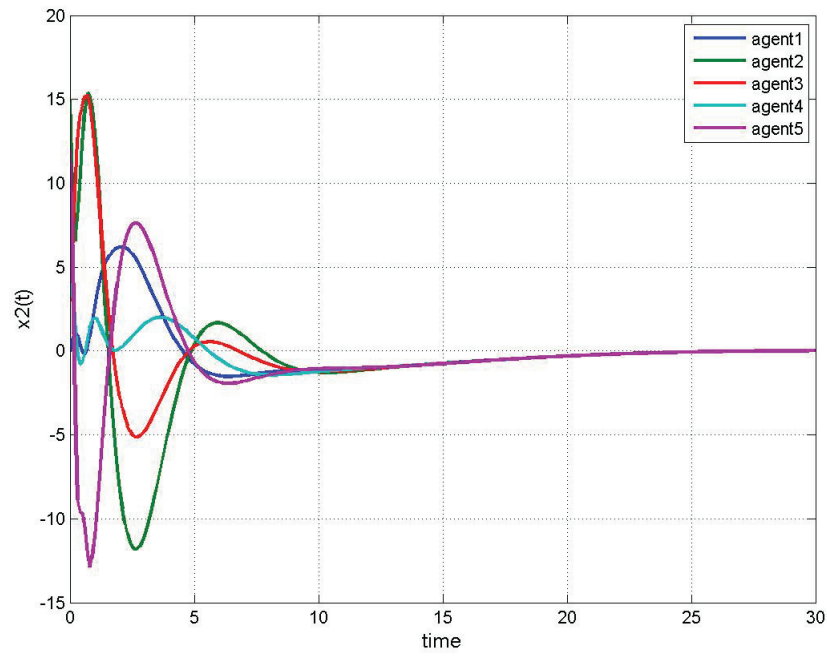


Figure 4.12: Synchronization of state 2 for dynamic output feedback protocol under healthy scenario.

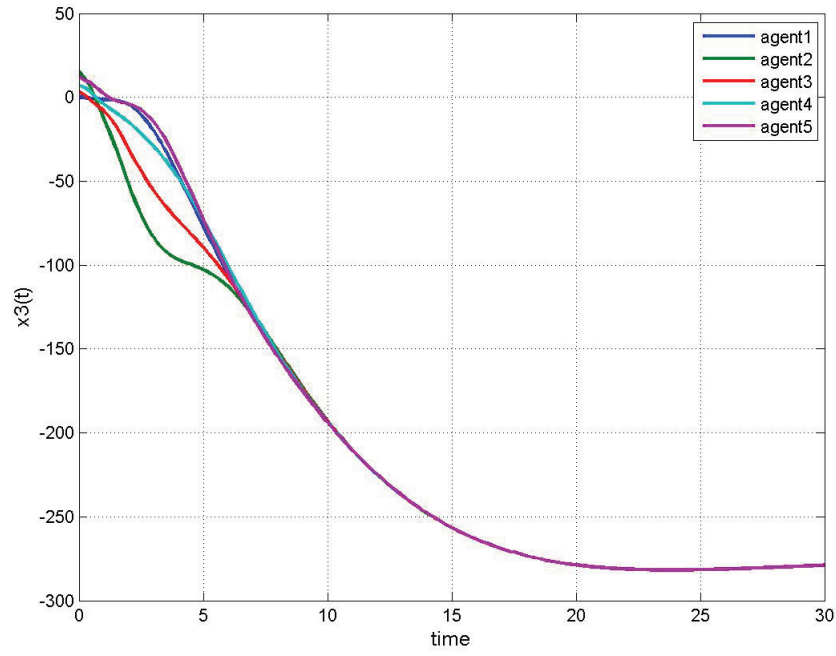


Figure 4.13: Synchronization of state 3 for dynamic output feedback protocol under healthy scenario.

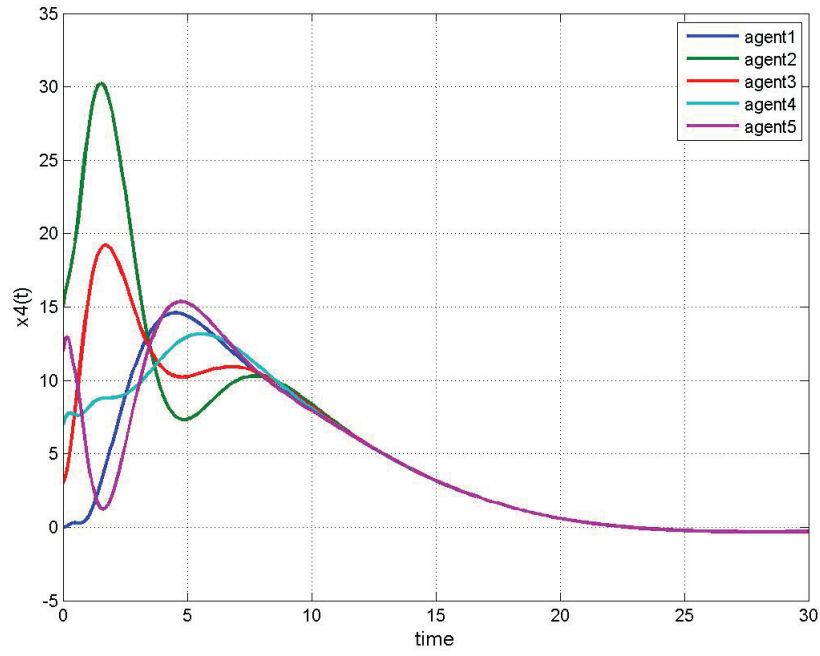


Figure 4.14: Synchronization of state 4 for dynamic output feedback protocol under healthy scenario.

## 4.2 Output Feedback Synchronization and Kalman Filtering

As in the previous chapter, the presence of noise on the states and measurement leads to a large covariance error on states. In order to fix this problem, Kalman filtering is employed for states estimation that are used in synchronization protocol.

Now, assume that there is an added noise on the agents' dynamics and their measurements as

$$\begin{aligned}\dot{\mathbf{x}}_i(t) &= A\mathbf{x}_i(t) + B\mathbf{u}_i(t) + Bw_i(t), \quad i = 1, 2, \dots, N \\ y_i(t) &= C\mathbf{x}_i(t) + v_i(t)\end{aligned}\tag{4.7}$$

where  $w_i \in \mathbb{R}^n$ ,  $v_i \in \mathbb{R}^n$ , and the statistics of the states initial condition, disturbance and measurement noises are assumed to be known. The statistics of the states' initial conditions are

$$\begin{aligned}E[\mathbf{x}_i(0)] &= \hat{\mathbf{x}}_{i0} \quad i = 1, 2, \dots, N \\ E\{[\mathbf{x}_i(0) - \hat{\mathbf{x}}_{i0}][\mathbf{x}_i(0) - \hat{\mathbf{x}}_{i0}]^T\} &= P_{i0}\end{aligned}\tag{4.8}$$

The term  $w_i(t)$  accounts for the environmental disturbances and could be modeled as a zero mean Gaussian white noise process as

$$\begin{aligned}E[w_i(t)] &= 0 \quad i = 1, 2, \dots, N \\ E\{w_i(t)w_i(\tau)^T\} &= Q_i(t)\delta(t - \tau)\end{aligned}\tag{4.9}$$

which is specified by its spectral density matrix  $Q_i(t)$ . The measurement noise is a zero mean Gaussian white noise process as

$$\begin{aligned}E[v_i(t)] &= 0 \quad i = 1, 2, \dots, N \\ E\{v_i(t)v_i(\tau)^T\} &= R_i(t)\delta(t - \tau)\end{aligned}\tag{4.10}$$

where the measurement uncertainty is expressed by its spectral density matrix  $R_i(t)$ , which is a  $p \times p$  matrix.

It is also assumed that the states and measurement noises are uncorrelated. For the sake of simplicity, the matrices  $R_i$  and  $Q_i$  are to be constant.

**Lemma 4.2.** *Under the following dynamic controller*

$$\begin{aligned}
\dot{\zeta}_i(t) &= (A + BK)\zeta_i(t) + \sum_{j \in \mathcal{N}_i} a_{ij}(\zeta_j(t) - \zeta_i(t) + \hat{\mathbf{x}}_i(t) - \hat{\mathbf{x}}_j(t)) \\
\dot{\hat{\mathbf{x}}}_i(t) &= A\hat{\mathbf{x}}_i(t) + B\mathbf{u}_i(t) + H_i(\hat{y}_i(t) - y_i(t)) \\
\mathbf{u}_i(t) &= K\zeta_i(t) \quad i = 1, \dots, N
\end{aligned} \tag{4.11}$$

the system in (4.7) will synchronize in a mean square sense.

where  $\zeta_i(t) \in \mathbb{R}^n$ ,  $B \in \mathbb{R}^{n \times m}$ ,  $K \in \mathbb{R}^{m \times n}$ , and  $H_i$  are respectively controller state, control input matrix, and the controller and observer gains associated with agent  $i$ . The index dependence of  $H_i(t)$  shows that it does not necessarily needs to be identical for all agents.

*Proof.* By defining the new state-space variables as  $s_i(t) = \hat{\mathbf{x}}_i(t) - \zeta_i(t)$  and  $e_i(t) = \mathbf{x}_i(t) - \hat{\mathbf{x}}_i(t)$ , the augmented state-space representations of their expected values become

$$\dot{\bar{\zeta}}(t) = (\tilde{A}_N + \tilde{B}_N \tilde{K}_N) \bar{\zeta}(t) + L \bar{s}(t) \tag{4.12a}$$

$$\dot{\bar{s}}(t) = \tilde{A}_N \bar{s}(t) - L \bar{s}(t) + \tilde{H} \tilde{C} \bar{e}(t) \tag{4.12b}$$

$$\dot{\bar{e}}(t) = (\tilde{A}_N + \tilde{H} \tilde{C}) \bar{e}(t) \tag{4.12c}$$

Equation (4.12) could be rewritten as

$$\begin{bmatrix} \dot{\bar{\zeta}}(t) \\ \dot{\bar{s}}(t) \\ \dot{\bar{e}}(t) \end{bmatrix} = \begin{bmatrix} (\tilde{A}_N + \tilde{B}_N \tilde{K}_N) & L & 0 \\ 0 & \tilde{A}_N - L & \tilde{H} \tilde{C} \\ 0 & 0 & (\tilde{A}_N + \tilde{H} \tilde{C}) \end{bmatrix} \begin{bmatrix} \bar{\zeta}(t) \\ \bar{s}(t) \\ \bar{e}(t) \end{bmatrix} \tag{4.13}$$

where  $L = [l_{ij}]$  is the Laplacian matrix of the system, with  $l_{ii} = \sum_{j \in \mathcal{N}_i} a_{ij}$  and  $l_{ij} = -a_{ij}$  for  $i \neq j$ ,  $\tilde{A}_N = I_N \otimes A$ ,  $\tilde{C}_N = I_N \otimes C$ ,  $\tilde{H}_N = \text{diag}(H_1, H_2, \dots, H_N)$ ,  $\tilde{B}_N = I_N \otimes B$ ,  $\tilde{K}_N = I_N \otimes K$ ,  $\bar{\zeta} = [\bar{\zeta}_1^T, \bar{\zeta}_2^T, \dots, \bar{\zeta}_N^T]^T$ ,  $\bar{s} = [\bar{s}_1^T, \bar{s}_2^T, \dots, \bar{s}_N^T]^T$ ,  $\bar{e} = [\bar{e}_1^T, \bar{e}_2^T, \dots, \bar{e}_N^T]^T$ , and  $E\{e_i(t)\} = \bar{e}_i(t)$ ,  $E\{s_i(t)\} = \bar{s}_i(t)$  and  $E\{\zeta_i(t)\} = \bar{\zeta}_i(t)$  for  $i = 1, 2, \dots, N$ . For the sake of simplicity, time-dependence of the variables is omitted.

First, it will be shown that the expected values of  $e_i(t)$ s (and consequently  $\bar{e}(t)$  in (4.13)) go to zero. To do so, a Kalman-Bucy filter is employed to estimate the entire state vector based upon the

observed data as

$$\begin{aligned}\dot{\hat{\mathbf{x}}}_i(t) &= A\hat{\mathbf{x}}_i(t) + B\mathbf{u}_i(t) + H_i(y_i(t) - \hat{y}_i(t)), \quad i = 1, 2, \dots, N \\ \hat{y}_i(t) &= C\hat{\mathbf{x}}_i(t)\end{aligned}\tag{4.14}$$

where  $H_i$  will be the steady state solution for

$$H_i = \lim_{t \rightarrow \infty} P_i(t) C^T R_i^{-1} \quad i = 1, 2, \dots, N\tag{4.15}$$

where  $P_i(t)$  is the solution to

$$\begin{aligned}\dot{P}_i(t) &= AP_i(t) + P_i(t)A^T + BQ_iB^T - H_i(t)CP_i(t) = \\ &AP_i(t) + P_i(t)A^T + BQ_iB^T - P_i(t)C^T R_i^{-1} CP_i(t), \quad i = 1, 2, \dots, N\end{aligned}\tag{4.16}$$

It is clear that  $H_i$  cannot exist if  $R_i$  is singular. Also, for a stable estimation filter,  $R_i$  must be positive-definite. For the sake of simplicity, equation (4.16) has been solved in steady state. In other words,  $P_i$ , which is obtained by solving an algebraic Riccati equation, is a fixed matrix. Therefore, the Kalman-Bucy gain  $H_i$  obtained out of equation (4.15) is fixed and still optimum in steady state.

By defining the new state variable as  $e_i(t) = \mathbf{x}_i(t) - \hat{\mathbf{x}}_i(t)$ , the state estimation error dynamics becomes

$$\begin{aligned}\dot{e}_i(t) &= Ae_i(t) + Bw_i(t) - H_i(y_i(t) - \hat{y}_i(t)) = \\ &(A - H_iC)e_i(t) + Bw_i(t) + H_iv_i(t), \quad i = 1, 2, \dots, N \\ y_i(t) &= Ce_i(t)\end{aligned}\tag{4.17}$$

Similar to Lemma , it can be shown that if the Kalman filter gain  $H_i$  is chosen such that  $A - H_iC < 0$ , and knowing that  $w_i(t)$  and  $v_i(t)$  are zero mean white Gaussian noises, the expected value of state estimation error  $E[e_i(t)]$  goes to zero. Moreover, if the Kalman gain  $H_i$  also satisfies (4.15), where  $P_i$  is obtained as the steady state solution of (4.16), the state's error covariance will be minimum.

When  $E[e_i(t)]$  goes to zero, equation (4.12) reduces to equation (3.87). Therefore, the rest of

proof becomes identical to Lemma 3.8. □

### 4.2.1 Simulation Results

In this section, the synchronization of the AUVs' longitudinal plane model (3.13), by having partial state information with measurement matrix (2.13), is considered. For simulation purposes, the spectral density matrix of the state and measurement noises are considered to be identical for all agents as  $Q = I_{(m \times m)}$  and  $R = 0.5 \times I_{(n \times n)}$ , respectively.

For simulations purposes, the control gain  $K$  and Kalman-Bucy filter gain  $H$  are designed as

$$\begin{aligned} K &= \begin{bmatrix} -0.3812 & -0.1793 & 0.3063 & -1.5953 \end{bmatrix} \\ H &= \begin{bmatrix} -0.1795 & 0.0041 & -0.0107 \\ 0.3790 & -0.0251 & 0.0511 \\ -0.0251 & 1.0237 & -0.2797 \\ 0.0511 & -0.2797 & 0.1463 \end{bmatrix} \end{aligned} \quad (4.18)$$

In the following figures, results for the state estimation are first presented in Figures 4.15 - 4.18. Then, the average consensus of the controllers' states as well as the agents' states will be shown in Figures 4.19 - 4.28.

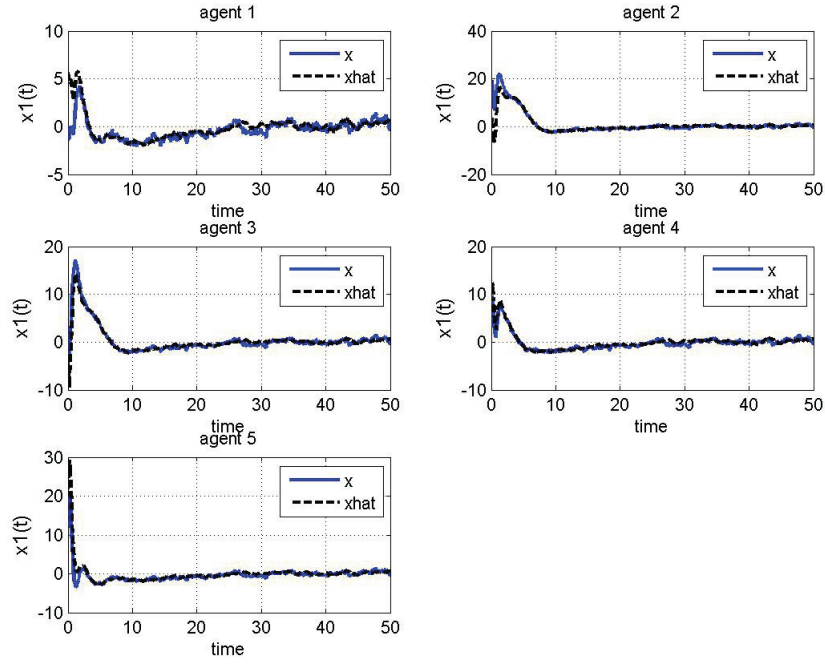


Figure 4.15: State estimation of state 1 for dynamic output feedback protocol subject to noise and by employing Kalman filter.

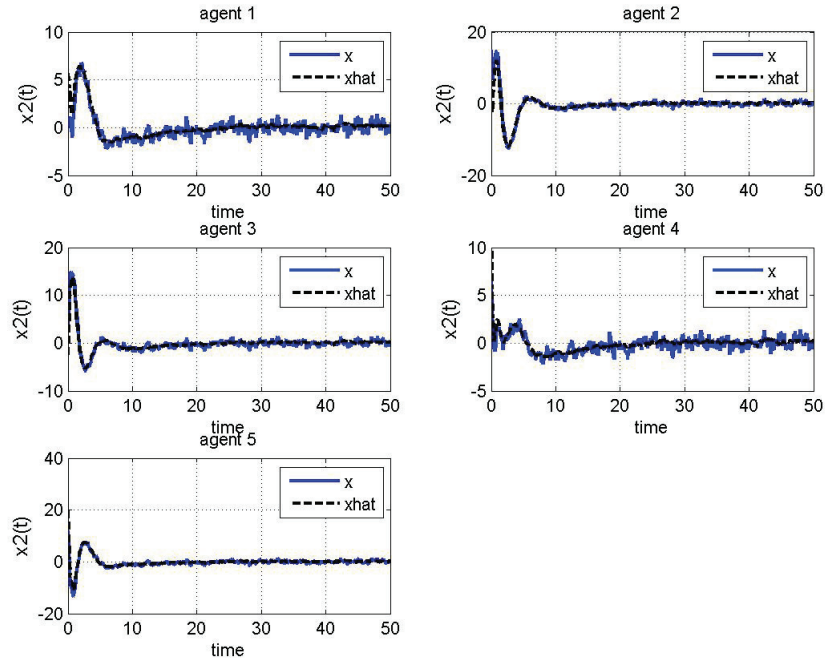


Figure 4.16: State estimation of state 2 for dynamic output feedback protocol subject to noise and by employing Kalman filter.

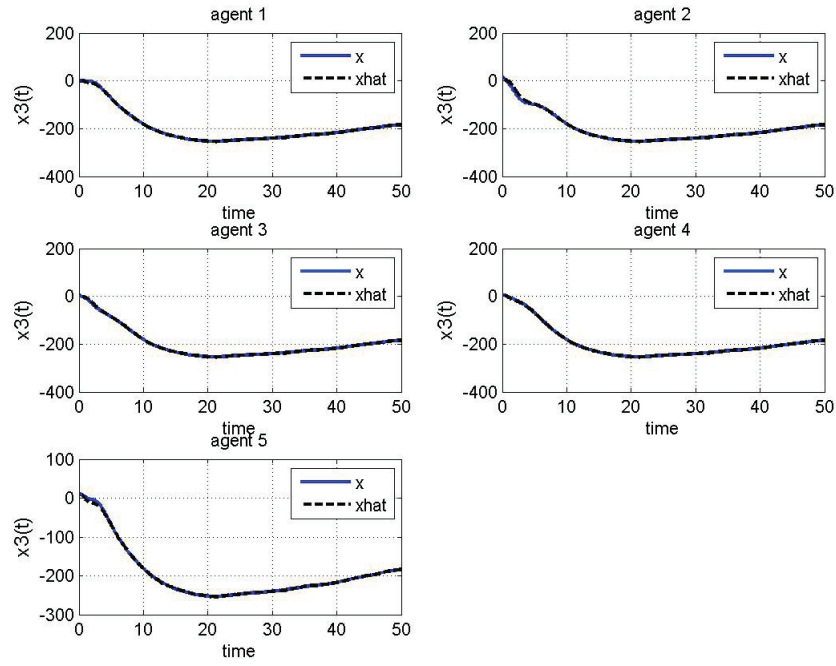


Figure 4.17: State estimation of state 3 for dynamic output feedback protocol subject to noise and by employing Kalman filter.

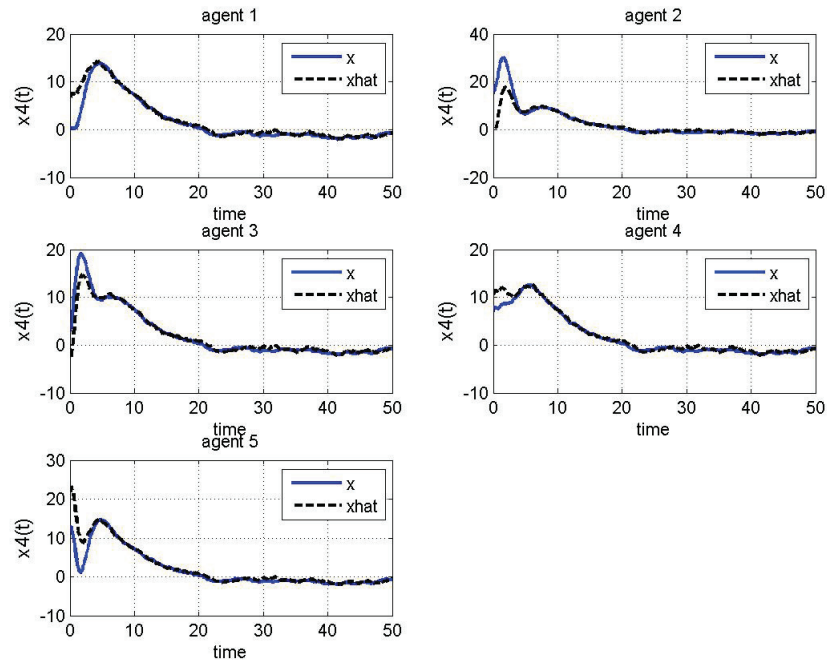


Figure 4.18: State estimation of state 4 for dynamic output feedback protocol subject to noise and by employing Kalman filter.

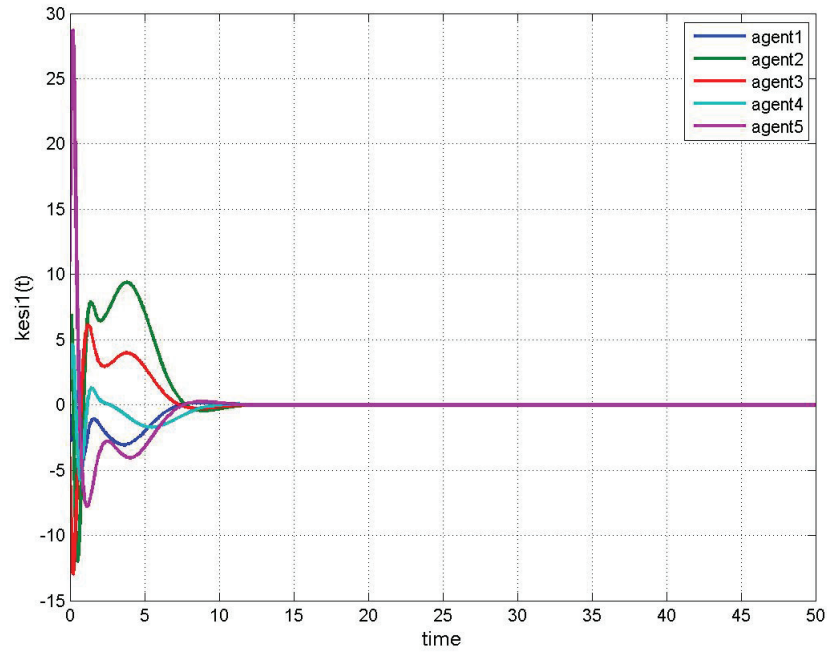


Figure 4.19: Controller state 1 for dynamic output feedback protocol subject to noise and by employing Kalman filter.

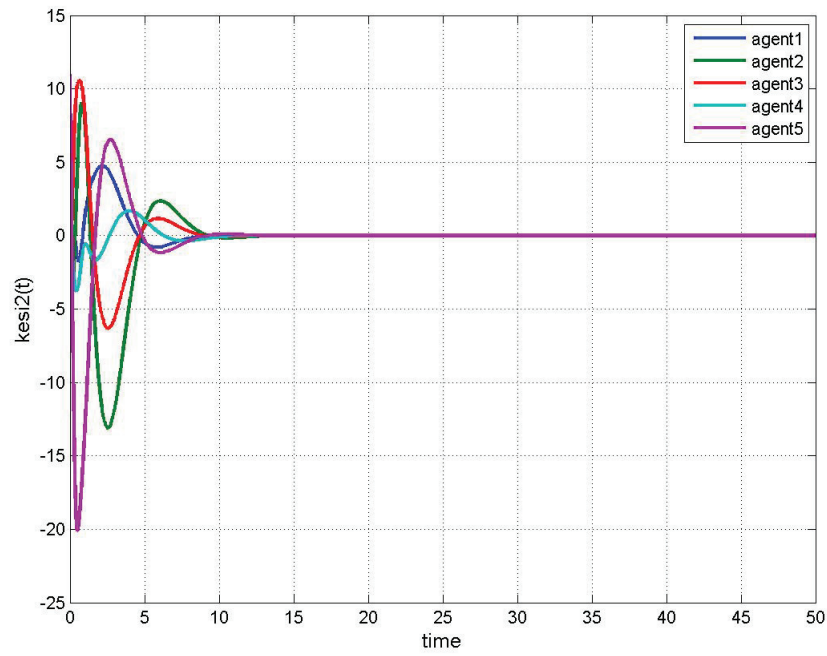


Figure 4.20: Controller state 2 for dynamic output feedback protocol subject to noise and by employing Kalman filter.

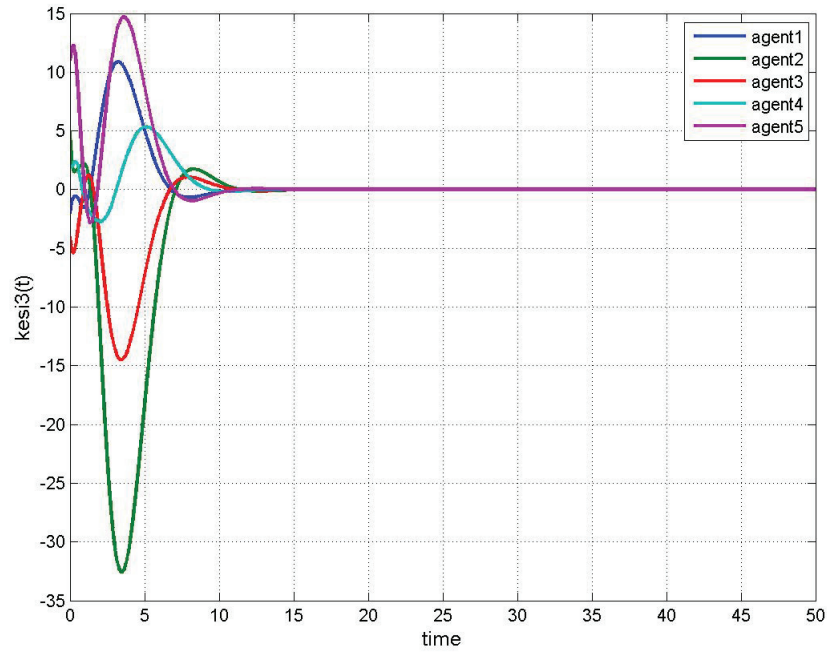


Figure 4.21: Controller state 3 for dynamic output feedback protocol subject to noise and by employing Kalman filter.

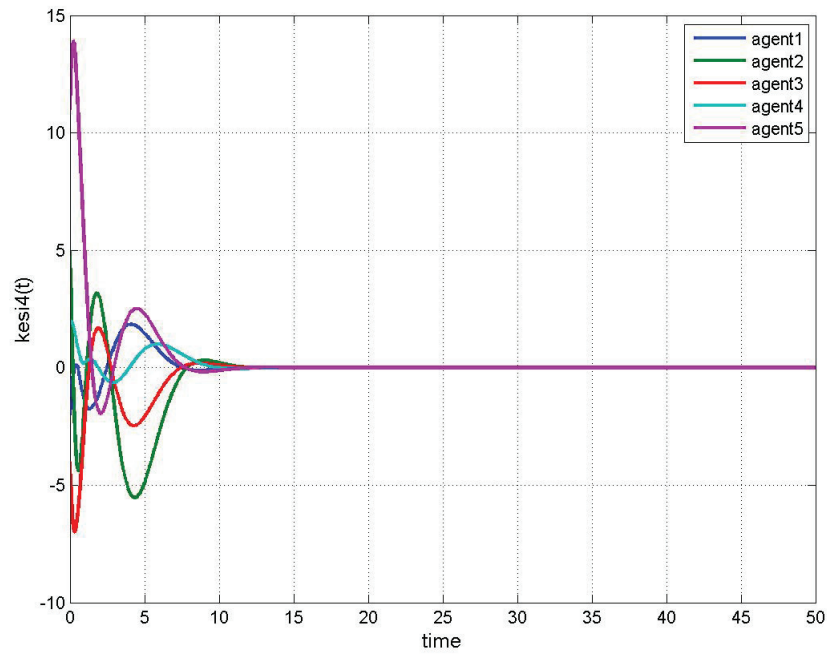


Figure 4.22: Controller state 4 for dynamic output feedback protocol subject to noise and by employing Kalman filter.

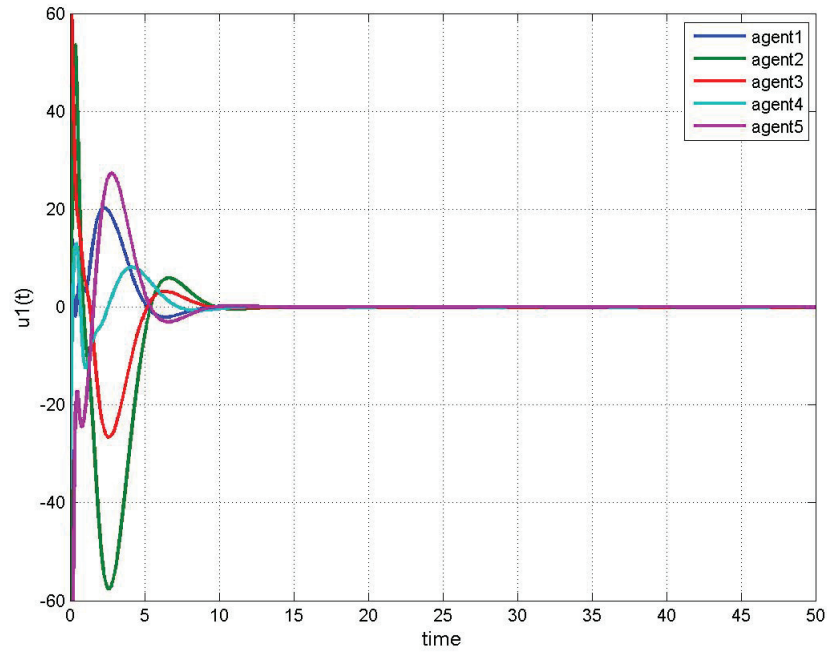


Figure 4.23: Control input signal  $u_1(t)$  for dynamic output feedback protocol subject to noise and by employing Kalman filter.

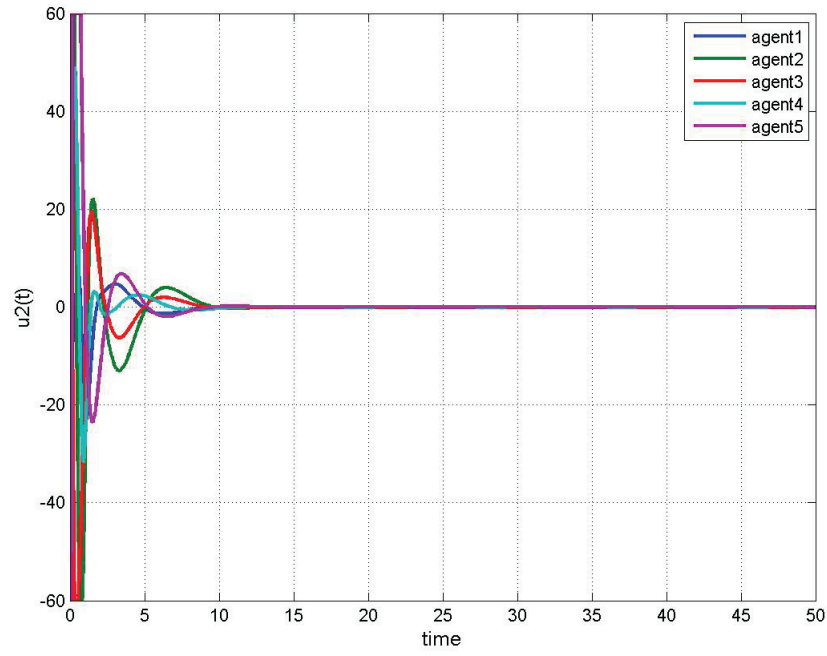


Figure 4.24: Control input signal  $u_2(t)$  for dynamic output feedback protocol subject to noise and by employing Kalman filter.

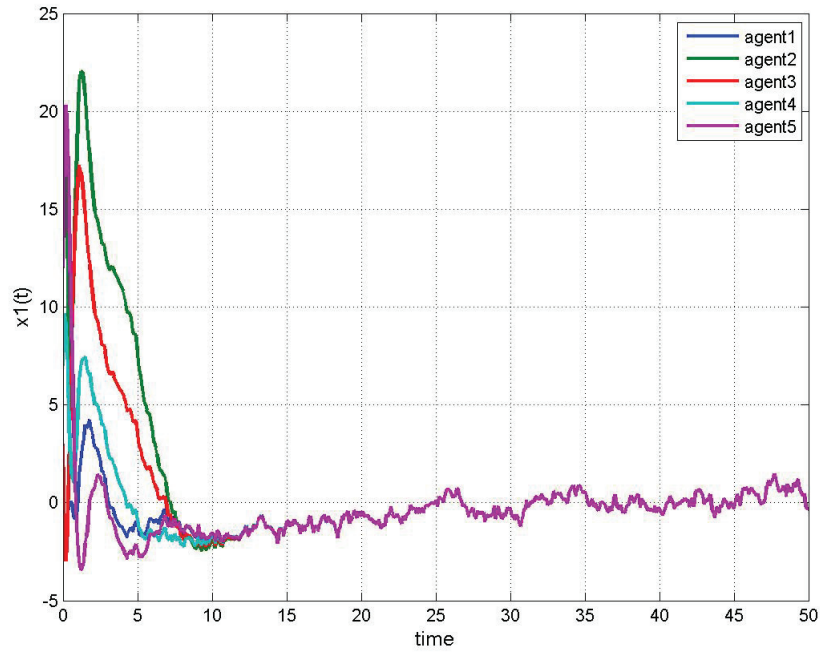


Figure 4.25: Synchronization of state 1 for dynamic output feedback protocol subject to noise and by employing Kalman filter.

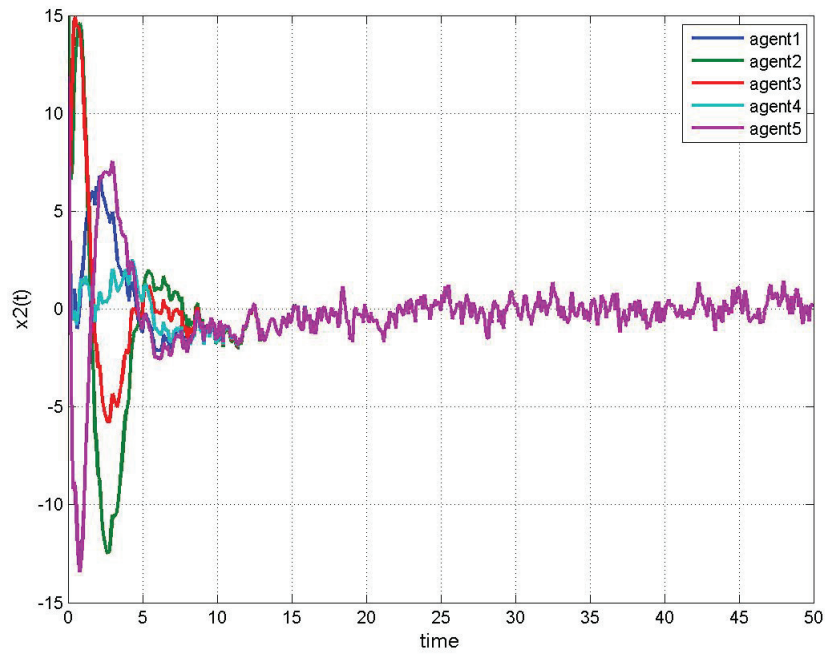


Figure 4.26: Synchronization of state 2 for dynamic output feedback protocol subject to noise and by employing Kalman filter.

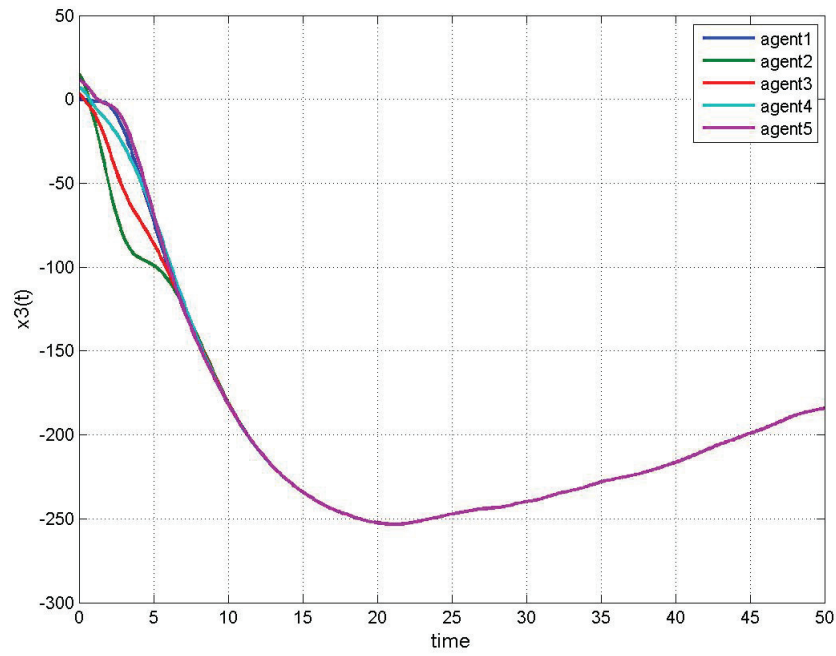


Figure 4.27: Synchronization of state 3 for dynamic output feedback protocol subject to noise and by employing Kalman filter.

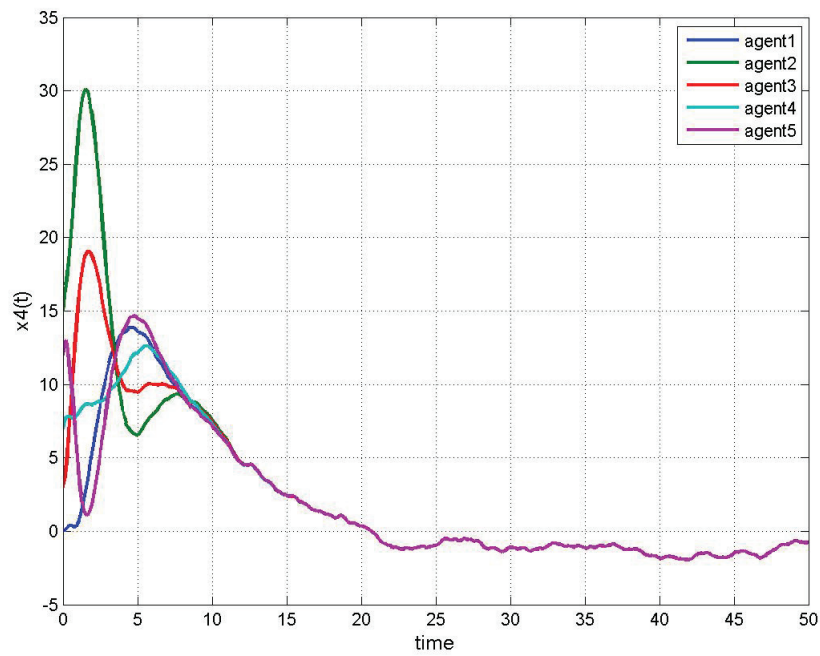


Figure 4.28: Synchronization of state 4 for dynamic output feedback protocol subject to noise and by employing Kalman filter.

### 4.3 Output Feedback Synchronization Subject to Faults

In the following sections, three different methodologies of the output feedback synchronization is proposed. These three approaches are as follows:

- Static output feedback controller
- Dynamic output feedback controller
- Dynamic output feedback observer and controller

### 4.4 Fault-Tolerant Static Output Feedback Synchronization Protocol with Single-input Channel Subject to the LOE Fault

In this section, a static output feedback (SOF) protocol is applied to synchronize marginally-stable linear multi-agent systems with single-input channel, and under a fixed and connected network topology. It will be shown that by applying the proposed static protocol the synchronization can be met even in the presence of fault.

Consider a group of  $N$  homogeneous multi-agent systems as

$$\begin{aligned}\dot{\mathbf{x}}_i(t) &= A\mathbf{x}_i(t) + B\mathbf{u}_i(t) \\ y_i(t) &= C\mathbf{x}_i(t) \quad i = 1, 2, \dots, N\end{aligned}\tag{4.19}$$

where  $A \in \mathbb{R}^{n \times n}$  is a marginally stable matrix,  $B \in \mathbb{R}^{n \times 1}$ ,  $C \in \mathbb{R}^{p \times n}$  and system  $(A, B, C)$  is stabilizable using static output feedback.

To reach an agreement under the network topology  $\mathcal{G}$ , an static output feedback synchronization protocol is applied as

$$\mathbf{u}_i(t) = -K \sum_{j \in \mathcal{N}_i} a_{ij}(y_i(t) - y_j(t)) \quad i = 1, 2, \dots, N\tag{4.20}$$

where  $a_{ii} = 0$  and for  $i \neq j$ ,  $a_{ij} = 1$  if the information flows from agent  $j$  to agent  $i$  and 0 otherwise, and  $K \in \mathbb{R}^{1 \times m}$  is the control gain matrix.

Knowing that agents are homogeneous, the augmented state-space representation of the system described in equation (4.19) can be written as

$$\dot{\mathbf{X}}(t) = (I_N \otimes A - L \otimes BKC)\mathbf{X}(t)\tag{4.21}$$

where  $\mathbf{X}(t) = [\mathbf{x}_1^T(t), \mathbf{x}_2^T(t), \dots, \mathbf{x}_N^T(t)]^T$  is the augmented state of all agents, and  $L[l_{ij}]$  is the Laplacian matrix of the system with  $l_{ii} = \sum_{j \in \mathcal{N}_i} a_{ij}$ , and  $l_{ij} = -a_{ij}$  for  $i \neq j$ . The eigenvalues of  $L$  ( $\lambda_i$ 's) for  $i = 1, 2, \dots, N$  have positive real parts except for  $\lambda_1$ , which is zero.

Using the results from [34], and knowing that the network topology  $\mathcal{G}$  has a spanning tree, by applying protocol (4.20) the system (4.19) synchronizes when

$$A - \lambda_i BKC < 0 \quad i = 2, \dots, N \quad (4.22)$$

This is a static output feedback problem that has a non-convex solution set. Therefore, it is a non-trivial computational and analytical task to find its solutions. One possible approach is to apply iterative LMI, which is a numerical methodology to find a solution for  $K$ . This requires high amount of computations. As opposed to this approach, another method is to choose  $K$  as a solution of  $KC = B^T P$  whose existence is guaranteed if [34]

$$\text{Rank}(C) = \text{Rank} \begin{bmatrix} C \\ B^T P \end{bmatrix} \quad (4.23)$$

where  $P$  is one possible solution of (4.24),

$$A^T P + PA \leq 0 \quad (4.24)$$

It will be shown how the synchronization is affected when there is a change in the Laplacian matrix. For the team of agents described by (4.19), under the network topology  $\mathcal{G}$ , and by designing the matrix  $K$  to satisfy  $KC = B^T P$  the following observations can be made. Specifically, from (4.22), one knows that

$$\begin{aligned} & ((A - (\sigma_i + jw_i)BKC)^H P + P(A - (\sigma_i + jw_i)BKC)) = \\ & ((A - (\sigma_i + jw_i)BB^T P)^H P + P(A - (\sigma_i + jw_i)BB^T P)) = \\ & ((A - (\sigma_i - jw_i)BB^T P)^T P + P(A - (\sigma_i + jw_i)BB^T P)) = \\ & (A^T P + PA) - 2\sigma_i((PB)(PB)^T) < 0 \end{aligned} \quad (4.25)$$

Knowing that the underlying graph is connected,  $\sigma_i = \text{Re}\{\lambda_i\}$  will be positive for all  $i = 2, \dots, N$ , and consequently the second term on the right hand side of (4.25) will be negative definite. Therefore, from (4.24), one may easily conclude even if there is a change in Laplacian matrix, regardless of the  $\sigma_i$  value (as long as it is a positive real value, which means as long as the graph is connected), (4.25) holds for all  $i = 2, \dots, N$ .

Let us now assume that there is a LOE fault in the actuator of agent 1. The effect of this fault can be modeled as

$$\mathbf{u}_1^f(t) = -K \sum_{j \in \mathcal{N}_i} \gamma_1 a_{1j} (y_1(t) - y_j(t)) \quad (4.26)$$

where  $\gamma_1$  represents the actual effectiveness factor of the faulty actuator.

From (4.26) it follows that the change in the effectiveness factor of the actuator in agent 1 can be modeled as a multiplication of associated row of the Laplacian matrix by  $\gamma_1$ . Therefore, the augmented state-space representation of the faulty system becomes

$$\dot{\mathbf{X}}(t) = (I_N \otimes A - L' \otimes BK'C) \mathbf{X}(t) \quad (4.27)$$

where all rows and columns of  $L'$  are the same as the ones in  $L$  in equation (4.21) except for the row associated with the faulty agent  $k$ , which is now changed to

$$l'_{1j} = \gamma_1 l_{1j} \quad j = 1, \dots, N \quad (4.28)$$

$L' = [l'_{ij}]$  is still a zero row sum matrix, with  $l'_{ii} = \sum_{j \in \mathcal{N}_i} a'_{ij}$ , and  $l'_{ij} = -a'_{ij}$  for  $i \neq j$ . Therefore, the graph associated with  $L'$ , which is interpreted as the Laplacian matrix after fault occurrence, has a spanning tree. Using the results from Lemma 2.1,  $L'$  has a single zero eigenvalue and the rest of its eigenvalues,  $\lambda'_i = \sigma'_i + j\omega'_i$  ( $i = 2, \dots, N$ ), have positive real parts.

**Remark 4.3.** *The first row of the Laplacian matrix is changed according to equation (4.28) to model the LOE fault in agent 1.*

**Lemma 4.3.** *The loss of effectiveness (LOE) fault in a team of linear multi-agent systems in (4.19), under the interaction network  $\mathcal{G}$ , and by using the protocol (4.26) does not change synchronization achievement, if*

1. *There exist a matrix  $P = P^T > 0$  such that it satisfies  $A^T P + PA \leq 0$ .*
2. *There exist a control gain matrix  $K$  as solution of  $KC = B^T P$ .*
3. *For those values of  $y \in \mathbb{R}^n$  that makes  $y^T(A^T P + PA)y = 0$ ,  $y^T(PB(PB)^T)y \neq 0$ .*

*Proof.* By applying the results obtained for the synchronization in healthy case, the system described by equation (4.27) (where  $L'$  reflects the adjusted Laplacian to model the fault) will synchronize if  $A - \lambda'_i BK'C$  is Hurwitz for all  $\lambda'_i$  ( $i = 2, \dots, N$ ) where  $\lambda'_i$ 's are the eigenvalues of  $L'$

arranged in an ascending order with  $\lambda'_1 = 0$ , and the rest of them have positive real parts. However, using the observations described at the beginning of this section, if the control gain  $K$  of the healthy system is the solution of  $KC = B^T P$  where  $P$  satisfies (4.24), it is guaranteed that regardless of the effectiveness of fault and consequently the value of  $\lambda'_i$  the faulty system represented by augmented states' dynamics as (4.27) will still synchronize.  $\square$

In the case of multi-input channel similar to single input scenario, it is assumed that the fault occurs in agent 1. Therefore, the state-space equation of the agents becomes

$$\begin{aligned}\dot{\mathbf{x}}_1(t) &= A\mathbf{x}_1(t) + B\mathbf{u}_1^f \\ \dot{\mathbf{x}}_i(t) &= A\mathbf{x}_i(t) + B\mathbf{u}_i\end{aligned}\quad (4.29)$$

From (2.23) and (2.25) one knows when there is a LOE fault in the actuator of agent 1, by employing the static protocol of (4.20) the effect of this fault can be modeled as

$$\begin{aligned}\dot{\mathbf{x}}_i(t) &= A\mathbf{x}_i(t) - BK \sum_{j \in \mathcal{N}_i} a_{ij}(y_i(t) - y_j(t)) \\ \dot{\mathbf{x}}_1(t) &= A\mathbf{x}_1(t) - BK \sum_{j \in \mathcal{N}_1} a_{1j}(y_1(t) - y_j(t)) - B(\Gamma - I)K \sum_{j \in \mathcal{N}_1} a_{1j}(y_1(t) - y_j(t))\end{aligned}\quad (4.30)$$

where  $K \in \mathbb{R}^{m \times n}$  is the control gain matrix.

By applying a change of variables as  $\sigma_i = \mathbf{x}_1 - \mathbf{x}_i$ ,  $i = 2, 3, \dots, N$ , state-space representation of  $\sigma_i$  becomes

$$\begin{aligned}\dot{\sigma}_i(t) &= A\mathbf{x}_1(t) - BK \sum_{j \in \mathcal{N}_1} a_{1j}(y_1(t) - y_j(t)) - B(\Gamma - I)K \sum_{j \in \mathcal{N}_1} a_{1j}(y_1(t) - y_j(t)) \\ &\quad - A\mathbf{x}_i(t) + BK \sum_{j \in \mathcal{N}_i} a_{ij}(y_i(t) - y_j(t)) = \\ &= A\mathbf{x}_1(t) - BKC \sum_{j \in \mathcal{N}_1} a_{1j}(\mathbf{x}_1(t) - \mathbf{x}_j(t)) - B(\Gamma - I)KC \sum_{j \in \mathcal{N}_1} a_{1j}(\mathbf{x}_1(t) - \mathbf{x}_j(t)) \\ &\quad - A\mathbf{x}_i(t) + BKC \sum_{j \in \mathcal{N}_i} a_{ij}(\mathbf{x}_i(t) - \mathbf{x}_j(t)) = \\ &= A\sigma_i(t) - BKC \left( \sum_{j \in \mathcal{N}_1} a_{1j}\sigma_j(t) - \sum_{j \in \mathcal{N}_i} a_{ij}(\sigma_j(t) - \sigma_i(t)) \right) - B(\Gamma - I)KC \sum_{j \in \mathcal{N}_1} a_{1j}\sigma_j(t)\end{aligned}\quad (4.31)$$

Therefore, the augmented state-space representation is as follows:

$$\dot{\sigma}(t) = (I_{N-1} \otimes A - (L_r + 1.\alpha^T) \otimes BKC)\sigma(t) - (1.\alpha^T) \otimes B(\Gamma - I)KC\sigma(t) \quad (4.32)$$

where the variables are defined as in (3.7).

**Remark 4.4.** *It is worth mentioning that the appearance of  $\mathbf{x}_i(t)$  in (4.31) does not mean that it is been used by control protocol. It has been appeared by rewriting  $y_i(t)$  as  $C\mathbf{x}_i(t)$ , which is only for the analysis purpose.*

When there is no fault ( $\Gamma = I$ ), the second term of the augmented state-space representation of (4.32) will be zero and therefore the problem will reduce to (4.21). When there is a fault even if  $\Gamma$  is available the second term will not disappear. In such a case, it is not straight forward to design a control gain  $K$ . Furthermore, even under healthy scenario it is not easy to come up with a control gain  $K$  that satisfies (4.22) either by solving (4.23) or using the iterative LMI approach.

Therefore, in order to deal with the effect of fault and also a more straight-forward approach to design a controller under healthy scenario, dynamic output feedback protocols are proposed in the following section.

## 4.5 Observer-based Dynamic Output Feedback Synchronization Protocol Subject to the LOE/Float Fault

In this section, a dynamic output feedback protocol that also employs an observer is applied to synchronize marginally-stable multi-agent systems with multi-input channel and under a fixed and connected network topology. It will be shown when there is an actuator fault by applying the proposed protocol and by re-designing only the control gain of the faulty agent, the synchronization can still be achieved.

Consider a team of  $N$  homogeneous multi-agent systems as

$$\begin{aligned}\dot{\mathbf{x}}_i(t) &= A\mathbf{x}_i(t) + B\mathbf{u}_i(t) \\ y_i(t) &= C\mathbf{x}_i(t) \quad i = 1, 2, \dots, N\end{aligned}\tag{4.33}$$

where  $A \in \mathbb{R}^{n \times n}$  is a marginally stable matrix,  $B \in \mathbb{R}^{n \times m}$ ,  $C \in \mathbb{R}^{p \times n}$ ,  $\mathbf{x}_i \in \mathbb{R}^n$ ,  $\mathbf{u}_i \in \mathbb{R}^{m \times 1}$ , and the system  $(A, B, C)$  is stabilizable and detectable.

Let us assume that a fault occurs in the first actuator of agent 1, and therefore the control input matrix becomes

$$\begin{aligned}B_1 &= B\Gamma_1, \quad \Gamma_1 = \text{diag}(\gamma_1^1, 1, \dots, 1) \\ B_2 &= B_3 = \dots = B_N = B\end{aligned}\tag{4.34}$$

where  $0 < \gamma_1^1 < 1$  indicates the effectiveness factor of the faulty actuator.

When a LOE/float fault occurs in a multi-input system, the control input matrices of agents are not identical any longer. Therefore, we are dealing with heterogeneous agents. The following methodology is adopted from the literature [36] with required modifications. In that paper, they consider a group of homogeneous agents. However, the problem in this section deals with after fault occurrence that the agents are not identical anymore. Particularly, there is a group of heterogeneous agents that try to synchronize.

**Lemma 4.4.** *The dynamic protocol*

$$\begin{aligned}\dot{\zeta}_i(t) &= (A + B'_i K'_i) \zeta_i(t) + \sum_{j \in \mathcal{N}_i} a_{ij} (\zeta_j(t) - \zeta_i(t) + \hat{\mathbf{x}}_i(t) - \hat{\mathbf{x}}_j(t)) \\ \dot{\hat{\mathbf{x}}}_i &= A \hat{\mathbf{x}}_i + B'_i \mathbf{u}_i + H(\hat{y}_i - y_i) \\ \mathbf{u}_i(t) &= K'_i \zeta_i(t) \quad i = 1, \dots, N\end{aligned}\tag{4.35}$$

will synchronize the system (4.33) when the exact fault information is available.

where  $\zeta_i \in \mathbb{R}^n$ ,  $B'_i \in \mathbb{R}^{n \times m}$  and  $K'_i \in \mathbb{R}^{m \times n}$  are respectively the controller state, estimated control input matrix, and redesigned control gain associated with the agent  $i$ . Moreover, since the exact fault estimation information is available, we will have  $B'_1 = B\Gamma_1$ ,  $B'_i = B_i = B$  for  $i = 2, \dots, N$ , and  $K'_i = K_i = K$  for  $i = 2, \dots, N$ .

*Proof.* Defining the new state variables as  $s_i(t) = \hat{\mathbf{x}}_i(t) - \zeta_i(t)$  and  $e_i(t) = \mathbf{x}_i(t) - \hat{\mathbf{x}}_i(t)$ , the augmented state-space system is as follows:

$$\dot{\zeta}_i(t) = (\tilde{A}_N + B_d K_d) \zeta_i(t) + L s(t) \tag{4.36a}$$

$$\dot{s}(t) = \tilde{A}_N s(t) - L s(t) + \tilde{H} \tilde{C} e(t) \tag{4.36b}$$

$$\dot{e}(t) = (\tilde{A}_N + \tilde{H} \tilde{C}) e(t) \tag{4.36c}$$

Equation (4.36) could be rewritten as

$$\begin{bmatrix} \dot{\zeta}(t) \\ \dot{s}(t) \\ \dot{e}(t) \end{bmatrix} = \begin{bmatrix} (\tilde{A}_N + B_d K_d) & L & 0 \\ 0 & \tilde{A}_N - L & \tilde{H}\tilde{C} \\ 0 & 0 & (\tilde{A}_N + \tilde{H}\tilde{C}) \end{bmatrix} \begin{bmatrix} \zeta(t) \\ s(t) \\ e(t) \end{bmatrix} \quad (4.37)$$

where  $L = [l_{ij}]$  is the Laplacian matrix of the system with  $l_{ii} = \sum_{j \in \mathcal{N}_i} a_{ij}$  and  $l_{ij} = -a_{ij}$  for  $i \neq j$ ,  $\tilde{A}_N = I_N \otimes A$ ,  $\tilde{C}_N = I_N \otimes C$ ,  $\tilde{H}_N = I_N \otimes H$ ,  $B_d = \text{block diag}(B'_1, B, \dots, B)$ ,  $K_d = \text{block diag}(K'_1, K, \dots, K)$ ,  $\zeta = [\zeta_1^T, \zeta_2^T, \dots, \zeta_N^T]^T$ ,  $s = [s_1^T, s_2^T, \dots, s_N^T]^T$ ,  $e = [e_1^T, e_2^T, \dots, e_N^T]^T$ , and  $B'_1 \in \mathbb{R}^{n \times m}$  and  $K'_1$  are the post-fault control input and control gain matrices of the faulty agent 1, respectively.

From equation (4.36), it can be seen that neither the observer error dynamics (4.36c) nor the synchronization dynamics (4.36b) depend on the state dynamics or the augmented control input matrix  $B_d$ . Therefore, knowing that the pair  $(A, C)$  is observable, there exists a matrix  $H$  which will still make the equation (4.36c) to stabilize. Moreover, using the result of Lemma 2.2 it is guaranteed that  $s_i(t)$  in equation (4.36b) will asymptotically synchronize and the synchronization trajectory will be governed by  $\dot{s}_c(t) = A s_c(t)$ . This is concluded by using the fact that Laplacian is a zero row sum matrix.

Hence, when the synchronization trajectory (4.36b) reaches its steady state and therefore  $Ls(t)$  reaches zero, the augmented dynamics of the controller (4.36a) reduces to

$$\dot{\zeta} = (\tilde{A}_N + B_d K_d) \zeta \quad (4.38)$$

This implies when the control gain  $K_d$  is designed to make  $\tilde{A}_N + B_d K_d$  Hurwitz, the dynamics of (4.38) will be exponentially stable and will asymptotically reach zero.

Finally,  $\mathbf{x}_i(t)$ , which is obtained by  $\mathbf{x}_i(t) = e_i(t) + s_i(t) + \zeta_i(t)$ , converges to  $s_i(t)$ . Therefore, one can conclude that the state of all agents will reach to a common solution of  $\dot{s}_c(t) = A s_c(t)$ .  $\square$

**Remark 4.5.** *It is worth noting that the LOE fault does not change the stabilizability of the faulty agent. Therefore, one is still able to design  $K'_1$  to make the faulty agent, and consequently the equation (4.38) stable.  $K'_1$  can be designed using any conventional control approach.*

**Remark 4.6.** *It should be noted that the fault changes neither the synchronization nor the observer*

dynamics.

**Remark 4.7.** *It is worth mentioning that after the fault occurrence, only the control gain of the faulty agent  $K'_1$  is redesigned to still keep the matrix  $A + B'_1 K'_1$  Hurwitz. The control gains for the rest of the agents as well as the observer gain remain the same as fault-free case.*

### 4.5.1 Simulation Results

In this section, the synchronization of autonomous underwater vehicles (AUVs) in the longitudinal plane by having partial state information, using the measurement matrix in (2.13), is investigated. For the sake of simulation, the multi-input model (3.53) is considered.

For simulations purposes, the control gain  $K$  and the observer gain  $H$  of equation (4.35) are designed according to (4.39).

$$\begin{aligned} K &= \begin{bmatrix} 0.3320 & -0.5650 & 1.0000 & -3.7872 \\ 0.7908 & 0.6427 & 0.0061 & 1.1843 \end{bmatrix} \\ H &= \begin{bmatrix} 3.1046 & 0.2406 & 0.0780 \\ -2.0866 & 0.5573 & -0.3120 \\ 0.4008 & 2.7676 & -2.0000 \\ 1.0000 & 0 & 1.0000 \end{bmatrix} \end{aligned} \quad (4.39)$$

The results for the passive simulation scenario of the float and LOE faults occurred at  $t = 2 \text{ Sec}$  are shown in Figures 4.29 - 4.31. It is assumed that there is no fault detection and recovery module and therefore the control gain  $K'_1$  of faulty agent is the same as the one in healthy agents  $K$ .

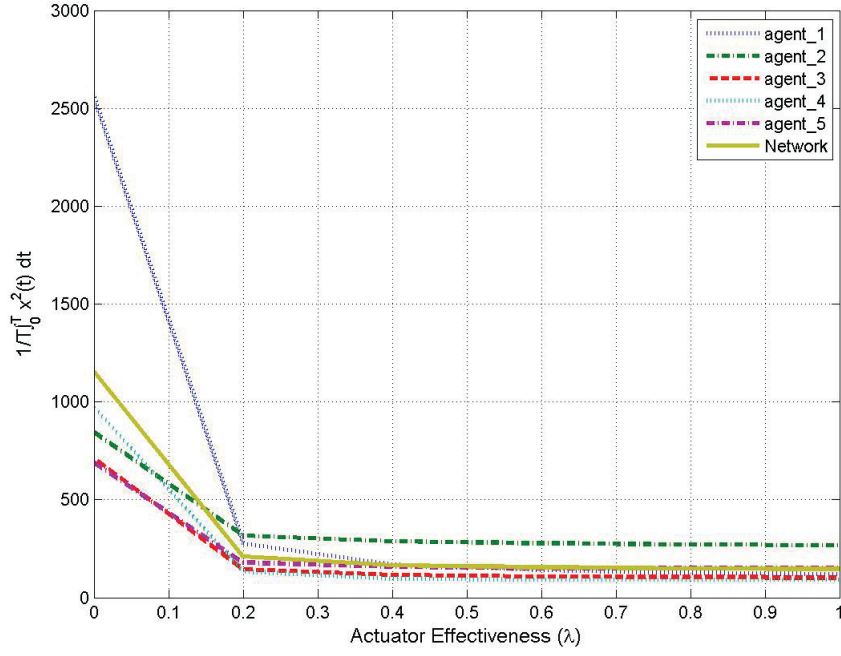


Figure 4.29: State performance index for dynamic output feedback protocol without fault recovery.

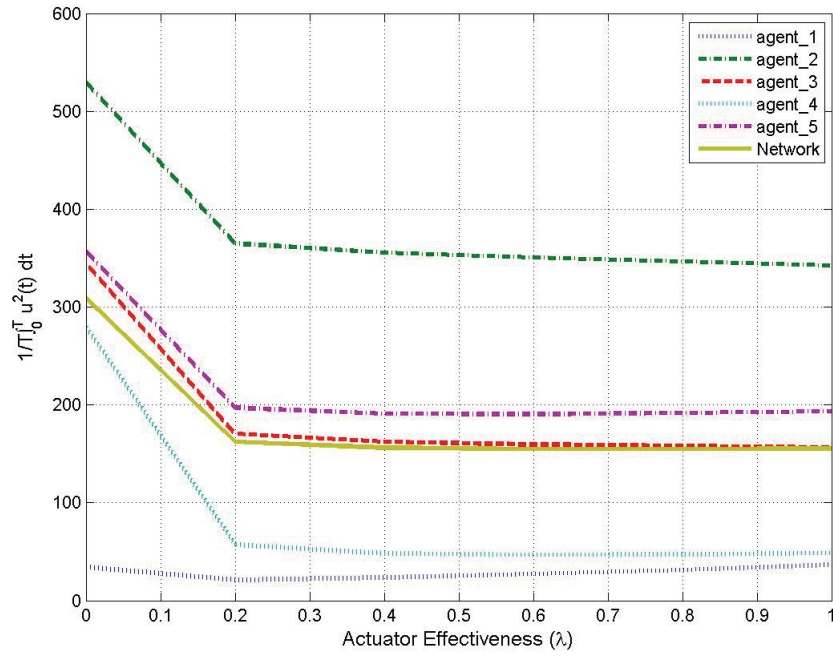


Figure 4.30: Control input performance index for dynamic output feedback protocol without fault recovery.

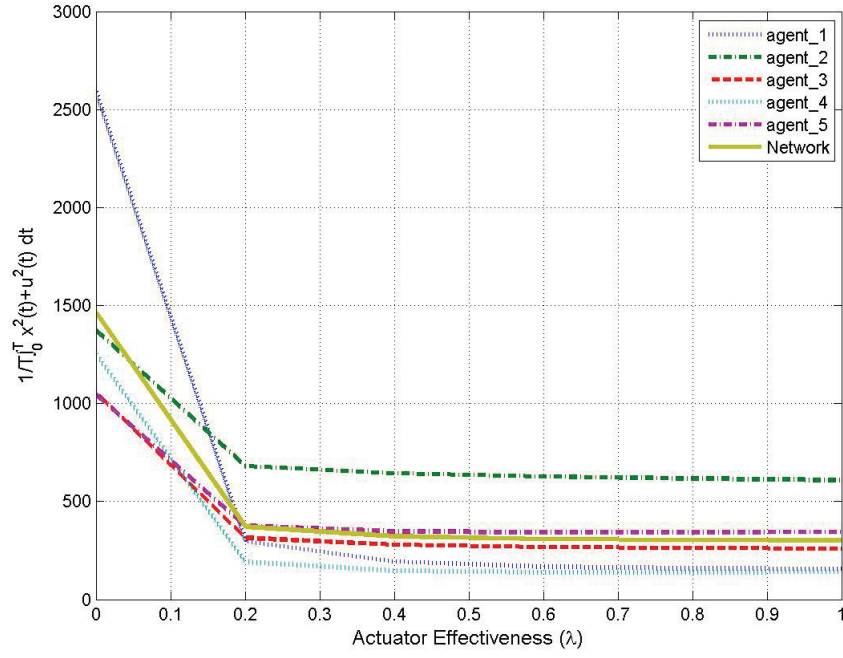


Figure 4.31: Total performance index for dynamic output feedback protocol without fault recovery.

As in Figures 4.29 - 4.31, when severity of fault increases, the performance indices worsen. Therefore, in order to have a better idea of the effect of fault, the detailed results for the worst case scenario (float fault) are shown in Figures 4.32 - 4.45.

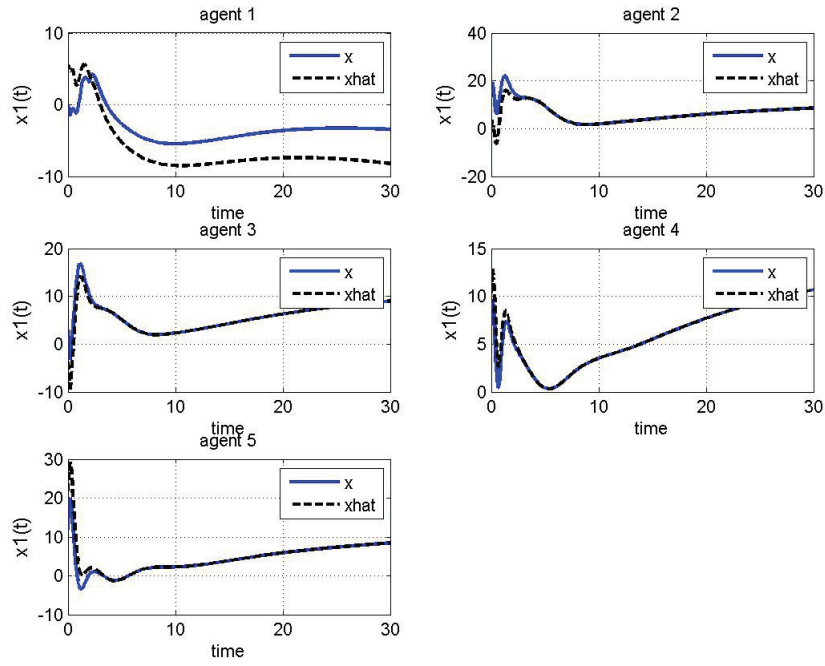


Figure 4.32: State estimation of state 1 for dynamic output feedback protocol without fault recovery (float fault).

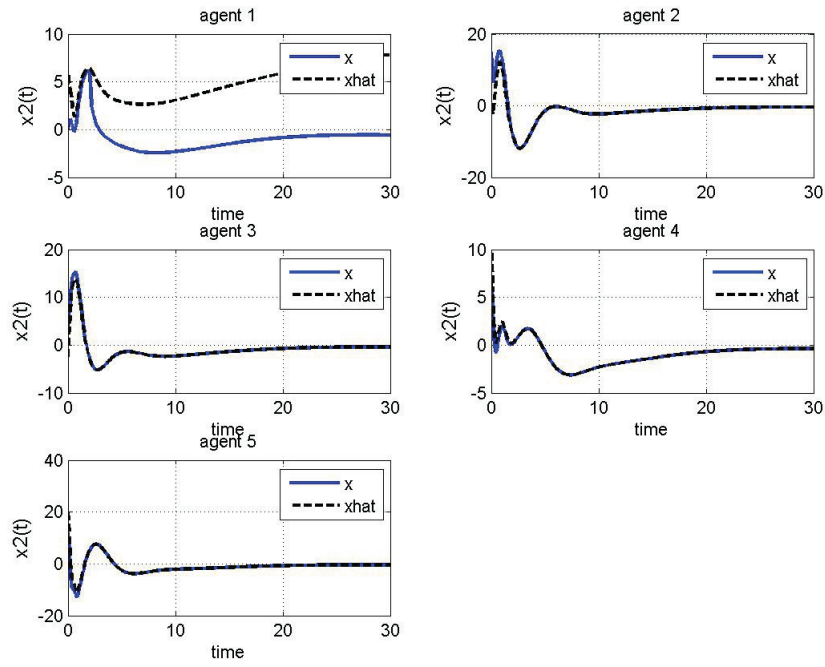


Figure 4.33: State estimation of state 2 for dynamic output feedback protocol without fault recovery (float fault).

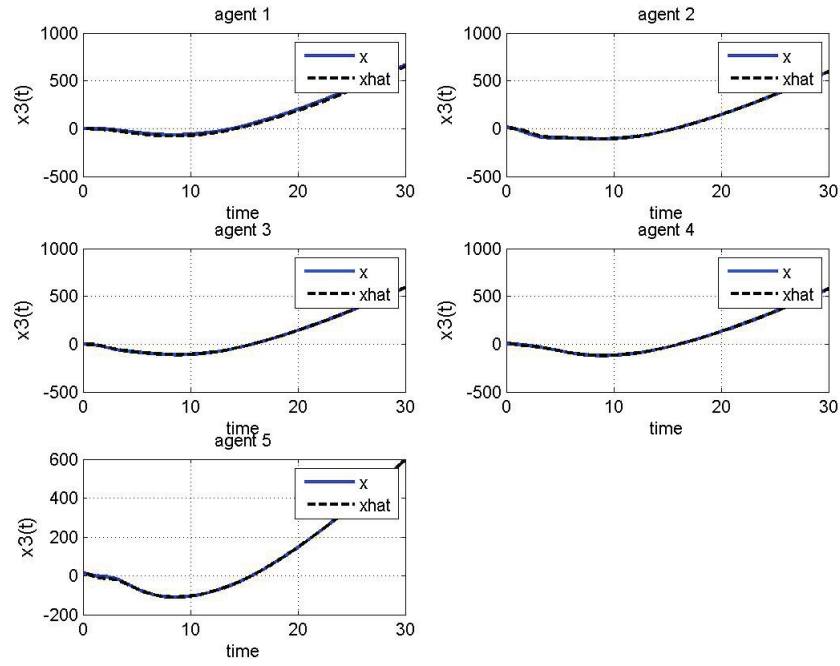


Figure 4.34: State estimation of state 3 for dynamic output feedback protocol without fault recovery (float fault).

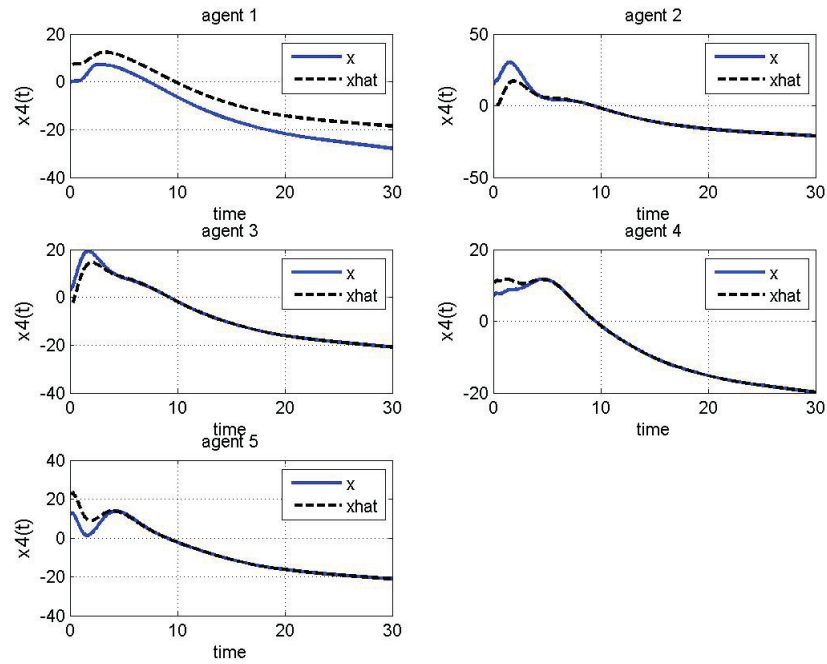


Figure 4.35: State estimation of state 4 for dynamic output feedback protocol without fault recovery (float fault).

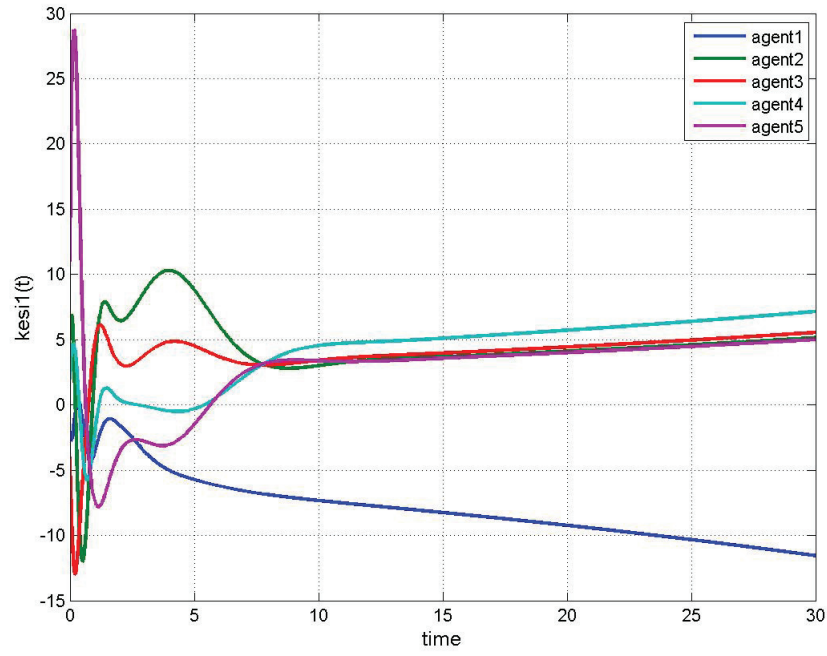


Figure 4.36: Controller state 1 for dynamic output feedback protocol without fault recovery (float fault).

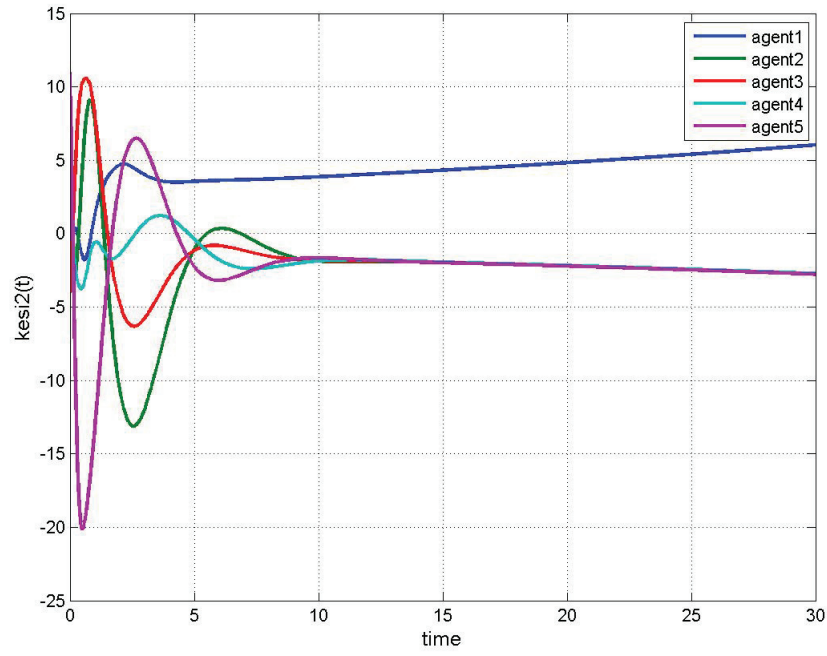


Figure 4.37: Controller state 2 for dynamic output feedback protocol without fault recovery (float fault).

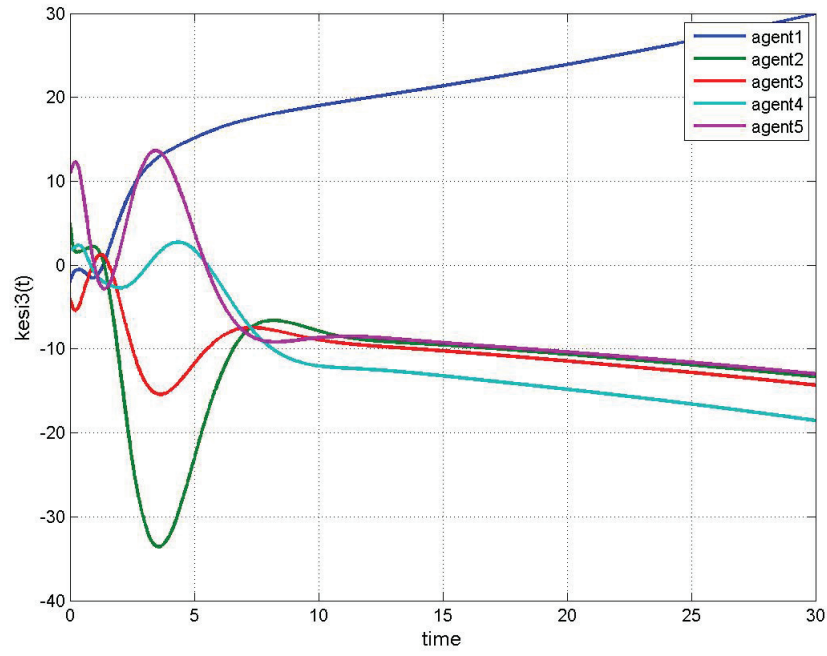


Figure 4.38: Controller state 3 for dynamic output feedback protocol without fault recovery (float fault).

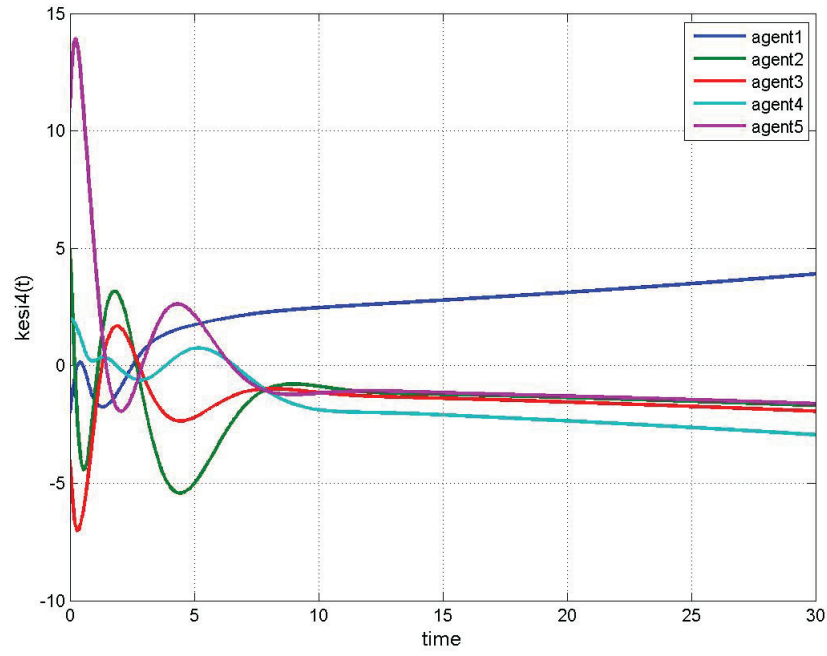


Figure 4.39: Controller state 4 for dynamic output feedback protocol without fault recovery (float fault).

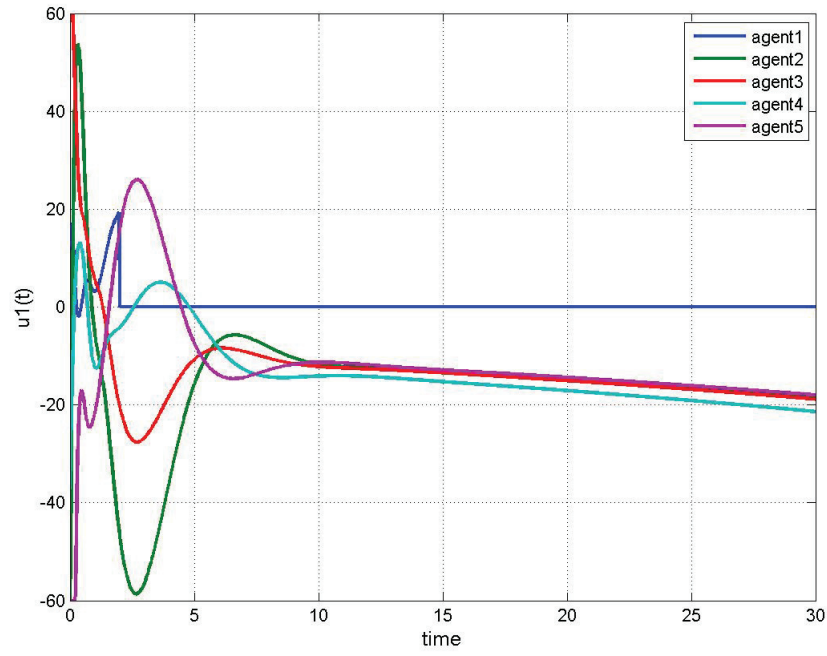


Figure 4.40: Control input signal  $u_1(t)$  for dynamic output feedback protocol without fault recovery (float fault).

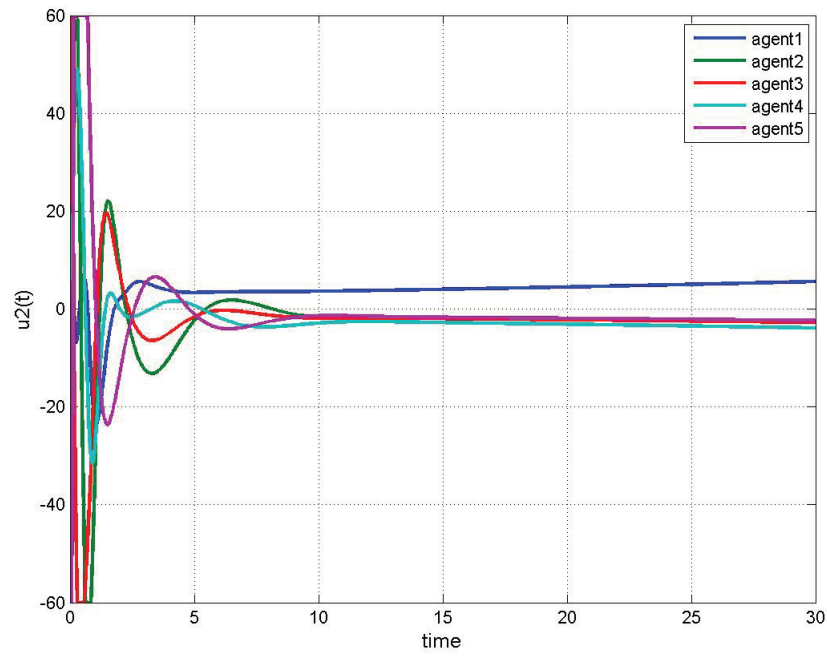


Figure 4.41: Control input signal  $u_2(t)$  for dynamic output feedback protocol without fault recovery (float fault).

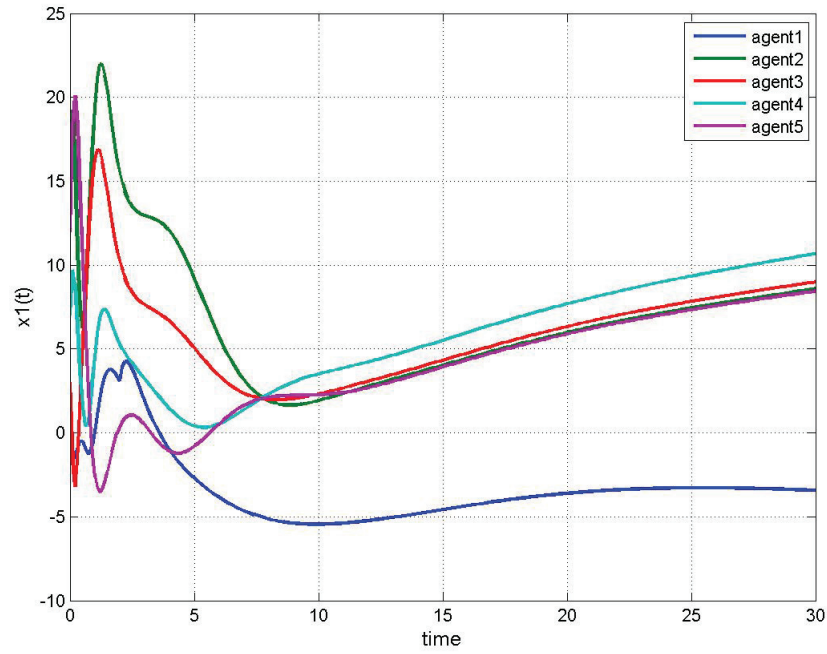


Figure 4.42: Consensus achievement of state 1 for dynamic output feedback protocol without fault recovery (float fault).

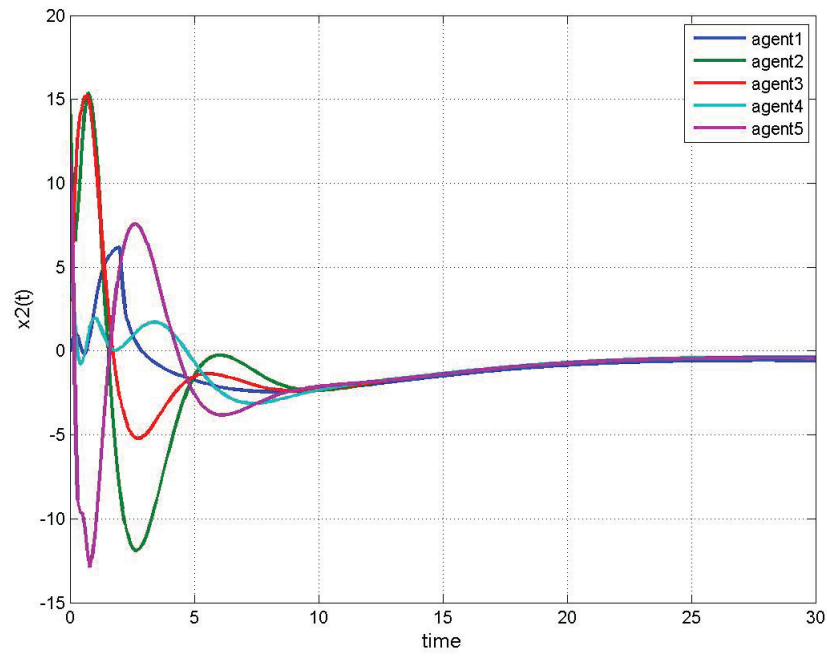


Figure 4.43: Consensus achievement of state 2 for dynamic output feedback protocol without fault recovery (float fault).

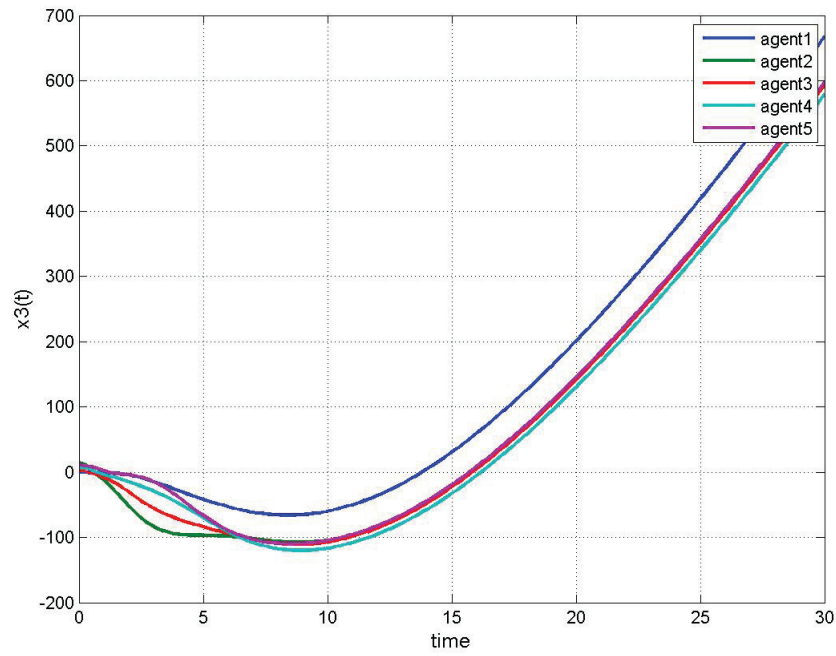


Figure 4.44: Consensus achievement of state 3 for dynamic output feedback protocol without fault recovery (float fault).

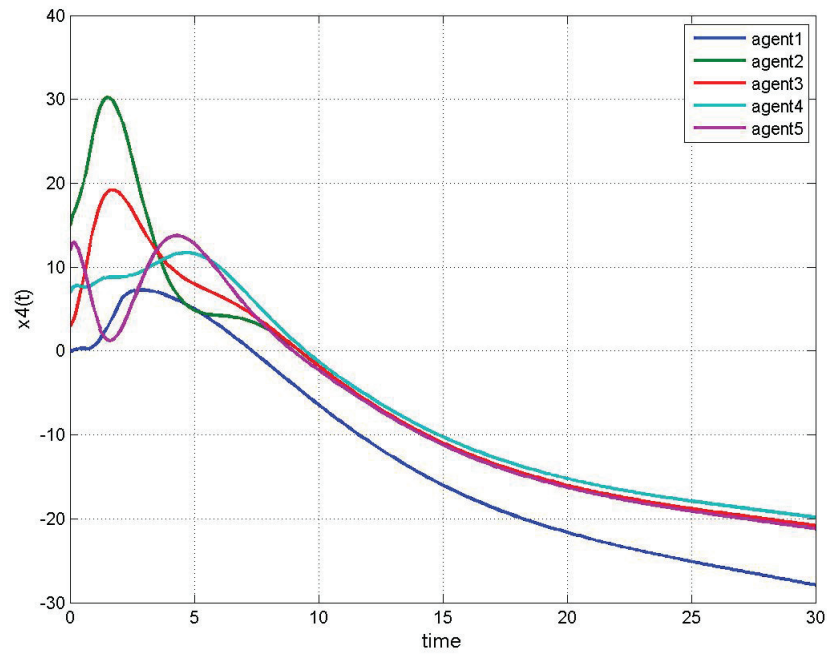


Figure 4.45: Consensus achievement of state 4 for dynamic output feedback protocol without fault recovery (float fault).

As it can be seen in Figures 4.32 - 4.45, when float fault occurs, the synchronization is not achieved any more and therefore the controller gain should be redesigned. The divergence of the state's and controller's states is due to the use of state estimation results. The states estimation diverges as the nominal control input matrix does not hold after fault occurrence.

Assuming that the effectiveness factor of the faulty agent 1 is available, its post-fault control gain is designed as

$$K'_1 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 5.7773 & -5.7613 & -4.8690 & 22.9602 \end{bmatrix} \quad (4.40)$$

The simulation results with fault recovery are presented in Figures 4.46 - 4.59. Simulations results show that by redesigning the control gain after fault occurrence the synchronization could still be achieved.

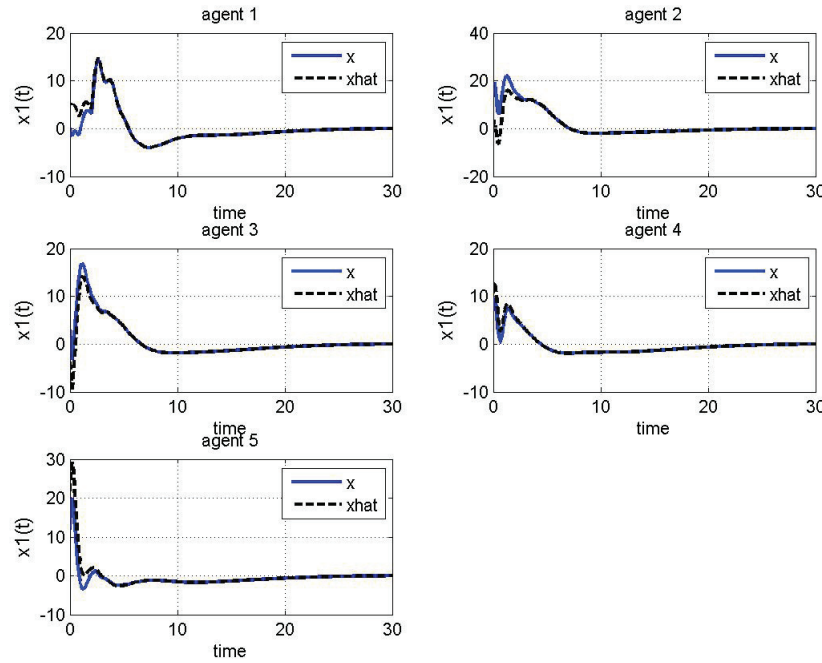


Figure 4.46: State estimation of state 1 for dynamic output feedback protocol with fault recovery (float fault).

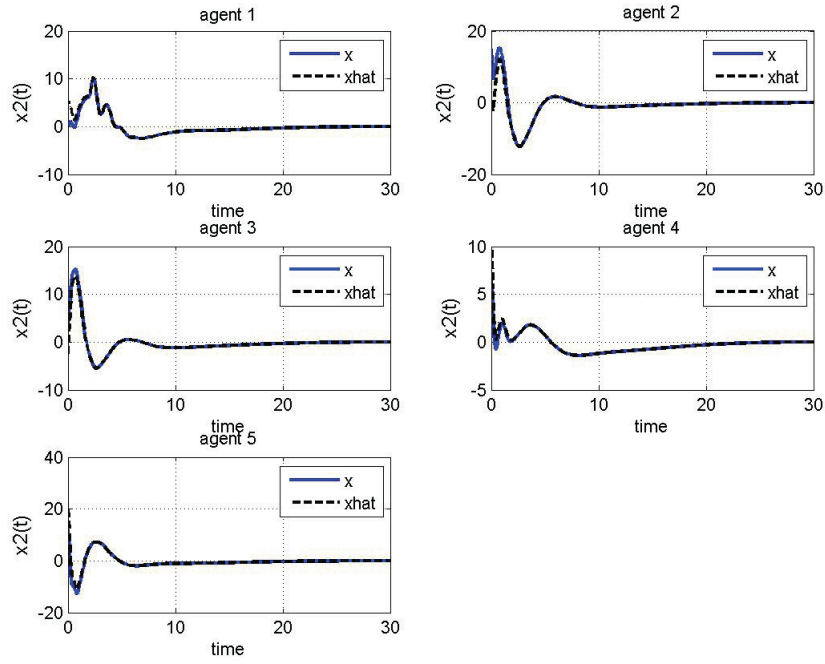


Figure 4.47: State estimation of state 2 for dynamic output feedback protocol with fault recovery (float fault).

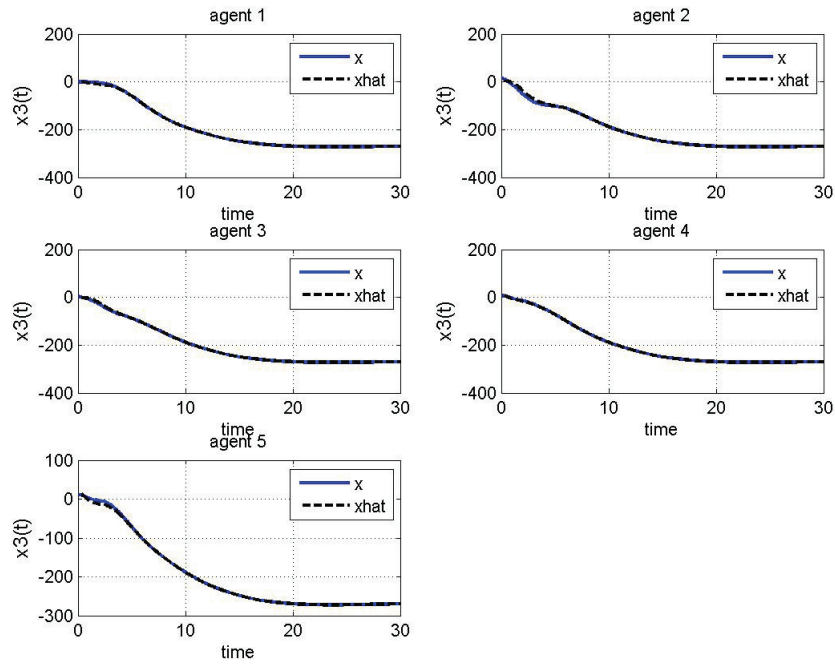


Figure 4.48: State estimation of state 3 for dynamic output feedback protocol with fault recovery (float fault).

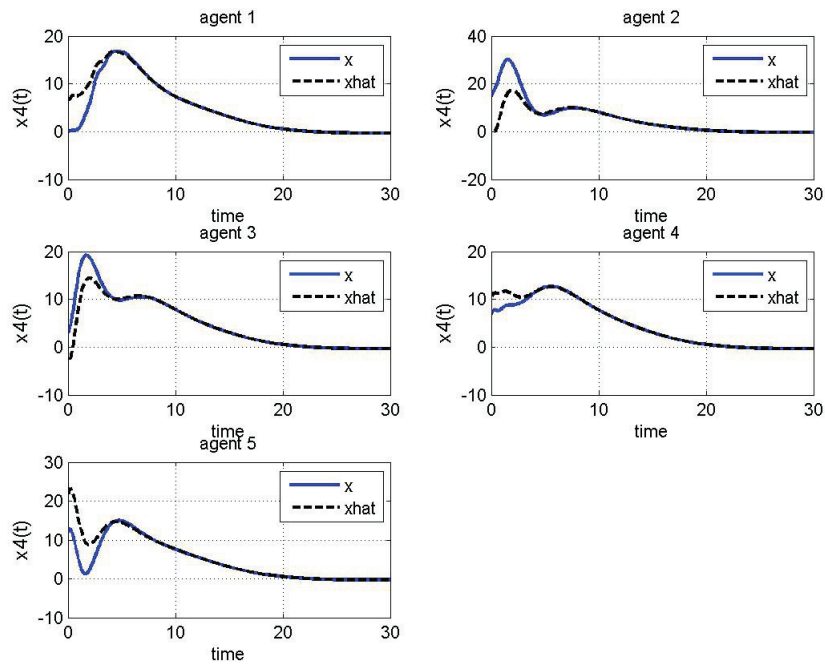


Figure 4.49: State estimation of state 4 for dynamic output feedback protocol with fault recovery (float fault).

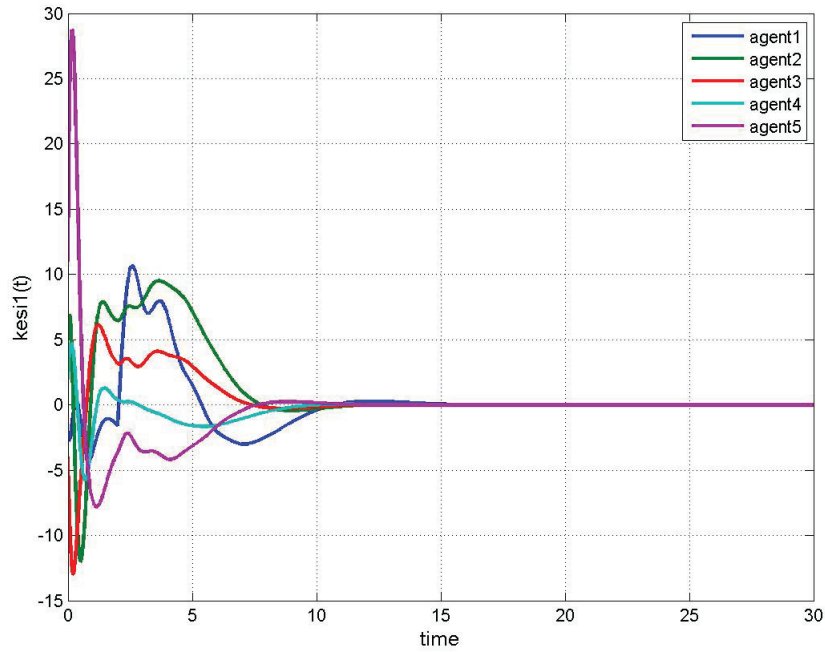


Figure 4.50: Controller state 1 for dynamic output feedback protocol with fault recovery (float fault).

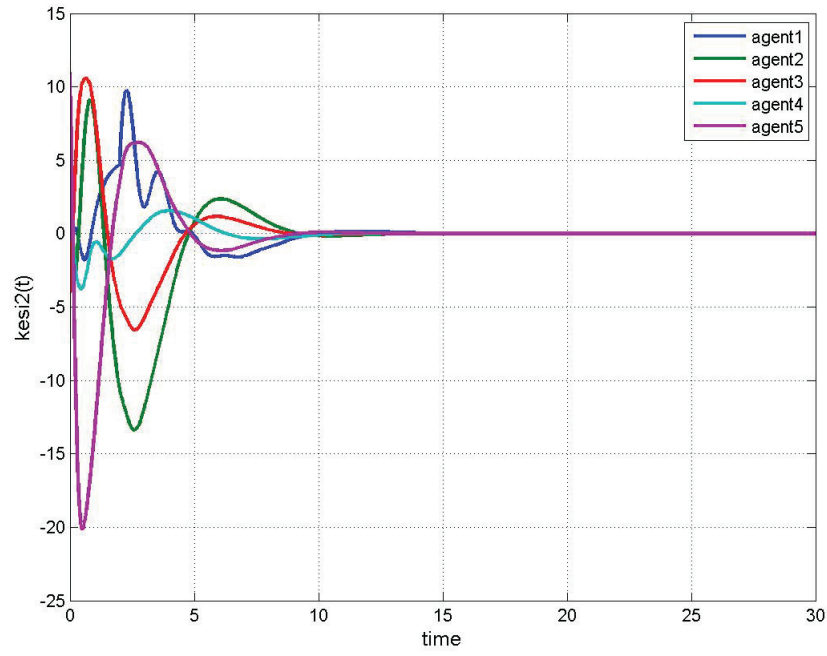


Figure 4.51: Controller state 2 for dynamic output feedback protocol with fault recovery (float fault).

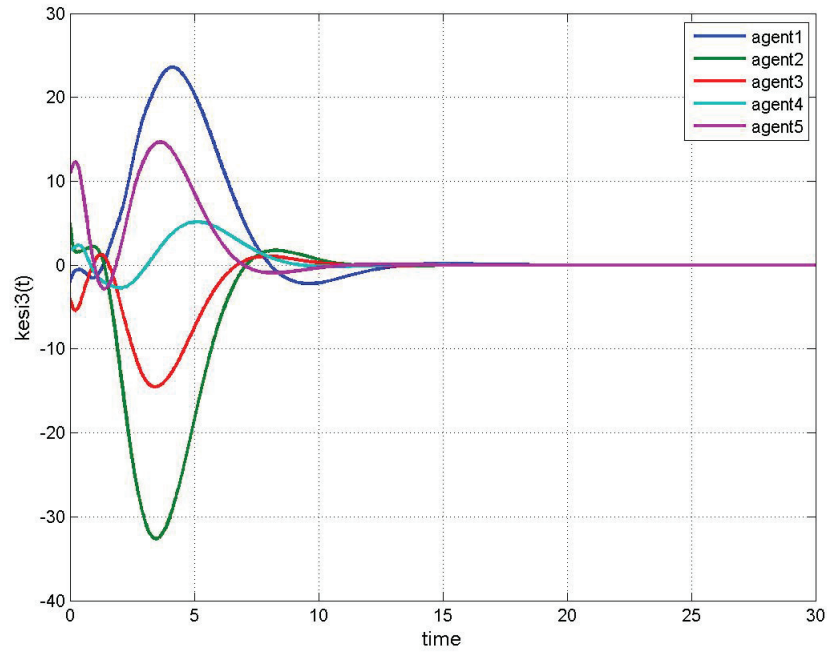


Figure 4.52: Controller state 3 for dynamic output feedback protocol with fault recovery (float fault).

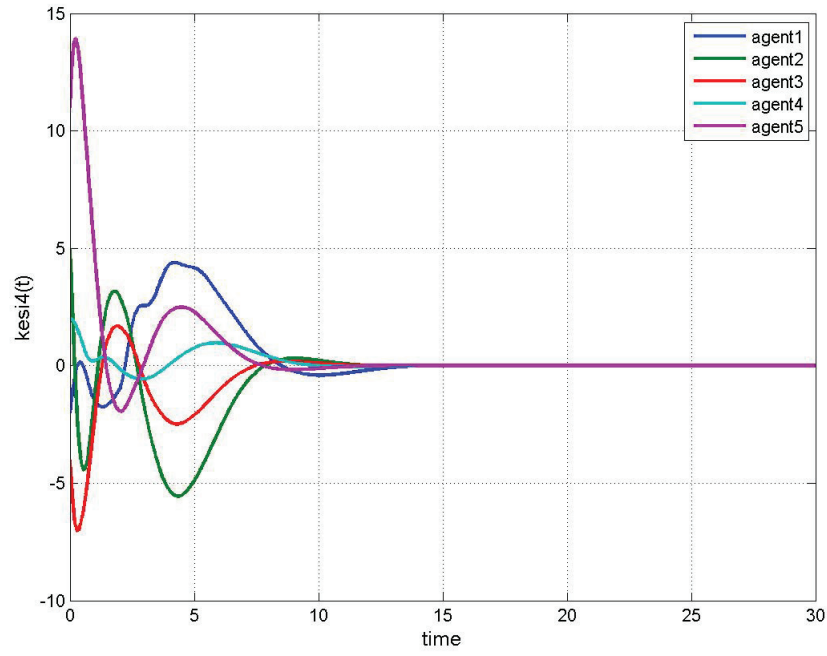


Figure 4.53: Controller state 4 for dynamic output feedback protocol with fault recovery (float fault).

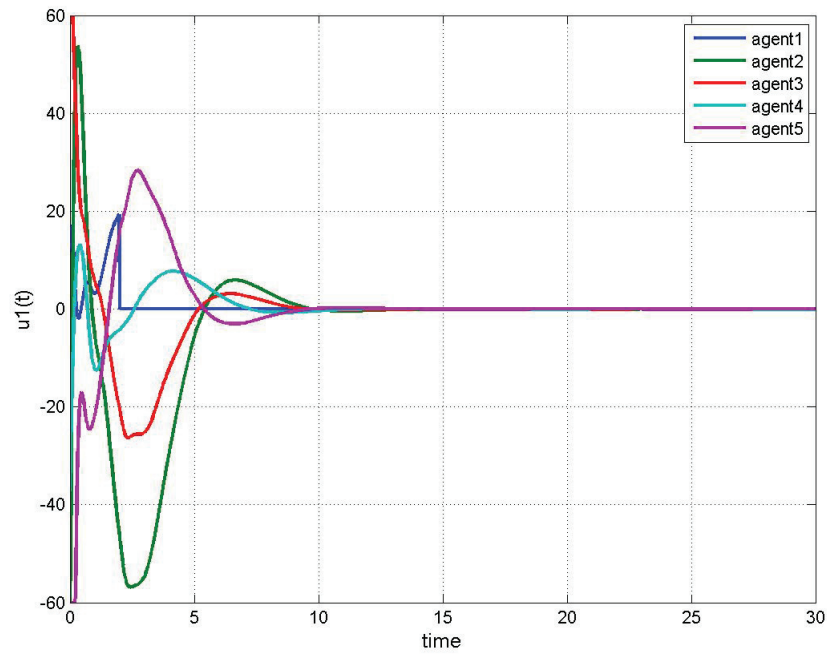


Figure 4.54: Control input signal  $u_1(t)$  for dynamic output feedback protocol with fault recovery (float fault).

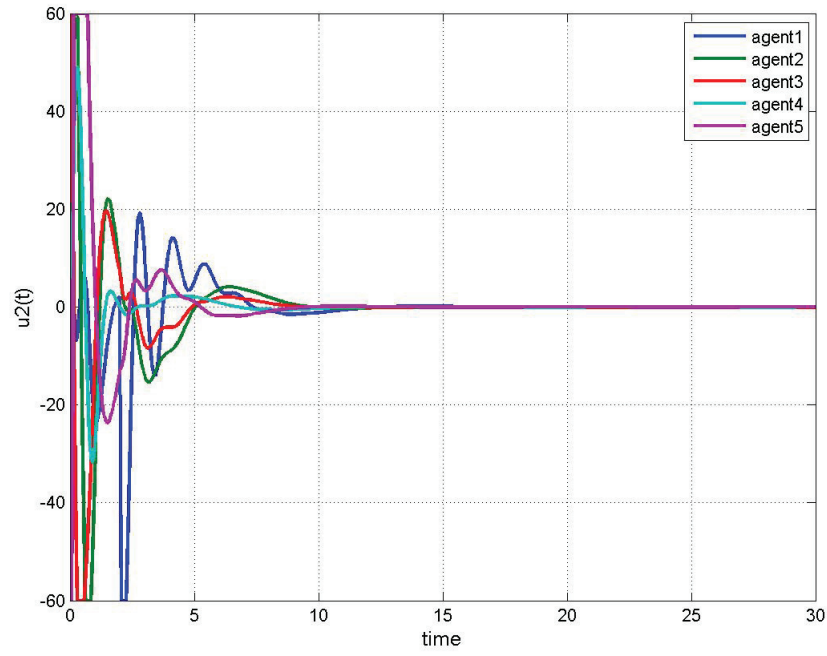


Figure 4.55: Control input signal  $u_2(t)$  for dynamic output feedback protocol with fault recovery (float fault).

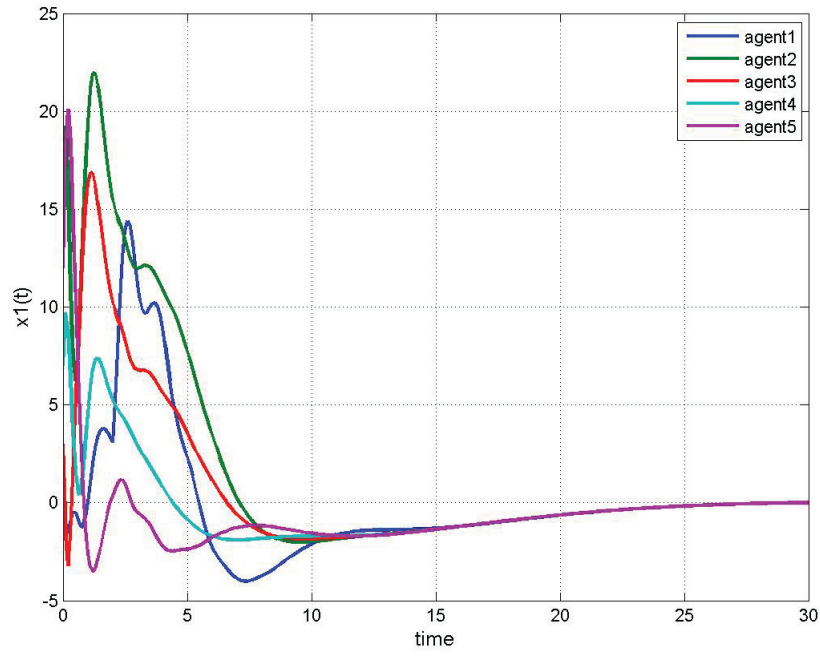


Figure 4.56: Synchronization of state 1 for dynamic output feedback protocol with fault recovery (float fault).

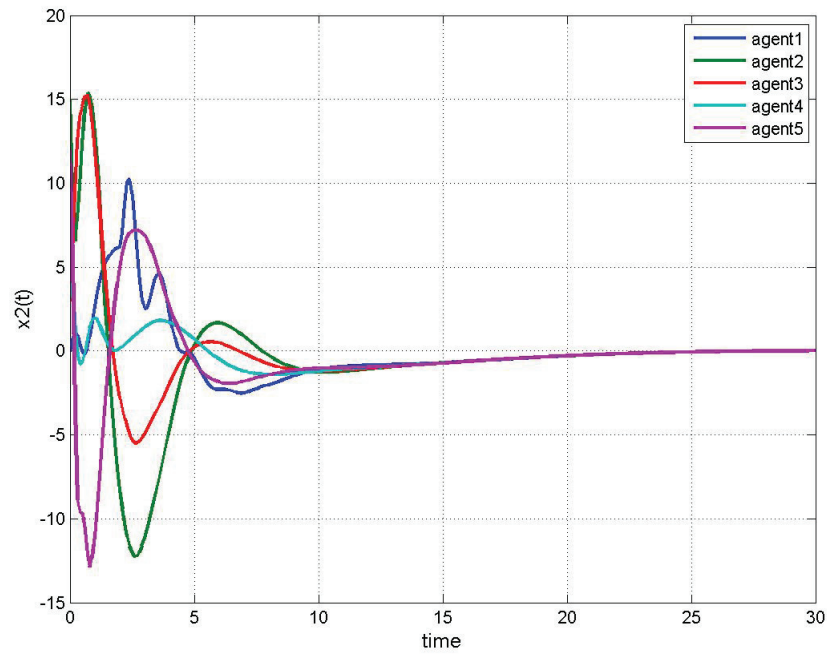


Figure 4.57: Synchronization of state 2 for dynamic output feedback protocol with fault recovery (float fault).

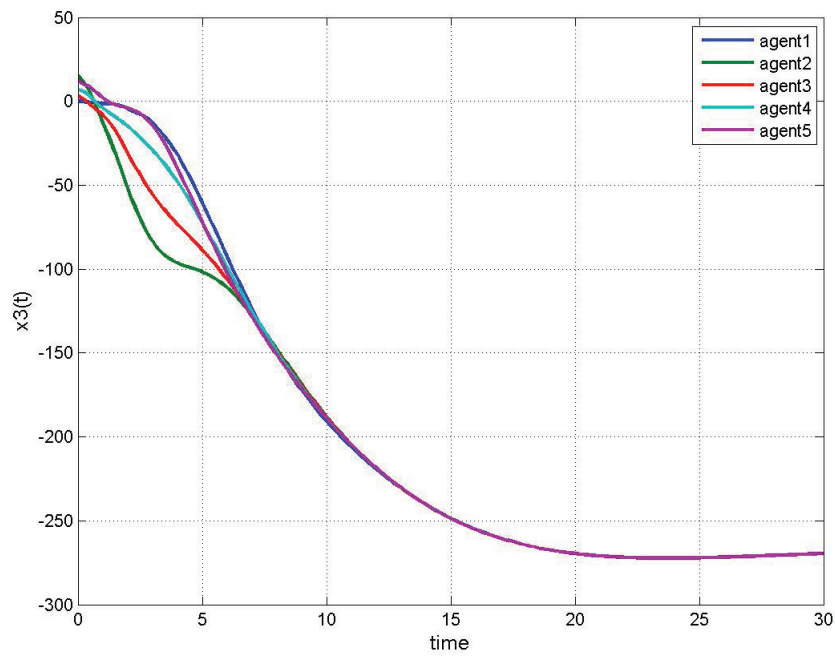


Figure 4.58: Synchronization of state 3 for dynamic output feedback protocol with fault recovery (float fault).

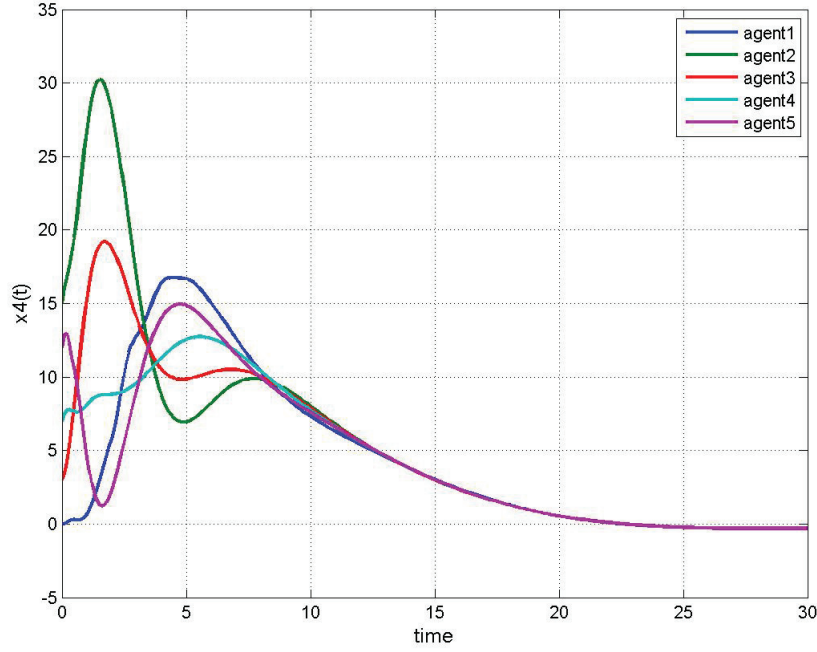


Figure 4.59: Synchronization of state 4 for dynamic output feedback protocol with fault recovery (float fault).

## 4.6 Dynamic Output Feedback Synchronization Protocol Subject to the LOE/Float Fault

In this section, to achieve synchronization in presence of fault, a fully novel control approach is proposed where it only employs a dynamic controller and no observer. It will be shown when there is a LOE fault in the actuator, by applying this protocol, how the synchronization is affected.

**Lemma 4.5.** *The dynamic protocol*

$$\begin{aligned} \dot{\xi}_i(t) &= (A + BK)\xi_i(t) + F \sum_{j \in \mathcal{N}_i} a_{ij}(C(\xi_i(t) - \xi_j(t)) - (y_i(t) - y_j(t))) \\ \mathbf{u}_i(t) &= K\xi_i(t) \end{aligned} \quad (4.41)$$

will synchronize the system (4.29) when the exact fault information is available.

where  $F \in \mathbb{R}^{n \times n}$ , and  $K \in \mathbb{R}^{n \times n}$  are designed such that  $I_{N-1} \otimes A - (L_r + 1_{N-1} \cdot \alpha^T) \otimes FC$  and  $A + BK$  are Hurwitz, and  $\xi_i(t) \in \mathbb{R}^n$  is the controller's state vector of the  $i$ th agent.

*Proof.* The method proposed in this section employs only a dynamic controller. When a LOE fault occurs in agent 1, the state-space representation of the controller changes to

$$\begin{aligned}\dot{\xi}_1(t) &= (A + BK)\xi_1(t) + F \sum_{j \in \mathcal{N}_1} a_{1j}(C(\xi_1(t) - \xi_j(t)) - (y_1(t) - y_j(t))) = \\ & (A + BK)\xi_1(t) + FC \sum_{j \in \mathcal{N}_1} a_{1j}((\xi_1(t) - \xi_j(t)) - (\mathbf{x}_1(t) - \mathbf{x}_j(t))) \\ \mathbf{u}_1^f(t) &= \Gamma K \xi_1(t)\end{aligned}\tag{4.42}$$

where  $\Gamma$  represents the actuator effectiveness.

Let us define the new state variables of  $s_i = \mathbf{x}_i - \xi_i$ , which leads to the state-space representation as follows:

$$\begin{aligned}\dot{s}_1(t) &= A s_1(t) + FC \sum_{j \in \mathcal{N}_1} a_{1j}(s_1(t) - s_j(t)) + B(\Gamma - I)K\xi_1 \\ \dot{s}_i(t) &= A s_i(t) + FC \sum_{j \in \mathcal{N}_i} a_{ij}(s_i(t) - s_j(t))\end{aligned}\tag{4.43}$$

By applying another change of variables as  $\sigma_i = s_1 - s_i$ , we have

$$\begin{aligned}\dot{\sigma}_i(t) &= \dot{s}_1(t) - \dot{s}_i(t) = \\ & A\sigma_i(t) + FC \sum_{j \in \mathcal{N}_1} a_{1j}\sigma_j(t) + FC \sum_{j \in \mathcal{N}_i} a_{ij}(\sigma_j(t) - \sigma_i(t)) + B(\Gamma - I)K\xi_1\end{aligned}\tag{4.44}$$

Using the new state variables, the dynamics of the controller of the faulty agent 1 and the healthy agents becomes

$$\begin{aligned}\dot{\xi}_1(t) &= (A + BK)\xi_1(t) - FC \sum_{j \in \mathcal{N}_1} a_{1j}(\sigma_j(t)) \\ \dot{\xi}_i(t) &= (A + BK)\xi_i(t) - FC \sum_{j \in \mathcal{N}_i} a_{ij}(\sigma_j(t) - \sigma_i(t))\end{aligned}\tag{4.45}$$

Moreover, by defining new state variables for controller's states as  $v_i = \xi_1 - \xi_i$  we have

$$\begin{aligned}
\dot{v}_i(t) &= \dot{\xi}_1(t) - \dot{\xi}_i(t) = (A + BK)(\xi_1(t) - \xi_i(t)) - FC \sum_{j \in \mathcal{N}_1} a_{1j}(s_1(t) - s_j(t)) \\
&+ FC \sum_{j \in \mathcal{N}_i} a_{ij}(s_i(t) - s_j(t)) = \\
&(A + BK)v_i(t) - FC \sum_{j \in \mathcal{N}_1} a_{1j}\sigma_j(t) + FC \sum_{j \in \mathcal{N}_i} a_{ij}(\sigma_j(t) - \sigma_i(t))
\end{aligned} \tag{4.46}$$

Finally, the augmented state-space representation of the entire system becomes as follows:

$$\begin{bmatrix} \dot{\sigma}(t) \\ \dot{v}(t) \\ \dot{\xi}_1(t) \end{bmatrix} = \begin{bmatrix} I_{N-1} \otimes A + (L_r + 1_{N-1} \cdot \alpha^T) \otimes FC & 0 & 1_{N-1} \otimes (\Gamma - I)K \\ -(L_r + 1_{N-1} \cdot \alpha^T) \otimes FC & I_{N-1} \otimes (A + BK) & 0 \\ -\alpha^T \otimes FC & 0 & (A + BK) \end{bmatrix} \begin{bmatrix} \sigma(t) \\ v(t) \\ \xi_1(t) \end{bmatrix} \tag{4.47}$$

As we see, the synchronization of a group of agents under LOE fault is formulated as a stabilization problem.

However, when there is no fault, (4.47) is reduced to a lower triangular matrix as

$$\begin{bmatrix} \dot{\sigma}(t) \\ \dot{v}(t) \\ \dot{\xi}_1(t) \end{bmatrix} = \begin{bmatrix} I_{N-1} \otimes A - (L_r + 1_{N-1} \cdot \alpha^T) \otimes FC & 0 & 0 \\ (L_r + 1_{N-1} \cdot \alpha^T) \otimes FC & I_{N-1} \otimes (A + BK) & 0 \\ \alpha^T \otimes FC & 0 & (A + BK) \end{bmatrix} \begin{bmatrix} \sigma(t) \\ v(t) \\ \xi_1(t) \end{bmatrix} \tag{4.48}$$

Equation (4.48) is a cascade of three subsystems with state variables as  $\sigma$ ,  $v$ , and  $\xi_1$ . The first subsystem is

$$\dot{\sigma}(t) = (I_{N-1} \otimes A - (L_r + 1_{N-1} \cdot \alpha^T) \otimes FC)\sigma(t) \tag{4.49}$$

where  $F$  is designed in a way to make  $\sigma$  asymptotically stable. This equation is analogous to (3.7) and therefore a similar methodology could be applied to make (4.49) stable. Under the fact that  $\sigma(t)$  asymptotically goes to zero, the next two subsystems reduces to

$$\begin{aligned}
\dot{v}(t) &= (I_{N-1} \otimes (A + BK))v(t) \\
\dot{\xi}_1(t) &= (A + BK)\xi_1(t)
\end{aligned} \tag{4.50}$$

When  $K$  is designed such that  $A + BK$  is Hurwitz, both subsystems in (4.50) will be asymptotically

stable. Finally, using the asymptotic stability results for  $\xi_1$ ,  $\nu_i = \xi_1 - \xi_i$  and  $\sigma_i = s_1 - s_i$ , one can easily show that  $\xi_1 = \xi_i = \xi_j = 0$  and  $s_1 = s_i = s_j$  for all  $i, j \in N$ . Therefore, knowing  $s_i = \mathbf{x}_i - \xi_i$ , it can be concluded that  $\mathbf{x}_1 = \mathbf{x}_i = \mathbf{x}_j$  for all  $i, j \in N$ .  $\square$

### 4.6.1 Simulation Results

In this section, the synchronization of autonomous underwater vehicles (AUVs) in the longitudinal plane by having partial state information, using the measurement matrix (2.13), is investigated. For the sake of simulation, the single-input model (3.53) is considered.

For simulations purposes, the control gain  $K$  and the observer gain  $F$  of equation (4.41) are designed according to (4.51).

$$\begin{aligned} K &= \begin{bmatrix} 0.3320 & -0.5650 & 1.0000 & -3.7872 \\ 0.7908 & 0.6427 & 0.0061 & 1.1843 \end{bmatrix} \\ F &= \begin{bmatrix} 0.3544 & 0.2186 & 0.3389 \\ 0.2176 & 0.1111 & 0.1051 \\ 0.1111 & 1.8092 & -0.5332 \\ 0.1051 & -0.5332 & 0.9565 \end{bmatrix} \end{aligned} \quad (4.51)$$

The simulation results of healthy scenario are shown in Figures 4.60 - 4.69.

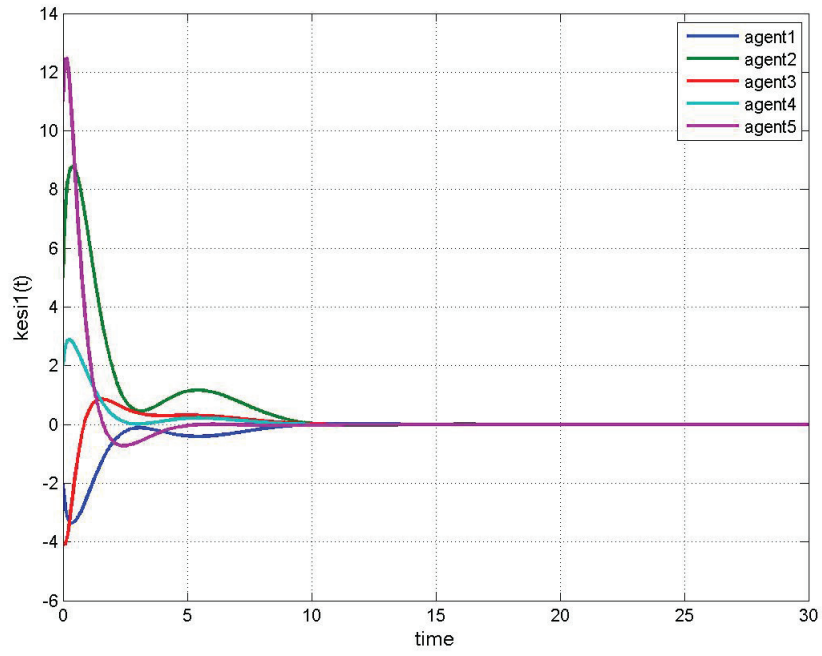


Figure 4.60: Controller state 1 for proposed dynamic output feedback protocol under healthy scenario.

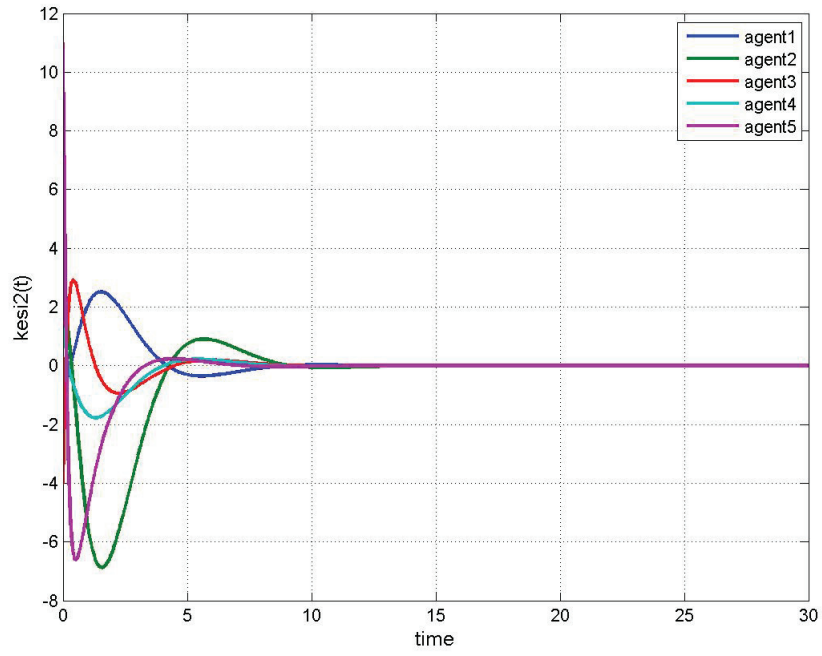


Figure 4.61: Controller state 2 for proposed dynamic output feedback protocol under healthy scenario.

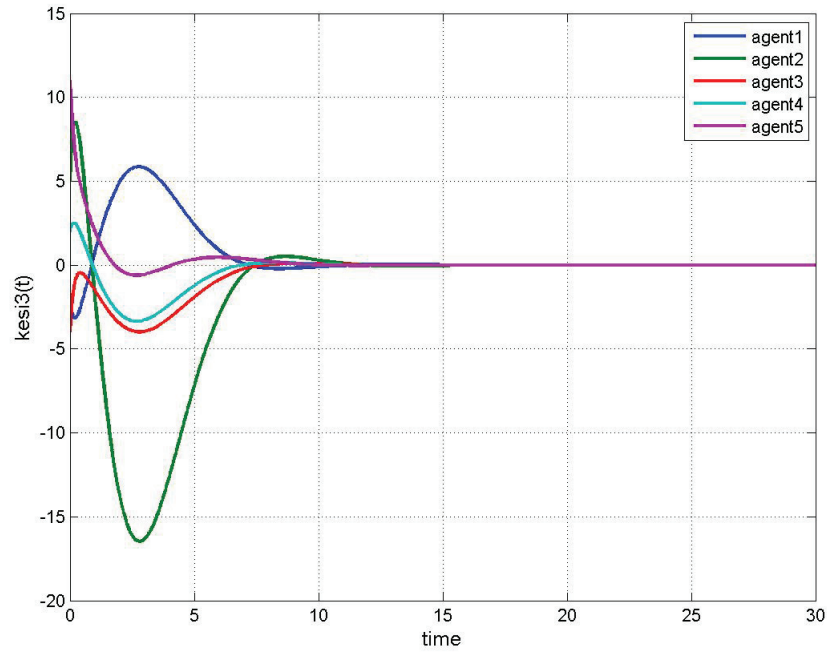


Figure 4.62: Controller state 3 for proposed dynamic output feedback protocol under healthy scenario.

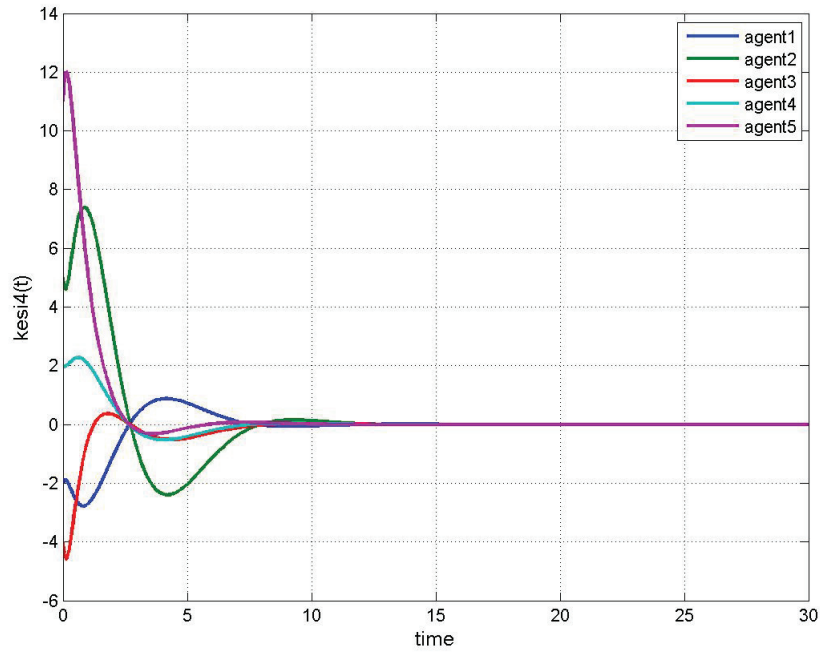


Figure 4.63: Controller state 4 for proposed dynamic output feedback protocol under healthy scenario.

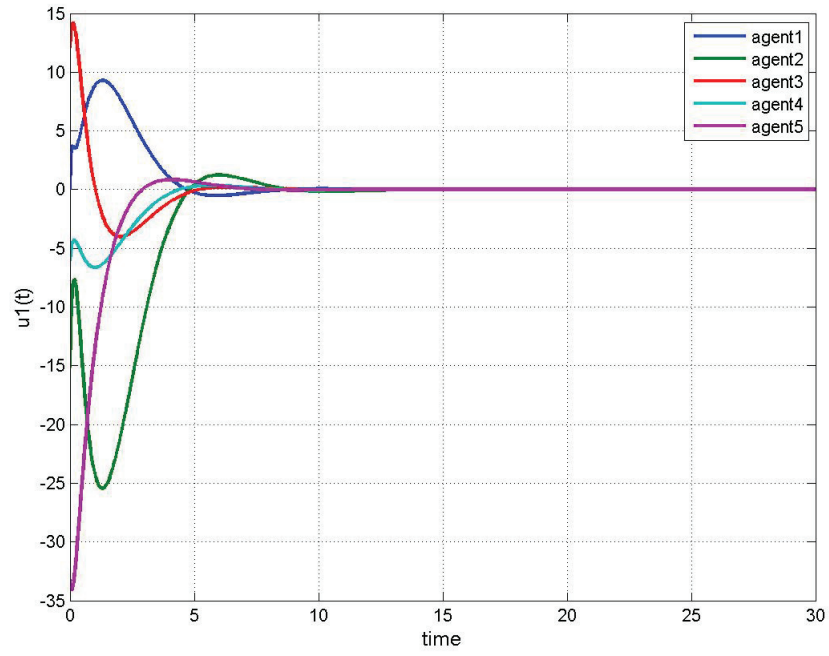


Figure 4.64: Control input signal  $u_1(t)$  for proposed dynamic output feedback protocol under healthy scenario.

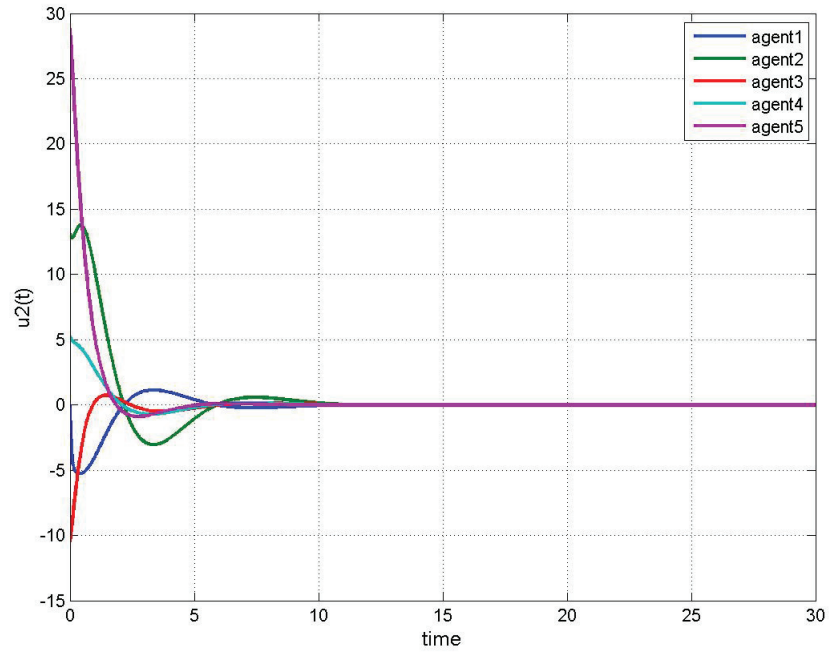


Figure 4.65: Control input signal  $u_2(t)$  for proposed dynamic output feedback protocol under healthy scenario.

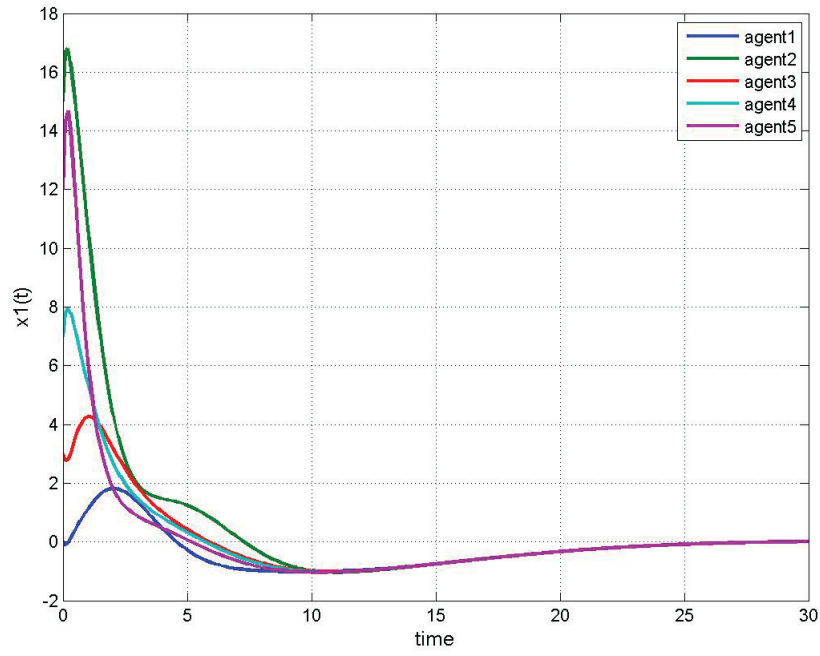


Figure 4.66: Synchronization of state 1 for proposed dynamic output feedback protocol under healthy scenario.

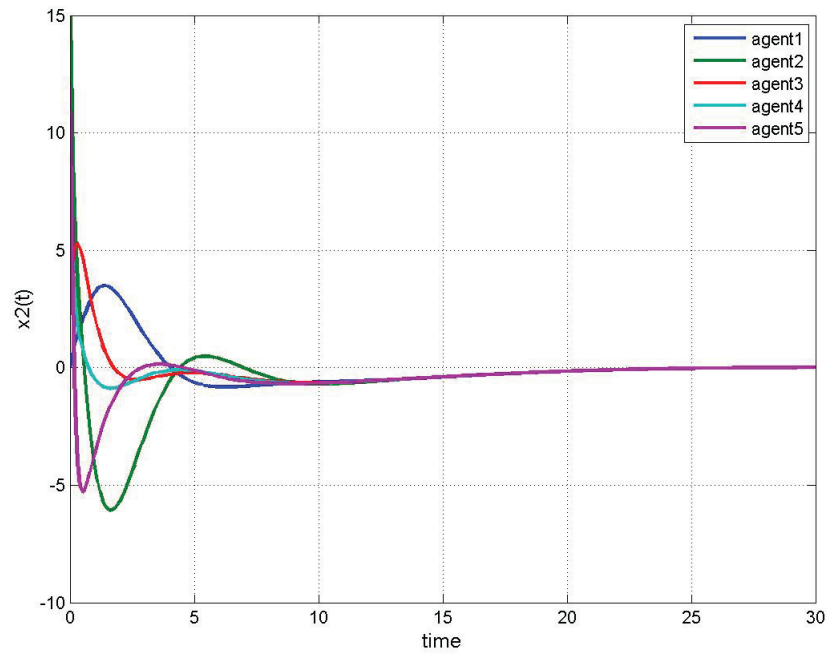


Figure 4.67: Synchronization of state 2 for proposed dynamic output feedback protocol under healthy scenario.

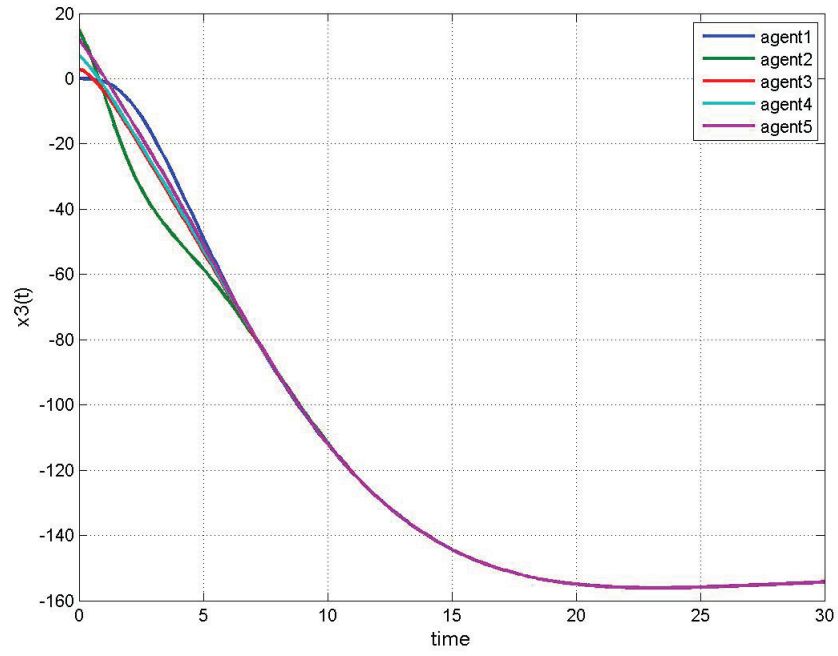


Figure 4.68: Synchronization of state 3 for proposed dynamic output feedback protocol under healthy scenario.

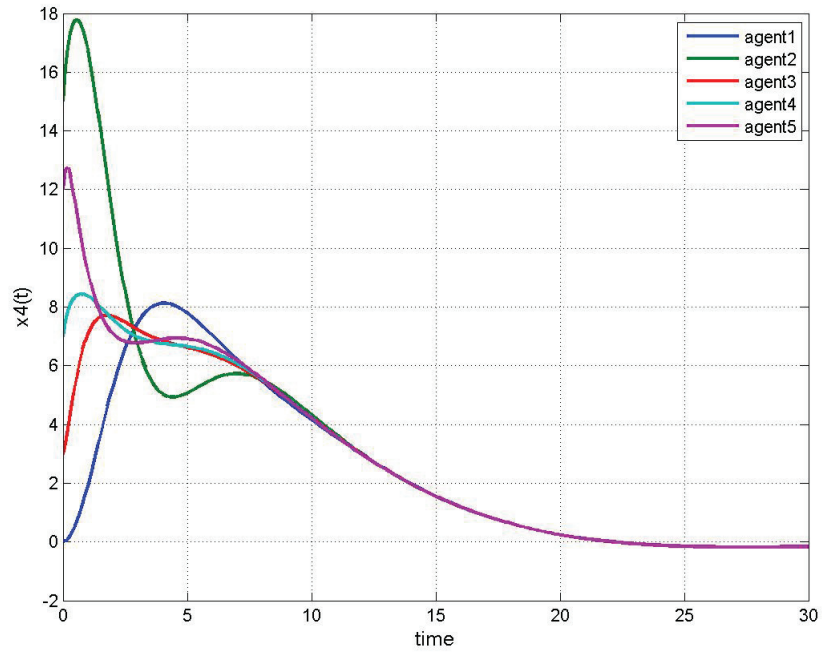


Figure 4.69: Synchronization of state 4 for proposed dynamic output feedback protocol under healthy scenario.

In Figures 4.70 - 4.72, the simulation results of the system subject to float and LOE faults are shown. It is assumed that there is no fault detection and recovery module.

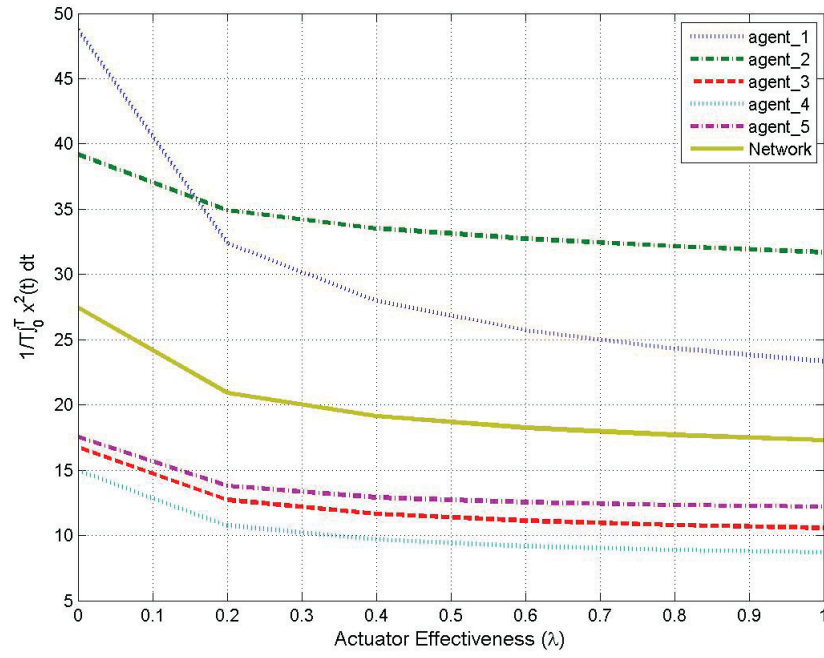


Figure 4.70: State performance index for proposed dynamic output feedback protocol without fault recovery.

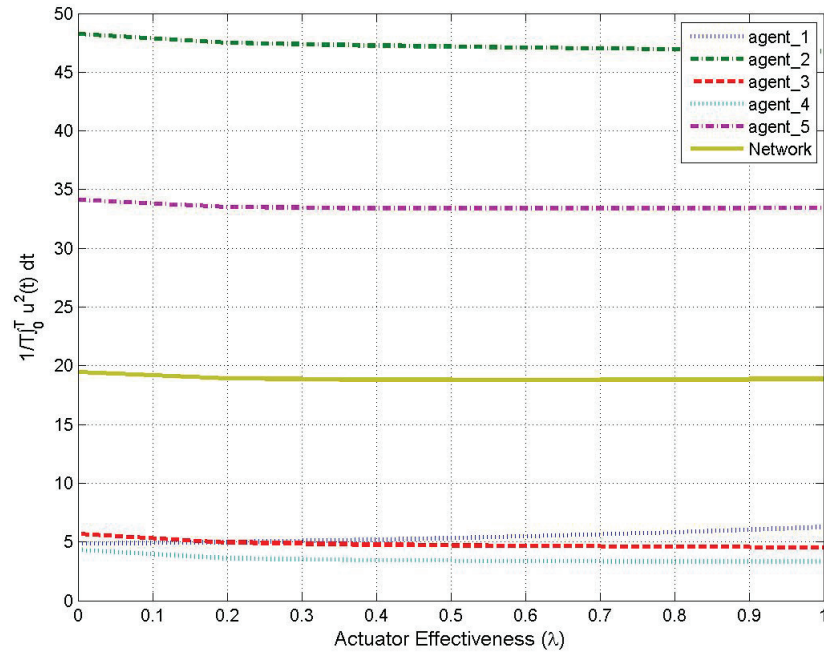


Figure 4.71: Control input performance index for proposed dynamic output feedback protocol without fault recovery.

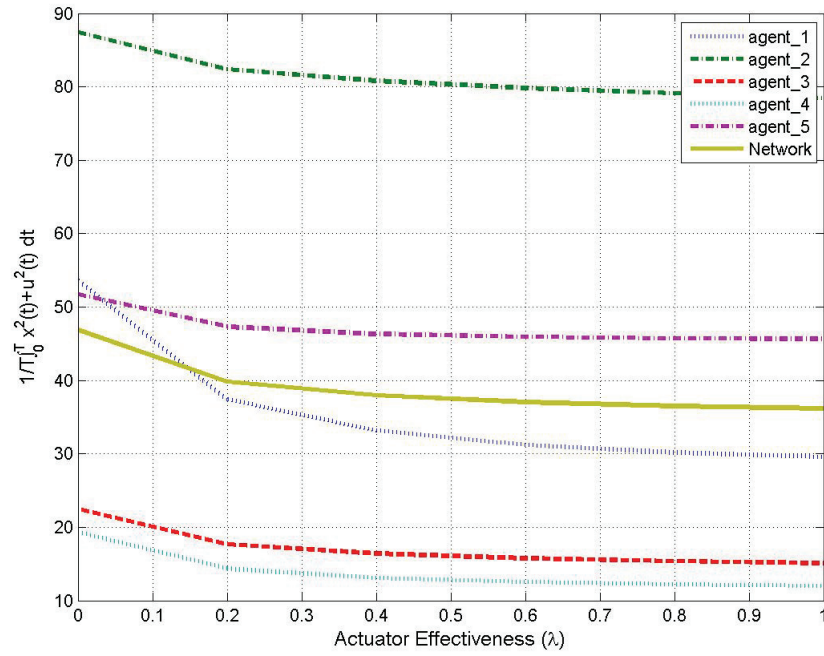


Figure 4.72: Total performance index for proposed dynamic output feedback protocol without fault recovery.

As in Figures 4.70 - 4.72, when the severity of fault increases the performance indices worsen. In order to know how severe the effect of fault is, detailed results for worst case scenario (float fault) are shown in Figures 4.73 - 4.82.

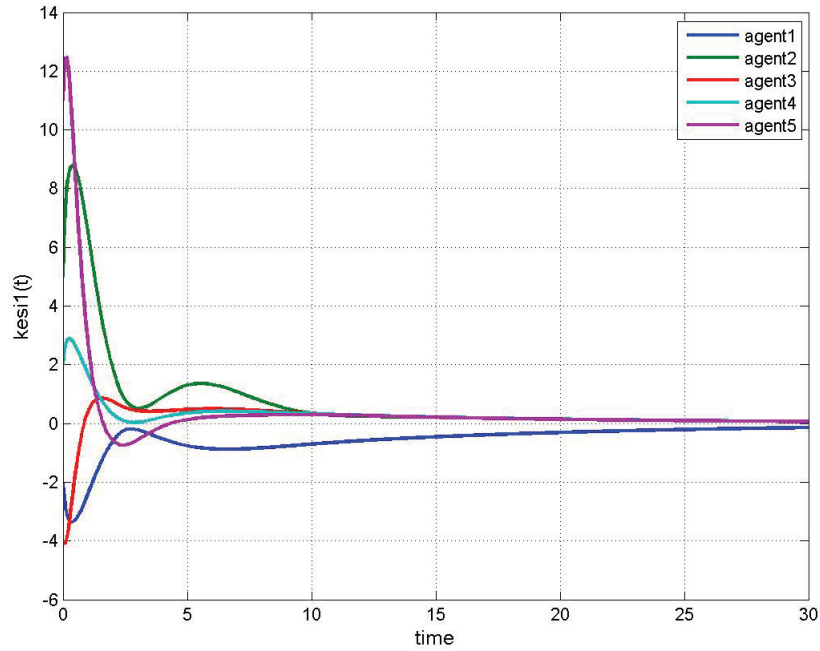


Figure 4.73: Controller state 1 for proposed dynamic output feedback protocol without fault recovery (float fault).

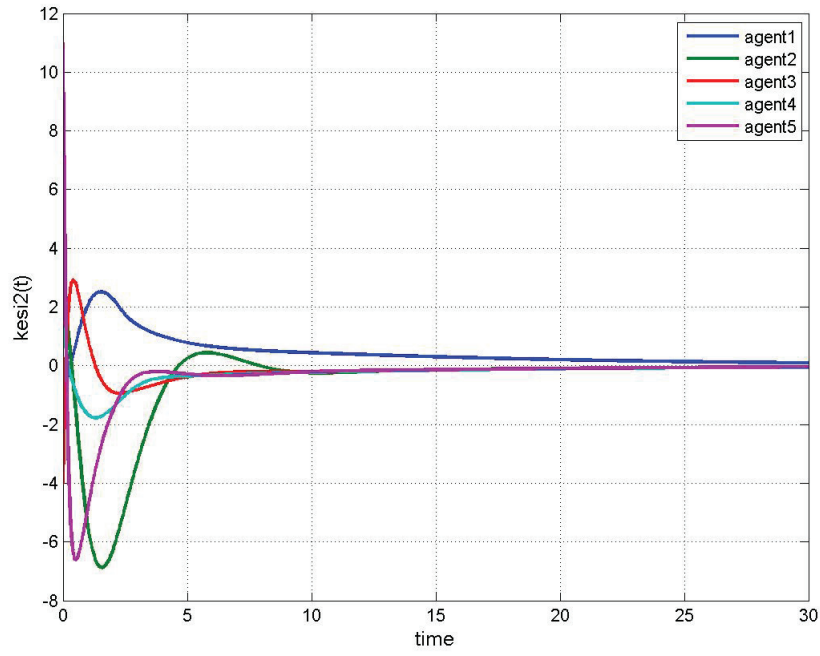


Figure 4.74: Controller state 2 for proposed dynamic output feedback protocol without fault recovery (float fault).

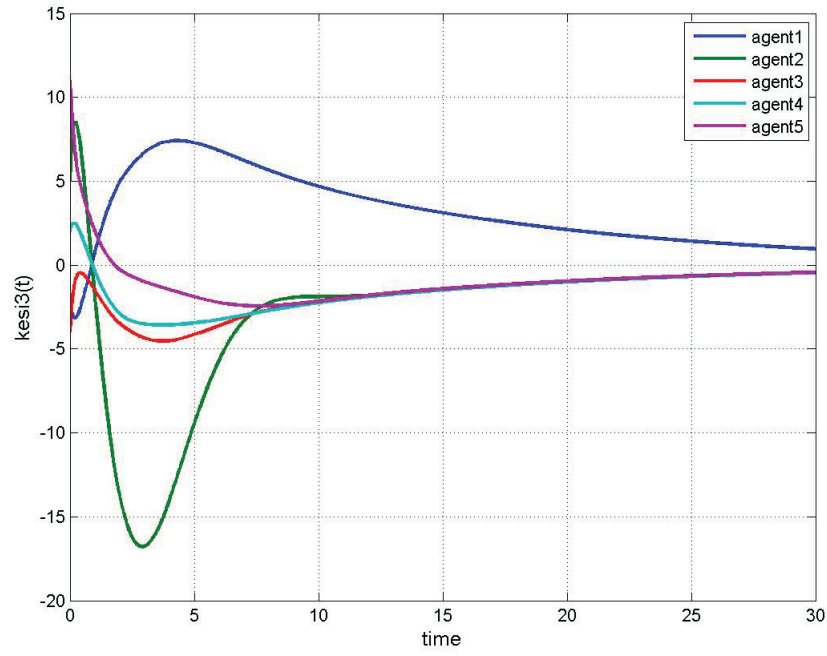


Figure 4.75: Controller state 3 for proposed dynamic output feedback protocol without fault recovery (float fault).

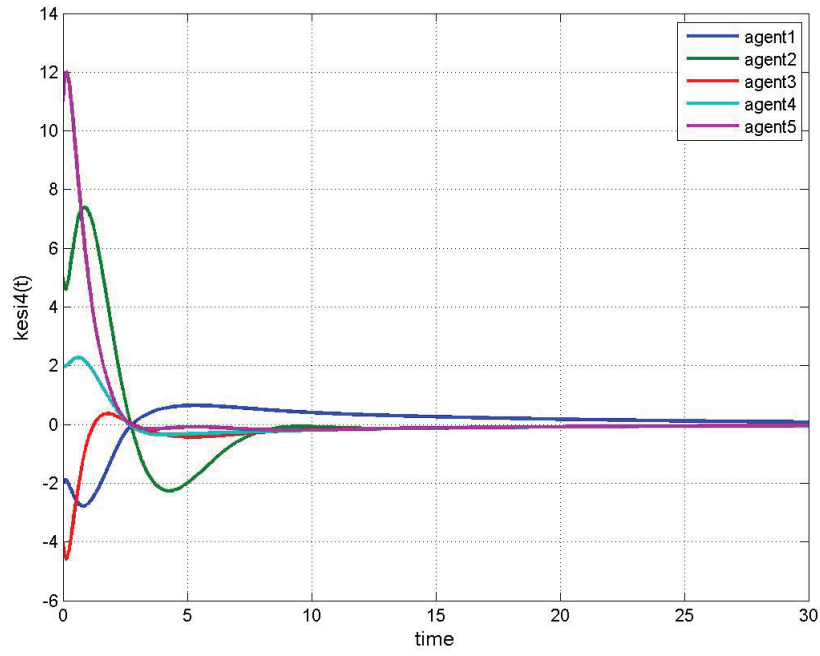


Figure 4.76: Controller state 4 for proposed dynamic output feedback protocol without fault recovery (float fault).

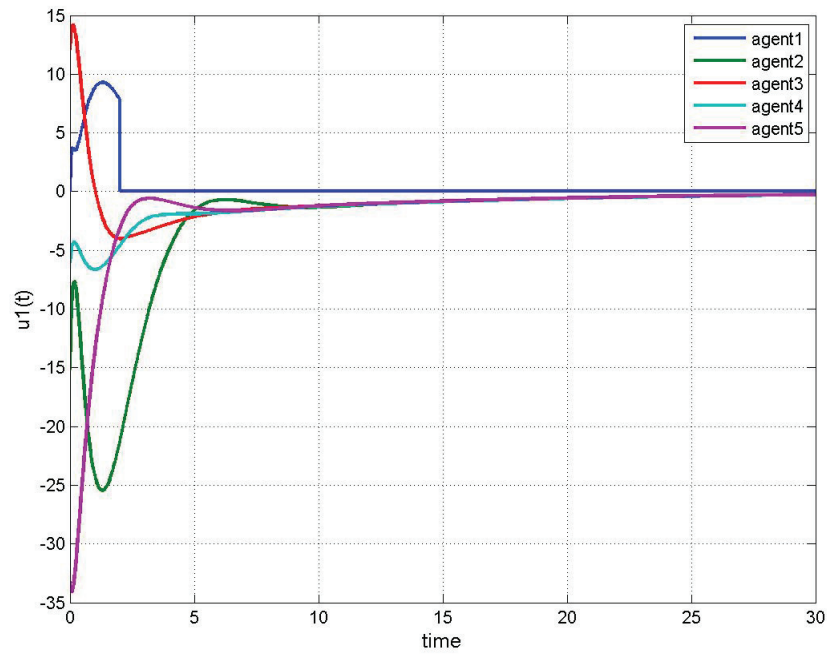


Figure 4.77: Control input signal  $u_1(t)$  for proposed dynamic output feedback protocol without fault recovery (float fault).

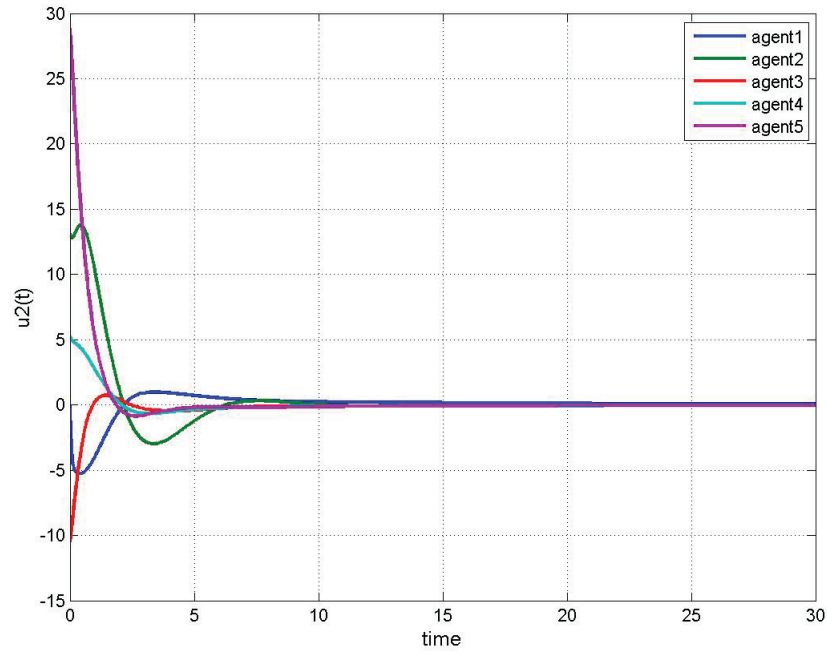


Figure 4.78: Control input signal  $u_2(t)$  for proposed dynamic output feedback protocol without fault recovery (float fault).

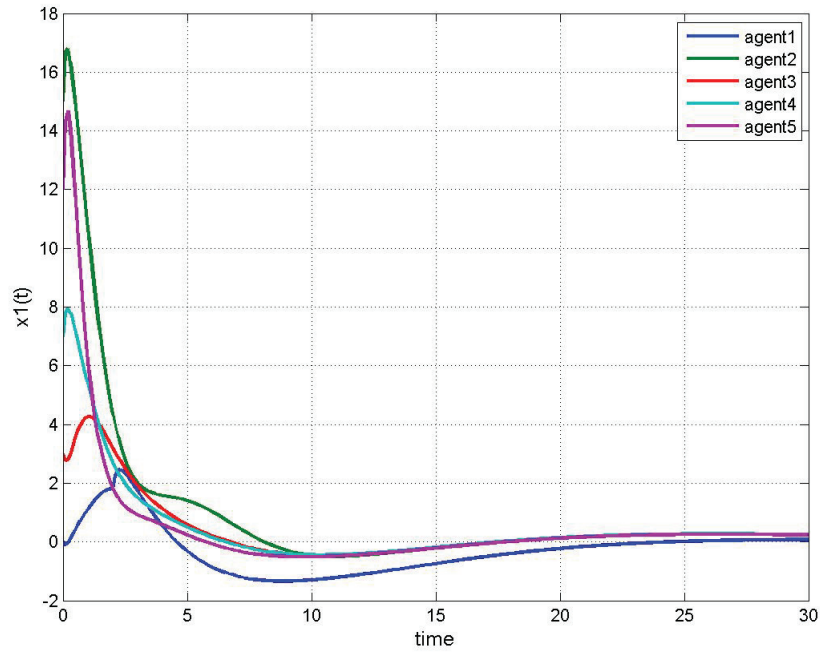


Figure 4.79: Synchronization of state 1 for proposed dynamic output feedback protocol without fault recovery (float fault).

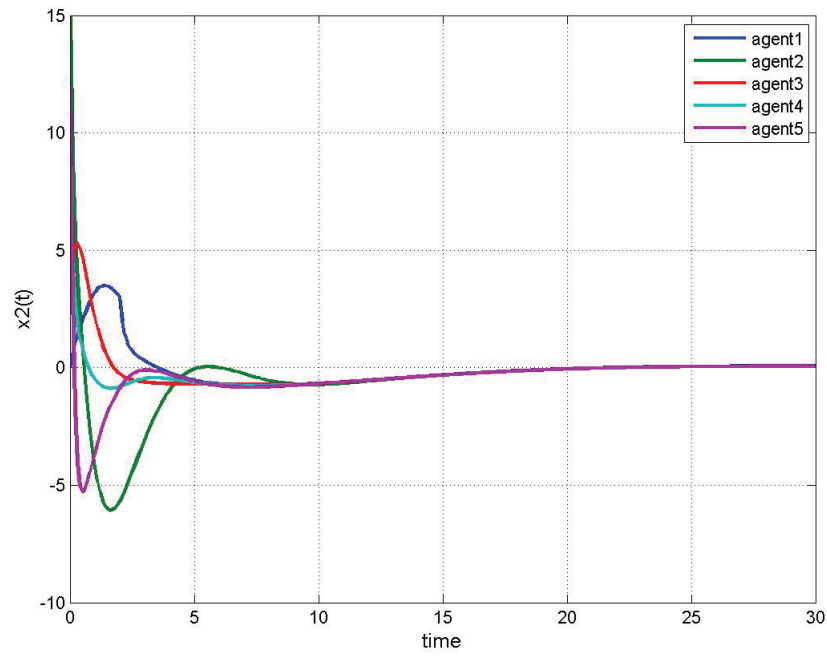


Figure 4.80: Synchronization of state 2 for proposed dynamic output feedback protocol without fault recovery (float fault).

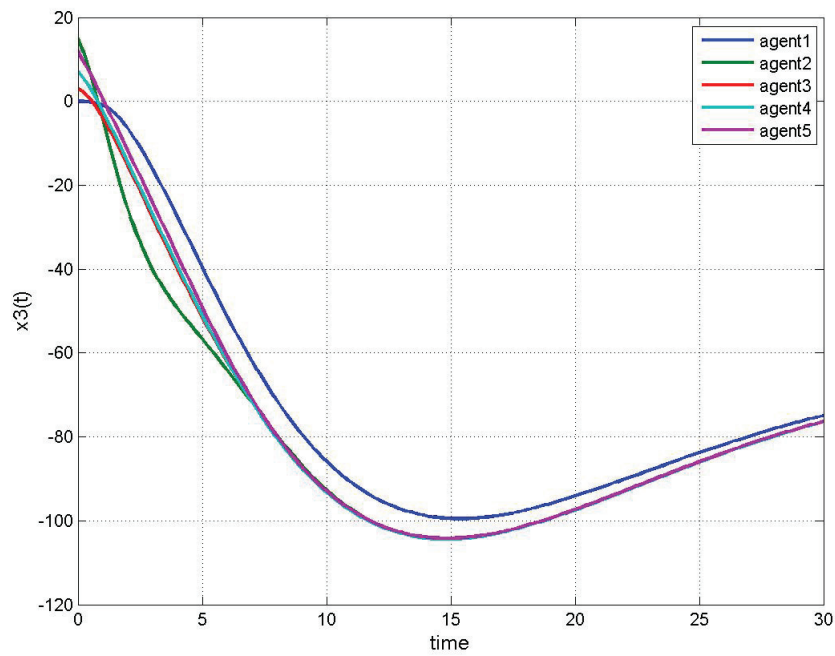


Figure 4.81: Synchronization of state 3 for proposed dynamic output feedback protocol without fault recovery (float fault).

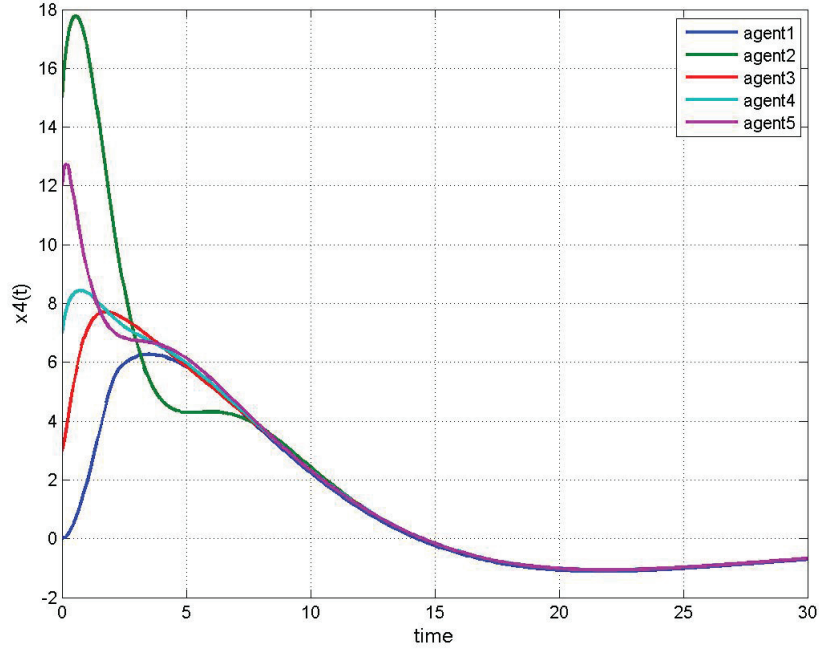


Figure 4.82: Synchronization of state 4 for proposed dynamic output feedback protocol without fault recovery (float fault).

From equation (4.47), which covers the system subject to the LOE/float fault, it can be seen that  $\sigma$  and  $\xi_1$  are independent from  $\nu$ . Therefore, casting them together and designing  $K$  and  $F$  such that the augmented system is stable, the entire system of (4.47) will stabilize as well.

As opposed to the faulty scenario, which requires  $K$  and  $F$  to be designed simultaneously, under healthy conditions,  $K$  and  $F$  in equation (4.48) are designed separately. The simultaneous design of  $K$  and  $F$  may make the design procedure more challenging.

As in the above simulation results, even after fault occurrence synchronization is achieved. This shows that this method has a high robustness. However, the performance indices increase. Therefore, to have a better performance, control gain redesign is required. In the following, the simulation results for float fault and by redesigning the controller gain after fault occurrence have been presented.

### Simulation Results for a Reconfigured System

In order to have a better performance, redesign of the control gain is required. Therefore, the control gain of the faulty agent (agent 1) after fault occurrence is obtained as

$$K_f = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0.8654 & 1.5632 & -1.0000 & 6.6600 \end{bmatrix} \quad (4.52)$$

The simulation results for this case are presented in Figures 4.83 - 4.92 and the performance indices are summarized in Tables 4.1 - 4.3.

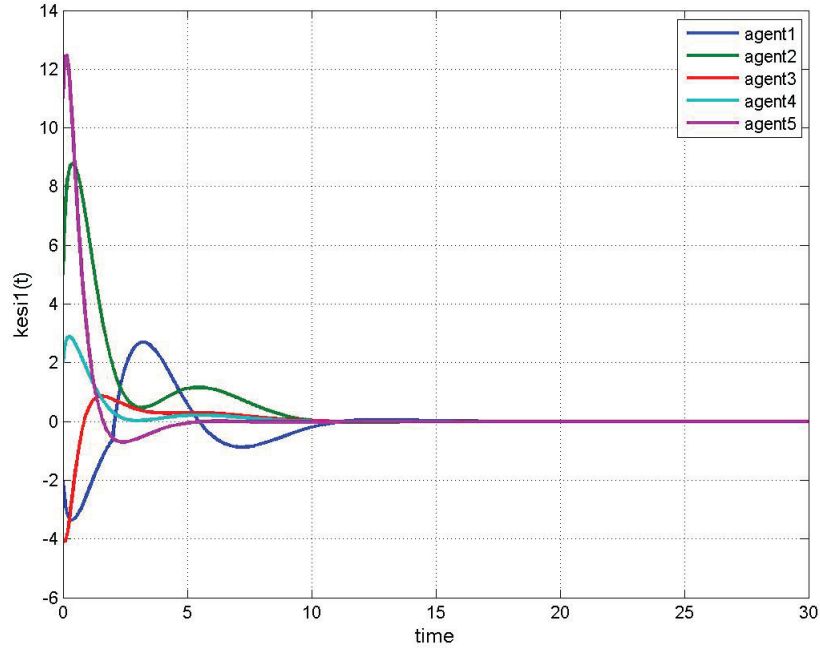


Figure 4.83: Controller state 1 for proposed dynamic output feedback protocol with fault recovery (float fault).

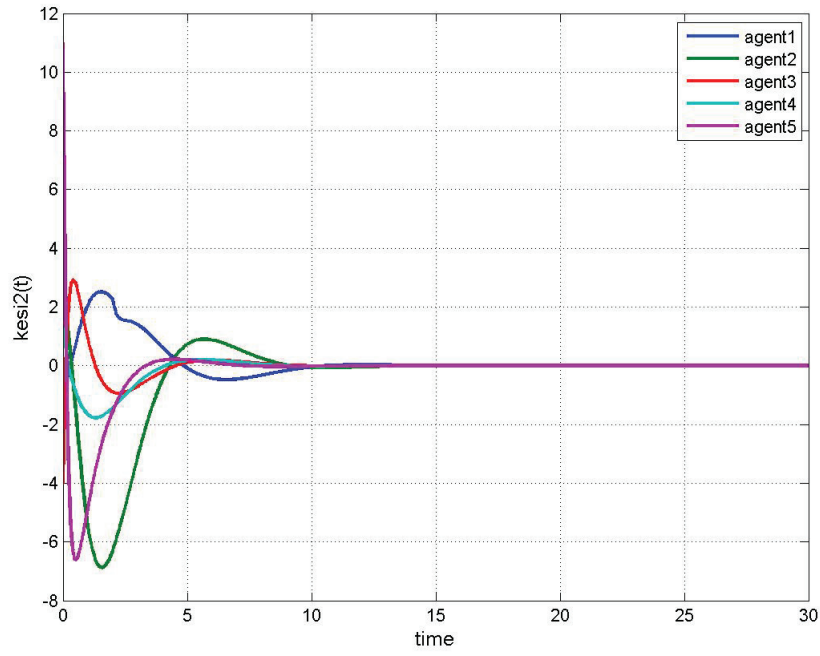


Figure 4.84: Controller state 2 for proposed dynamic output feedback protocol with fault recovery (float fault).

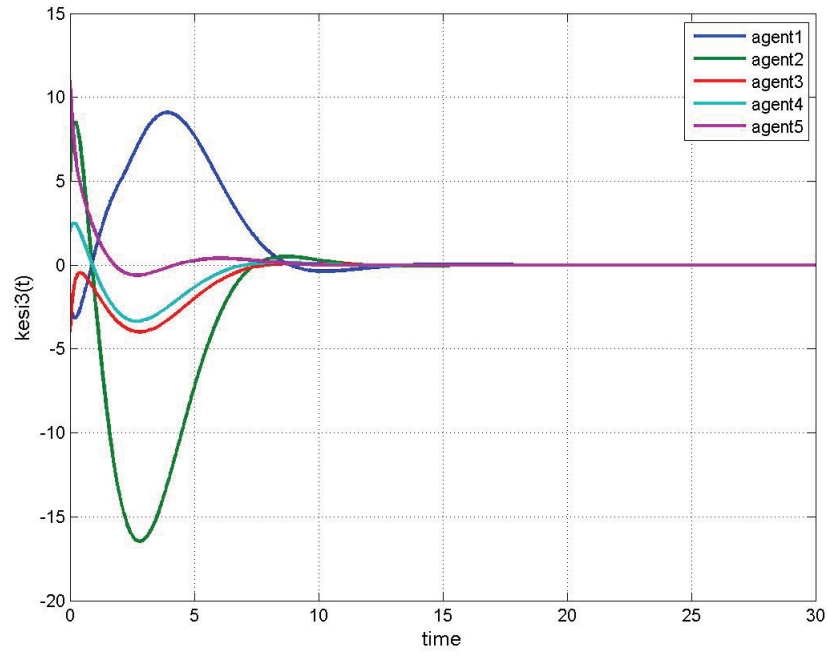


Figure 4.85: Controller state 3 for proposed dynamic output feedback protocol with fault recovery (float fault).

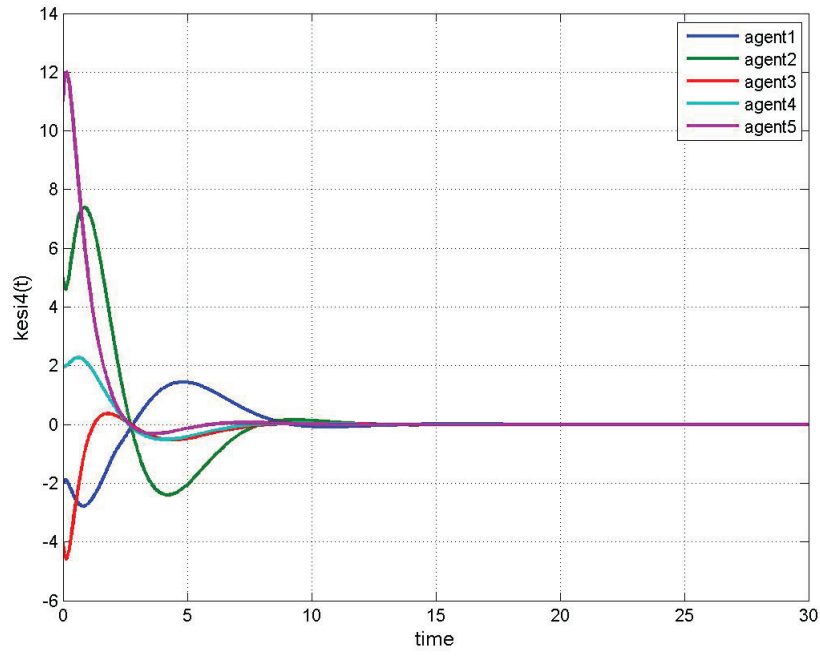


Figure 4.86: Controller state 4 for proposed dynamic output feedback protocol with fault recovery (float fault).

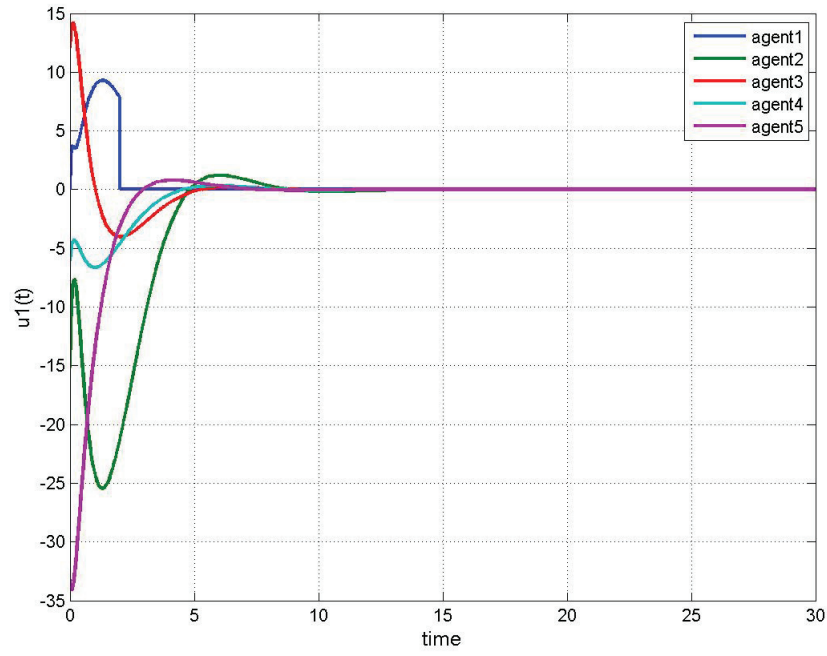


Figure 4.87: Control input signal  $u_1(t)$  for proposed dynamic output feedback protocol with fault recovery (float fault).

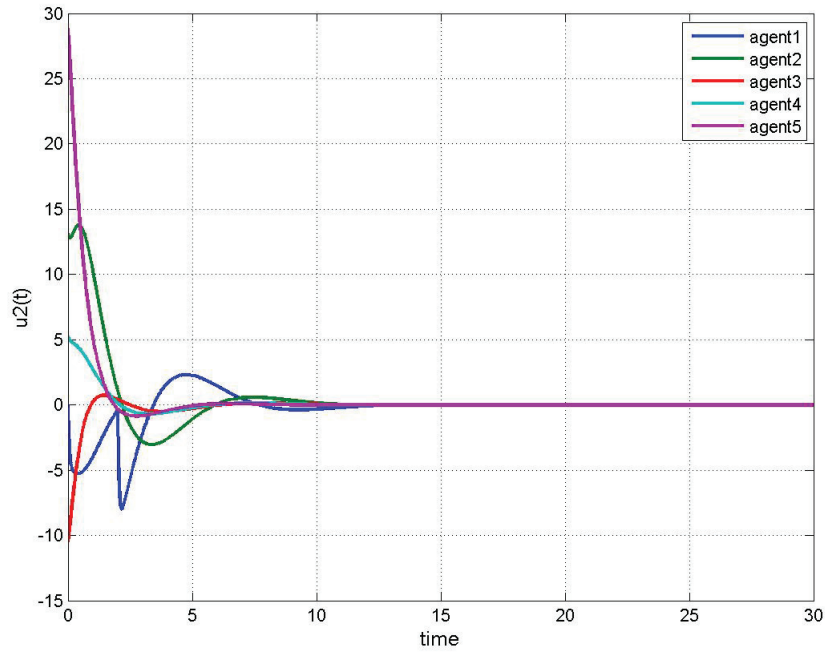


Figure 4.88: Control input signal  $u_2(t)$  for proposed dynamic output feedback protocol with fault recovery (float fault).

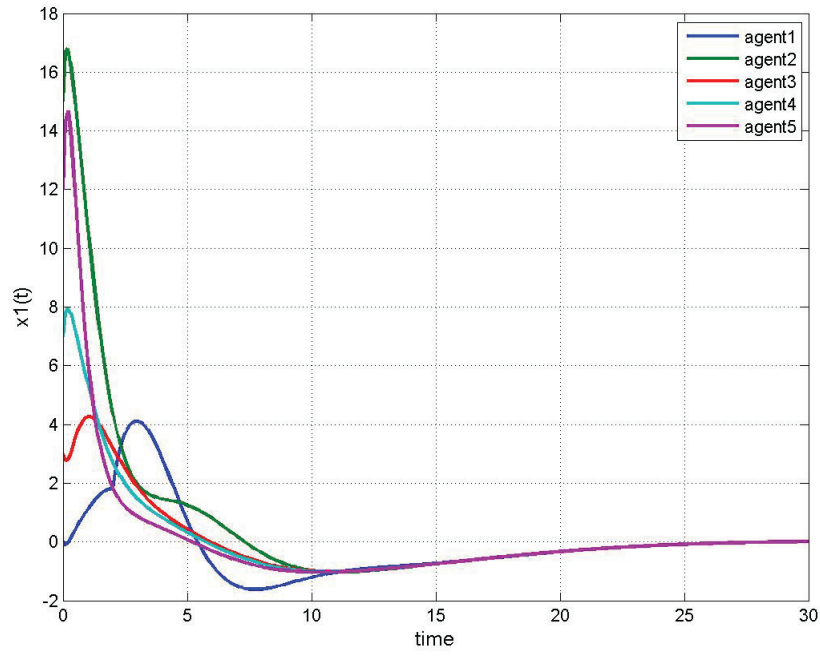


Figure 4.89: Synchronization of state 1 for proposed dynamic output feedback protocol with fault recovery (float fault).

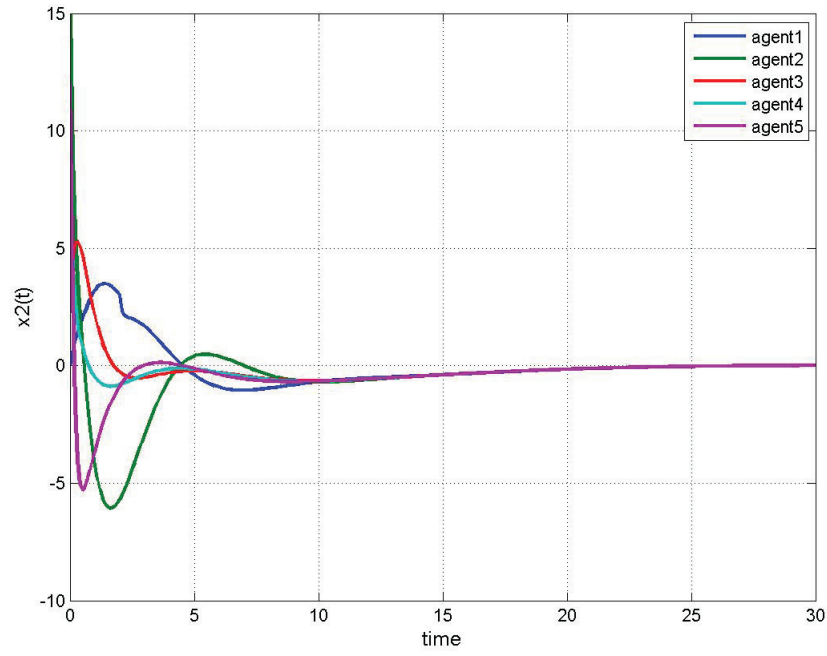


Figure 4.90: Synchronization of state 2 for proposed dynamic output feedback protocol with fault recovery (float fault).

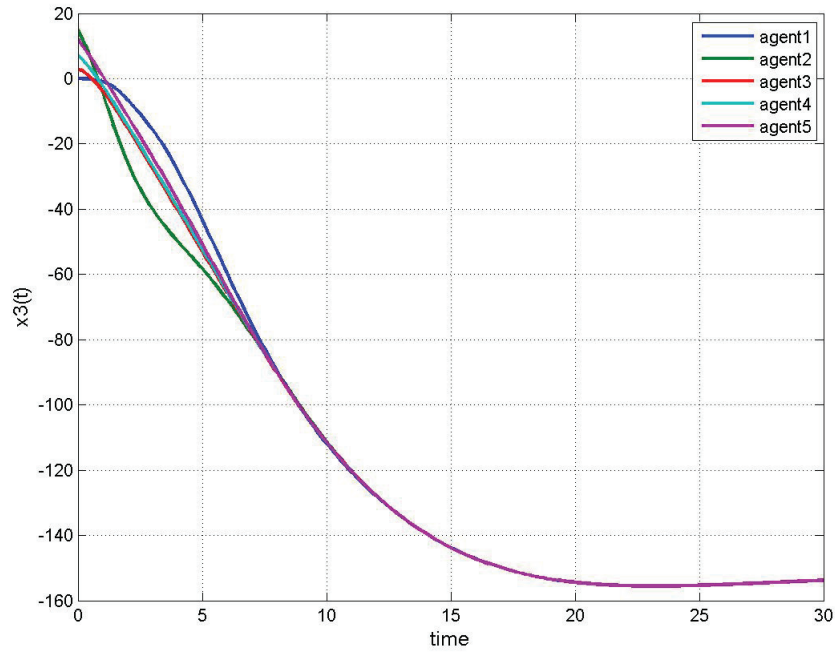


Figure 4.91: Synchronization of state 3 for proposed dynamic output feedback protocol with fault recovery (float fault).

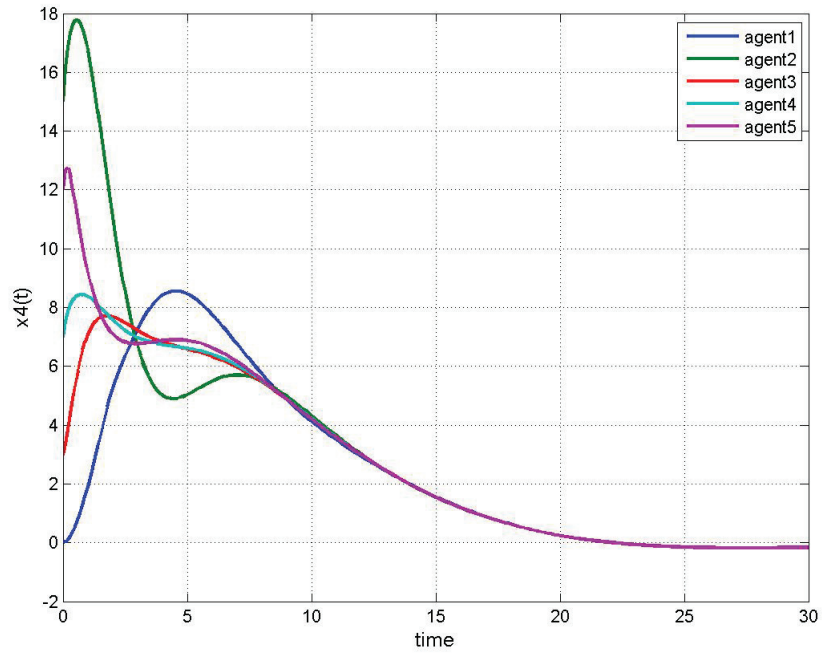


Figure 4.92: Synchronization of state 4 for proposed dynamic output feedback protocol with fault recovery (float fault).

Table 4.1: Comparison of state performance indices between healthy case and the one subject to the float fault.

$PI_X$	Agent 1	Agent 2	Agent 3	Agent 4	Agent 5	Network
No fault	23	32	11	9	12	17
No Recovery	49	39	17	15	17	27
With Recovery	6	47	5	3	33	21

Table 4.2: Comparison of control input performance indices between healthy case and the one subject to the float fault.

$PI_U$	Agent 1	Agent 2	Agent 3	Agent 4	Agent 5	Network
No fault	6	46	4	3	33	19
No Recovery	5	48	5	4	34	19
With Recovery	6	46	5	3	33	19

Table 4.3: Comparison of performance indices between healthy case and the one subject to the float fault.

$PI_{XU}$	Agent 1	Agent 2	Agent 3	Agent 4	Agent 5	Network
No fault	29	78	15	12	46	36
No Recovery	54	88	22	19	52	47
With Recovery	38	82	17	14	47	40

As it can be seen that by recovering the fault, the states performance indices become more close to the healthy case. However, the control input performance indices are almost the same as no-fault and without fault-recovery scenarios. This is due to the fact that the control input performance index is based on the signal fed to the system and not the one calculated by the controller.

## 4.7 Summary and Comparison of Three Output Feedback Synchronization Methods

In the first part of this chapter, the output feedback synchronization by devising a dynamic control protocol, and under healthy scenario has been addressed. The problem has been considered both in absence and presence of noise as follows:

- Output feedback synchronization by devising a dynamic control protocol and Luenberger observer in Section 4.1
- Output feedback synchronization by devising a dynamic control protocol and Kalman filtering in Section 4.2

The second part of chapter covers faulty scenarios. First, in Section 4.4, a static protocol is applied whose control gain matrix could be designed either by the approach of Section 4.4 or an iterative LMI method [112]. The methodology of Section 4.4 under specific restrictive conditions has a solution that may not always be satisfied. As opposed to this approach, to reach a solution, iterative LMI method requires less conditions. However, it suffers from a heavy computational burden to obtain a solution.

Comparisons of equation (4.37) with equation (4.47), and in general the methodology of Sections 4.5 with the one in Section 4.6 show that the method described in Section 4.6 has less computational effort. Particularly, the approach in Section 4.6 acquires only a dynamic controller and not an observer. However, the one in Section 4.5 requires both dynamic controller and observer. This significantly increases the order of the system, and consequently the computation's burden.

As in simulation results of Section 4.6, which are summarized in Tables 4.1 - 4.3, our proposed methodology is robust even with respect to the float fault (worst case scenario); however, the method in Section 4.5 is not. The results of Section 4.5 when subject to a fault have been shown in Figures 4.32 - 4.45. This shows that the protocol in equation (4.35) of Section 4.5, which is adopted from the literature [36], is not robust. The reason is that the control protocol in equation (4.35) employs the estimated states of an observer that uses the control input matrix information. When a fault occurs, this information is not valid any more and therefore it leads to the states estimation divergence. However, the protocol (4.41) of Section 4.6 is directly based on the output information and therefore no faulty information is fed back to the controller.

However, when there is a need to redesign the controller gain and the fault information is available, the methodology of Section 4.5 is preferred to the one in Section 4.6. The reason is that the approach in Section 4.6 becomes complicated when there is a fault even with a known severity. Particularly, when fault occurs the dynamics of  $\sigma$  and  $\zeta_1$  in (4.47) are not independent any more. Therefore, one should design  $K$  and  $F$  simultaneously to make both dynamics stable. However, the methodology of Section 4.5 breaks the design procedure to two separate steps of designing the observer gain  $H$  and the controller gain  $K$  (or  $K_d$  in faulty case).

# Chapter 5

## Conclusions and Future Work

The main objective of this thesis is to investigate the synchronization protocols in networked multi-vehicle systems of autonomous underwater vehicles. Fault-tolerant control of multi-agent systems have barely been addressed in the literature and therefore the thesis focuses on it. The limited papers in the literature of faulty scenarios cover the integrator systems, or leader-follower architecture. However, this thesis and specifically the following contributions cover the marginally stable linear systems in a leaderless platform, which has not been addressed in the literature.

In this thesis, fault-tolerant synchronization of autonomous underwater vehicles is addressed under two main categories: (i) State feedback synchronization, and (ii) Output feedback synchronization.

In the "state feedback" synchronization scheme, in order to achieve consensus, the absolute measurements of the AUV's states are available. For the state feedback problem and by employing a static protocol, it has been proven that the multi-agent system will synchronize in a stochastic mean square sense in the presence of noise. In order to decrease the high variation of control input and the synchronization error, a Kalman filter estimates the states to be used in synchronization. It has also been shown that the state feedback synchronization is achieved under certain conditions, even when a LOE/float fault occurs, which is demonstrated in the employment of static protocol. As one of the main contributions, our proposed methodology of PI controller is employed to synchronize the team subject to the LIP fault. This is a fully novel approach that models the LIP fault as a disturbance applied to the faulty system, and consequently to the augmented model of the team. Finally, given that the fault (LOE/float) severity is known, our proposed methodology of dynamic state feedback synchronization protocol has been applied.

In the "output feedback" synchronization approach, first a dynamic control protocol has been employed where it uses a Luenberger observer for the state estimation. Next, a Kalman filter has been replaced the Luenberger observer for the state estimation when there is a noise, and the

synchronization achievement in a stochastic mean-square sense has been proven. For LOE/float fault, it has been proven that the static output feedback protocol under some specified conditions can synchronize the system. Finally, two dynamic control methodologies are proposed. The first approach, which is adopted from the literature, exploits both a dynamic controller and a dynamic observer. However, our proposed methodology employs only a dynamic controller which has less computational effort and is more robust subject to the fault.

To name a few future works, fault-tolerant synchronization subject to imperfect diagnosis could be a challenging topic. This could happen in any stage of fault diagnosis including detection, isolation, and identification. The imperfect diagnosis can show itself as no fault detection, delayed fault detection, wrong fault isolation, delayed parameter identification convergence, bias and high covariance of parameter estimation. The integration of FDD with synchronization algorithms is an open research direction. In Appendix B, a preliminary formulation of this problem is discussed.

An alternative approach to deal with synchronization protocol when subject to LOE fault and inexact FDD information is to formulate it as a trajectory following problem. In this method, using a simple consensus protocol the agents agree on a trajectory that is the output of an exogenous system whose dynamics is predetermined.

Another research directions is to investigate the complexities associated with the network topology. The commonly used method for underwater communication is the acoustic channels which is a practical method for a typical water clarity and long ranges. Regarding speed of sound in underwater (approximately 1500 m/s), which is very low, there exist a large propagation delay and a large motion Doppler effect. Therefore, the first step in this direction is to investigate which type of time delays and for which class of linear systems, the synchronization problem has been considered in literature. This could be extended to the synchronization of AUVs.

Finally, an interesting topic in the domain is to investigate the high bit-error probability in underwater networks, and see how it should be modeled in state-space. One common way to consider the bit-error probability is to model it as a random network.

# **Appendices**

# Appendix A

## Fault Diagnosis

In this appendix, a detection observer in combination with a diagnosis observer is employed to identify the fault severity. This approach could be used along with synchronization approaches to keep the consensus of the network subject to fault.

- Detection Observer

The detection observer generates an alarm signal when a fault occurs in an agent. For this propose, the following observer is constructed

$$\dot{\hat{x}}_i(t) = A\hat{x}_i(t) + B\hat{x}_i(t) + L(\hat{y}_i(t) - y_i(t)) \quad (\text{A.1})$$

Let us define  $e_i(t) = \hat{x}_i(t) - x_i(t)$  and therefore

$$\begin{aligned} \dot{e}_i(t) &= \dot{\hat{x}}_i(t) - \dot{x}_i(t) = \\ &A\hat{x}_i(t) + Bu_i(t) + L(\hat{y}_i(t) - y_i(t)) - Ax_i(t) - \gamma Bu(t) = \\ &(A + LC)e_i(t) + (1 - \gamma)Bu(t) \end{aligned} \quad (\text{A.2})$$

Assuming that the pair  $(A, C)$  is detectable, the matrix  $L$  could be designed in a way that  $A + LC$  is Hurwitz. Therefore, when there is no fault  $e_i(t)$  will go to zero. However, in a case of faulty system,  $e_i(t)$  will be non zero. It is to be noted that this is a local observer. In other words, it only uses the local agent's information.

- Diagnosis Observer

In order to identify the fault after its detection, a diagnosis observer can be used as

$$\begin{aligned}\dot{x}_{mi}(t) &= Ax_{mi}(t) + \hat{\gamma}Bu_i(t) + L(y_{mi}(t) - y_i(t)) \\ &= Ax_{mi}(t) + \hat{\gamma}Bu_i(t) + LC(x_{mi}(t) - x_i(t))\end{aligned}\tag{A.3}$$

Defining  $e_{mi}(t) = x_{mi}(t) - x_i(t)$ , we will have

$$\begin{aligned}\dot{e}_{mi}(t) &= \dot{x}_{mi}(t) - \dot{x}_i(t) = \\ &Ax_{mi}(t) + \hat{\gamma}Bu_i(t) + LCe_{mi}(t) - Ax_i(t) - \gamma Bu_i(t) = \\ &Ae_{mi}(t) + (\hat{\gamma} - \gamma)Bu_i(t) + LCe_{mi}(t) = \\ &(A + LC)e_{mi}(t) + \tilde{\gamma}Bu_i(t)\end{aligned}\tag{A.4}$$

and therefore,

$$\begin{aligned}\epsilon_{mi}(t) &= Ce_{mi}(t) = C(SI - A - LC)^{-1}\tilde{\gamma}Bu_i(t) = \\ &\tilde{\gamma}w_1(s)u(t)\end{aligned}\tag{A.5}$$

Equation (A.5) is in the standard format of the error equations widely used in adaptive control. Therefore, the adaptive updating laws used in adaptive control can be directly applied to tune  $\tilde{\gamma}$  as

$$\dot{\tilde{\gamma}} = -F \frac{\epsilon_{mi}(t)\zeta(t)}{1 + \zeta(t)'\zeta(t)}\tag{A.6}$$

when  $\zeta(t) = w_1(s)u(t)$  is stable, the above tuning rule realizes a bounded  $\tilde{\gamma}$ , and  $\dot{\tilde{\gamma}} \in L2$ .  $F > 0$  is a pre-specified gain that defines the tuning rate. It is well known that a persistently exciting signal  $u(t)$  can improve the accuracy of the estimation and therefore we may need to modify it. However, this may not be feasible or it may cause a problem in practical systems. In this case, an exponentially decaying persistent signal should be used to improve the accuracy of the adaptive diagnosis algorithm. Finally, when  $\hat{\gamma} \rightarrow \gamma$ , one can redesign the controller gain.

# Bibliography

- [1] J.-H. Li and P.-M. Lee, “Design of an adaptive nonlinear controller for depth control of an autonomous underwater vehicle,” *Ocean Engineering*, vol. 32, no. 17, pp. 2165–2181, 2005.
- [2] S. Wadoo and P. Kachroo, *Autonomous underwater vehicles: modeling, control design and simulation*. CRC Press, 2010.
- [3] M. DeGroot, “Reaching a consensus,” *Journal of the American Statistical Association*, vol. 69, no. 345, pp. 118–121, 1974.
- [4] S. H. Strogatz, “Exploring complex networks,” *Nature*, vol. 410, no. 6825, pp. 268–276, 2001.
- [5] R. Olfati-Saber, “Ultrafast consensus in small-world networks,” in *American Control Conference, 2005. Proceedings of the 2005*, pp. 2371–2378, IEEE, 2005.
- [6] C. Breder Jr, “Equations descriptive of fish schools and other animal aggregations,” *Ecology*, pp. 361–370, 1954.
- [7] A. Okubo, “Dynamical aspects of animal grouping: swarms, schools, flocks, and herds,” *Advances in biophysics*, vol. 22, pp. 1–94, 1986.
- [8] K. Warburton and J. Lazarus, “Tendency-distance models of social cohesion in animal groups,” *Journal of Theoretical Biology*, vol. 150, no. 4, pp. 473–488, 1991.
- [9] G. Flierl, D. Grünbaum, S. Levins, and D. Olson, “From individuals to aggregations: the interplay between behavior and physics,” *Journal of Theoretical Biology*, vol. 196, no. 4, pp. 397–454, 1999.
- [10] A. Czirók and T. Vicsek, “Collective behavior of interacting self-propelled particles,” *Physica A: Statistical Mechanics and its Applications*, vol. 281, no. 1, pp. 17–29, 2000.
- [11] T. Vicsek, A. Czirók, E. Ben-Jacob, I. Cohen, and O. Shochet, “Novel type of phase transition in a system of self-driven particles,” *Physical Review Letters*, vol. 75, no. 6, pp. 1226–1229, 1995.
- [12] Z. Zhang and M.-Y. Chow, “Convergence analysis of the incremental cost consensus algorithm under different communication network topologies in a smart grid,” *Power Systems, IEEE Transactions on*, vol. 27, no. 4, pp. 1761–1768, 2012.

- [13] A. Bidram, A. Davoudi, F. L. Lewis, and Z. Qu, “Secondary control of microgrids based on distributed cooperative control of multi-agent systems,” *Generation, Transmission & Distribution, IET*, vol. 7, no. 8, pp. 822–831, 2013.
- [14] J. Fax and R. Murray, “Information flow and cooperative control of vehicle formations,” *Automatic Control, IEEE Transactions on*, vol. 49, no. 9, pp. 1465–1476, 2004.
- [15] R. Olfati-Saber and R. Murray, “Distributed cooperative control of multiple vehicle formations using structural potential functions,” in *IFAC World Congress*, pp. 346–352, 2002.
- [16] T. Eren, P. Belhumeur, and A. Morse, “Closing ranks in vehicle formations based on rigidity,” in *Decision and Control, 2002, Proceedings of the 41st IEEE Conference on*, vol. 3, pp. 2959–2964, IEEE, 2002.
- [17] R. Vidal, O. Shakernia, and S. Sastry, “Formation control of nonholonomic mobile robots with omnidirectional visual servoing and motion segmentation,” in *Robotics and Automation, 2003. Proceedings. ICRA’03. IEEE International Conference on*, vol. 1, pp. 584–589, IEEE, 2003.
- [18] C. Reynolds, “Flocks, herds and schools: A distributed behavioral model,” in *ACM SIGGRAPH Computer Graphics*, vol. 21, pp. 25–34, ACM, 1987.
- [19] J. Toner and Y. Tu, “Flocks, herds, and schools: A quantitative theory of flocking,” *Physical Review E*, vol. 58, no. 4, p. 4828, 1998.
- [20] J. Cortés and F. Bullo, “Coordination and geometric optimization via distributed dynamical systems,” *SIAM Journal on Control and Optimization*, vol. 44, no. 5, pp. 1543–1574, 2005.
- [21] F. Paganini, J. Doyle, and S. Low, “Scalable laws for stable network congestion control,” in *Decision and Control, 2001. Proceedings of the 40th IEEE Conference on*, vol. 1, pp. 185–190, IEEE, 2001.
- [22] V. Borkar and P. Varaiya, “Asymptotic agreement in distributed estimation,” *Automatic Control, IEEE Transactions on*, vol. 27, no. 3, pp. 650–655, 1982.
- [23] J. Tsitsiklis and M. Athans, “Convergence and asymptotic agreement in distributed decision problems,” *Automatic Control, IEEE Transactions on*, vol. 29, no. 1, pp. 42–50, 1984.
- [24] A. Jadbabaie, J. Lin, and A. Morse, “Coordination of groups of mobile autonomous agents using nearest neighbor rules,” *Automatic Control, IEEE Transactions on*, vol. 48, no. 6, pp. 988–1001, 2003.
- [25] R. Olfati-Saber and R. Murray, “Consensus problems in networks of agents with switching topology and time-delays,” *Automatic Control, IEEE Transactions on*, vol. 49, no. 9, pp. 1520–1533, 2004.
- [26] W. Ren and E. Atkins, “Distributed multi-vehicle coordinated control via local information exchange,” *International Journal of Robust and Nonlinear Control*, vol. 17, no. 10-11, pp. 1002–1033, 2007.

- [27] S. Li and X. Wang, "Finite-time consensus and collision avoidance control algorithms for multiple auvs," *Automatica*, vol. 49, no. 11, pp. 3359–3367, 2013.
- [28] Y. Kuriki and T. Namerikawa, "Consensus-based cooperative formation control with collision avoidance for a multi-uav system," in *American Control Conference (ACC), 2014*, pp. 2077–2082, IEEE, 2014.
- [29] T. Wang, Y. Zhao, and X. Wang, "Consensus algorithm of multi-agent system with binary-valued communication," in *Control Conference (ASCC), 2015 10th Asian*, pp. 1–6, IEEE, 2015.
- [30] H. Ren, Y. Yu, L. Zhang, and C. Sun, "Adaptive neural control for consensus of multiple uavs with heterogeneous matching uncertainties under a directed graph," in *Control and Decision Conference (CCDC), 2015 27th Chinese*, pp. 2205–2210, IEEE, 2015.
- [31] Z. Li, Z. Duan, G. Chen, and L. Huang, "Consensus of multiagent systems and synchronization of complex networks: A unified viewpoint," *Circuits and Systems I: Regular Papers, IEEE Transactions on*, vol. 57, no. 1, pp. 213–224, 2010.
- [32] J. Wang, D. Cheng, and X. Hu, "Consensus of multi-agent linear dynamic systems," *Asian Journal of Control*, vol. 10, no. 2, pp. 144–155, 2008.
- [33] Z. Lin, M. Broucke, and B. Francis, "Local control strategies for groups of mobile autonomous agents," *Automatic Control, IEEE Transactions on*, vol. 49, no. 4, pp. 622–629, 2004.
- [34] C. Ma and J. Zhang, "Necessary and sufficient conditions for consensusability of linear multi-agent systems," *Automatic Control, IEEE Transactions on*, vol. 55, no. 5, pp. 1263–1268, 2010.
- [35] S. Tuna, "Synchronizing linear systems via partial-state coupling," *Automatica*, vol. 44, no. 8, pp. 2179–2184, 2008.
- [36] L. Scardovi and R. Sepulchre, "Synchronization in networks of identical linear systems," *Automatica*, vol. 45, no. 11, pp. 2557–2562, 2009.
- [37] R. Olfati-Saber, J. Fax, and R. Murray, "Consensus and cooperation in networked multi-agent systems," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 215–233, 2007.
- [38] F. Xiao and L. Wang, "Asynchronous consensus in continuous-time multi-agent systems with switching topology and time-varying delays," *Automatic Control, IEEE Transactions on*, vol. 53, no. 8, pp. 1804–1816, 2008.
- [39] R. Bru, L. Elsner, M. Neumann, *et al.*, "Convergence of infinite products of matrices and inner–outer iteration schemes," *Electronic Transactions on Numerical Analysis*, vol. 2, pp. 183–193, 1994.
- [40] A. Vladimirov, L. Elsner, and W. Beyn, "Stability and paracontractivity of discrete linear inclusions," *Linear Algebra and its Applications*, vol. 312, no. 1, pp. 125–134, 2000.

- [41] L. Moreau, “Stability of continuous-time distributed consensus algorithms,” in *Decision and Control, 2004. CDC. 43rd IEEE Conference on*, vol. 4, pp. 3998–4003, IEEE, 2004.
- [42] Y. Hatano and M. Mesbahi, “Agreement over random networks,” *Automatic Control, IEEE Transactions on*, vol. 50, no. 11, pp. 1867–1872, 2005.
- [43] M. Porfiri and D. J. Stilwell, “Consensus seeking over random weighted directed graphs,” *Automatic Control, IEEE Transactions on*, vol. 52, no. 9, pp. 1767–1773, 2007.
- [44] Y. Zhang and Y.-P. Tian, “Consentability and protocol design of multi-agent systems with stochastic switching topology,” *Automatica*, vol. 45, no. 5, pp. 1195–1201, 2009.
- [45] M. Porfiri, D. J. Stilwell, and E. M. Boltt, “Synchronization in random weighted directed networks,” *Circuits and Systems I: Regular Papers, IEEE Transactions on*, vol. 55, no. 10, pp. 3170–3177, 2008.
- [46] D. Lee and M. W. Spong, “Agreement with non-uniform information delays,” in *American Control Conference, 2006*, pp. 6–pp, IEEE, 2006.
- [47] J. Hu and Y. Hong, “Leader-following coordination of multi-agent systems with coupling time delays,” *Physica A: Statistical Mechanics and its Applications*, vol. 374, no. 2, pp. 853–863, 2007.
- [48] P. Lin, Y. Jia, J. Du, and S. Yuan, “Distributed control of multi-agent systems with second-order agent dynamics and delay-dependent communications,” *Asian Journal of Control*, vol. 10, no. 2, pp. 254–259, 2008.
- [49] G. Wen, Z. Duan, W. Yu, and G. Chen, “Consensus of multi-agent systems with nonlinear dynamics and sampled-data information: a delayed-input approach,” *International Journal of Robust and Nonlinear Control*, vol. 23, no. 6, pp. 602–619, 2013.
- [50] Y. Zhang and Y.-P. Tian, “Allowable delay bound for consensus of linear multi-agent systems with communication delay,” *International Journal of Systems Science*, vol. 45, no. 10, pp. 2172–2181, 2014.
- [51] H. Kim, H. Shim, and J. H. Seo, “Output consensus of heterogeneous uncertain linear multi-agent systems,” *Automatic Control, IEEE Transactions on*, vol. 56, no. 1, pp. 200–206, 2011.
- [52] P. Wieland, R. Sepulchre, and F. Allgöwer, “An internal model principle is necessary and sufficient for linear output synchronization,” *Automatica*, vol. 47, no. 5, pp. 1068–1074, 2011.
- [53] K. D. Listmann, A. Wahrburg, J. Strubel, J. Adamy, and U. Konigorski, “Partial-state synchronization of linear heterogeneous multi-agent systems,” in *Decision and Control and European Control Conference (CDC-ECC), 2011 50th IEEE Conference on*, pp. 3440–3445, IEEE, 2011.
- [54] J. Lunze, “An internal-model principle for the synchronisation of autonomous agents with individual dynamics,” in *Decision and Control and European Control Conference (CDC-ECC), 2011 50th IEEE Conference on*, pp. 2106–2111, IEEE, 2011.

- [55] Y. Su and J. Huang, “Cooperative output regulation of linear multi-agent systems,” *Automatic Control, IEEE Transactions on*, vol. 57, no. 4, pp. 1062–1066, 2012.
- [56] P. Dorato, V. Cerone, and C. Abdallah, *Linear-quadratic control: an introduction*. Simon & Schuster, 1994.
- [57] Z. Gao and P. J. Antsaklis, “Stability of the pseudo-inverse method for reconfigurable control systems,” *International Journal of Control*, vol. 53, no. 3, pp. 717–729, 1991.
- [58] F. Wu, X. H. Yang, A. Packard, and G. Becker, “Induced  $l_2$ -norm control for lqv system with bounded parameter variation rates,” in *American Control Conference, Proceedings of the 1995*, vol. 3, pp. 2379–2383, IEEE, 1995.
- [59] J. S. Shamma and M. Athans, “Analysis of gain scheduled control for nonlinear plants,” *Automatic Control, IEEE Transactions on*, vol. 35, no. 8, pp. 898–907, 1990.
- [60] C. Y. Huang and R. F. Stengel, “Restructurable control using proportional-integral implicit model following,” *Journal of Guidance, Control, and Dynamics*, vol. 13, no. 2, pp. 303–309, 1990.
- [61] Y. Zhang and J. Jiang, “Fault tolerant control system design with explicit consideration of performance degradation,” *Aerospace and Electronic Systems, IEEE Transactions on*, vol. 39, no. 3, pp. 838–848, 2003.
- [62] K. J. Åström and B. Wittenmark, *Adaptive control*. Courier Corporation, 2013.
- [63] M. Krstic, P. V. Kokotovic, and I. Kanellakopoulos, *Nonlinear and adaptive control design*. John Wiley & Sons, Inc., 1995.
- [64] S. I. Roumeliotis, G. S. Sukhatme, and G. A. Bekey, “Fault detection and identification in a mobile robot using multiple-model estimation,” in *Robotics and Automation, 1998. Proceedings. 1998 IEEE International Conference on*, vol. 3, pp. 2223–2228, IEEE, 1998.
- [65] A. Andry, E. Shapiro, and J. Chung, “Eigenstructure assignment for linear systems,” *IEEE transactions on aerospace and electronic systems*, vol. 19, no. 5, pp. 711–729, 1983.
- [66] B. Charlet, J. Lévine, and R. Marino, “On dynamic feedback linearization,” *Systems & Control Letters*, vol. 13, no. 2, pp. 143–151, 1989.
- [67] K. Zhou and J. C. Doyle, *Essentials of robust control*, vol. 180. Prentice hall Upper Saddle River, NJ, 1998.
- [68] E. F. Camacho and C. B. Alba, *Model predictive control*. Springer Science & Business Media, 2013.
- [69] C. H. Houpis, S. J. Rasmussen, and M. Garcia-Sanz, *Quantitative feedback theory: fundamentals and applications*. CRC Press, 2005.
- [70] S. P. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan, *Linear matrix inequalities in system and control theory*, vol. 15. SIAM, 1994.

- [71] K. D. Young, V. I. Utkin, and U. Ozguner, "A control engineer's guide to sliding mode control," *IEEE transactions on control systems technology*, vol. 7, no. 3, pp. 328–342, 1999.
- [72] C. E. Garcia, D. M. Prett, and M. Morari, "Model predictive control: theory and practicea survey," *Automatica*, vol. 25, no. 3, pp. 335–348, 1989.
- [73] D. S. H. Jamouli and J. Keller, "Fault tolerant control using augmented fault detection filter," in *Proc. IEEE Int. Symp. Industrial Electronics*, pp. 109–114, 2004.
- [74] Y. Zhang and J. Jiang, "Design of integrated fault detection, diagnosis and reconfigurable control systems," in *Proc. IEEE Descision and Control Conference*, pp. 3587–3592, 1999.
- [75] Y. Zhang and J. Jiang, "Active fault-tolerant control system against partial actuator failures," in *Proc. IEEE American Control Conference*, pp. 95–104, 1999.
- [76] F. Amirarfaei, A. Baniamerian, and K. Khorasani, "Joint kalman filtering and recursive maximum likelihood estimation approaches to fault detection and identification of boeing 747 sensors and actuators," 2013.
- [77] Y. Zhang and J. Jiang, "Bibliographical review on reconfigurable fault-tolerant control systems," *Annual Reviews in Control*, vol. 32, no. 2, pp. 229–252, 2008.
- [78] Y. Zhang and J. Jiang, "Bibliographical review on reconfigurable fault-tolerant control systems," *Annual Reviews in Control*, vol. 32, no. 2, pp. 229–252, 2008.
- [79] J. Jiang and X. Yu, "Fault-tolerant control systems: A comparative study between active and passive approaches," *Annual Reviews in control*, vol. 36, no. 1, pp. 60–72, 2012.
- [80] N. Meskin and K. Khorasani, "Actuator fault detection and isolation for a network of unmanned vehicles," *Automatic Control, IEEE Transactions on*, vol. 54, no. 4, pp. 835–840, 2009.
- [81] N. Meskin and K. Khorasani, "A geometric approach to fault detection and isolation of continuous-time markovian jump linear systems," *Automatic Control, IEEE Transactions on*, vol. 55, no. 6, pp. 1343–1357, 2010.
- [82] M. Tousi, A. Aghdam, and K. Khorasani, "A hybrid fault diagnosis and recovery for a team of unmanned vehicles," in *System of Systems Engineering, 2008. SoSE'08. IEEE International Conference on*, pp. 1–6, IEEE, 2008.
- [83] E. Semsar-Kazerooni and K. Khorasani, "Optimal consensus algorithms for cooperative team of agents subject to partial information," *Automatica*, vol. 44, no. 11, pp. 2766–2777, 2008.
- [84] E. Semsar-Kazerooni and K. Khorasani, "Team consensus for a network of unmanned vehicles in presence of actuator faults," *Control Systems Technology, IEEE Transactions on*, vol. 18, no. 5, pp. 1155–1161, 2010.

- [85] H. Yang, M. Staroswiecki, B. Jiang, and J. Liu, "Fault tolerant cooperative control for a class of nonlinear multi-agent systems," *Systems & Control Letters*, vol. 60, no. 4, pp. 271–277, 2011.
- [86] S. Azizi and K. Khorasani, "A hierarchical architecture for cooperative fault accommodation of formation flying satellites in deep space," in *American Control Conference, 2009. ACC'09.*, pp. 4178–4183, IEEE, 2009.
- [87] E. Kazerooni and K. Khorasani, "Semi-decentralized optimal control of a cooperative team of agents," in *System of Systems Engineering, 2007. SoSE'07. IEEE International Conference on*, pp. 1–7, IEEE, 2007.
- [88] Y. Wang, Y. Song, and F. Lewis, "Robust adaptive fault-tolerant control of multi-agent systems with uncertain non-identical dynamics and undetectable actuation failures," 2015.
- [89] Z. Gallehdari, N. Meskin, and K. Khorasani, "Cost performance based control reconfiguration in multi-agent systems," in *American Control Conference (ACC), 2014*, pp. 509–516, IEEE, 2014.
- [90] Z. Gallehdari, N. Meskin, and K. Khorasani, "Robust cooperative control reconfiguration/recovery in multi-agent systems," in *Control Conference (ECC), 2014 European*, pp. 1554–1561, IEEE, 2014.
- [91] B. Zhou, W. Wang, and H. Ye, "Cooperative control for consensus of multi-agent systems with actuator faults," *Computers & Electrical Engineering*, vol. 40, no. 7, pp. 2154–2166, 2014.
- [92] Q. Xu, H. Yang, B. Jiang, D. Zhou, and Y. Zhang, "Fault tolerant formations control of uavs subject to permanent and intermittent faults," *Journal of Intelligent & Robotic Systems*, vol. 73, no. 1-4, pp. 589–602, 2014.
- [93] A. Mehrabian and K. Khorasani, "Distributed and cooperative quaternion-based attitude synchronization and tracking control for a network of heterogeneous spacecraft formation flying mission," *Journal of the Franklin Institute*, 2015.
- [94] A. Budiyo, "Advances in unmanned underwater vehicles technologies: Modeling, control and guidance perspectives," *Indian Journal of Marine Sciences*, vol. 38, no. 3, pp. 282–295, 2009.
- [95] R. W. Brockett *et al.*, *Asymptotic stability and feedback stabilization*. Citeseer, 1983.
- [96] T. I. Fossen, *Handbook of marine craft hydrodynamics and motion control*. John Wiley & Sons, 2011.
- [97] T. I. Fossen and A. Ross, "Nonlinear modelling, identification and control of uuv's," *IEE CONTROL ENGINEERING SERIES*, vol. 69, p. 13, 2006.
- [98] W. Ren and R. Beard, "Consensus seeking in multiagent systems under dynamically changing interaction topologies," *Automatic Control, IEEE Transactions on*, vol. 50, no. 5, pp. 655–661, 2005.

- [99] R. Horn and C. Johnson, *Matrix analysis*. Cambridge university press, 1990.
- [100] C. Godsil, G. Royle, and C. Godsil, *Algebraic graph theory*, vol. 8. Springer New York, 2001.
- [101] W. Ren, R. Beard, and T. McLain, “Coordination variables and consensus building in multiple vehicle systems,” *Cooperative Control*, pp. 439–442, 2005.
- [102] J. Brewer, “Kronecker products and matrix calculus in system theory,” *Circuits and Systems, IEEE Transactions on*, vol. 25, no. 9, pp. 772–781, 1978.
- [103] S. Osder, “Practical view of redundancy management application and theory,” *Journal of Guidance, Control, and Dynamics*, vol. 22, no. 1, pp. 12–21, 1999.
- [104] J. Ackerman, “Sampled-data control systems—analysis and synthesis, robust system design,” 1985.
- [105] J. Boskovic, S. Bergstrom, and R. Mehra, “Retrofit reconfigurable flight control in the presence of control effector damage,” in *American Control Conference, 2005. Proceedings of the 2005*, pp. 2652–2657, IEEE, 2005.
- [106] C. Silvestre and A. Pascoal, “Depth control of the infante auv using gain-scheduled reduced order output feedback,” *Control Engineering Practice*, vol. 15, no. 7, pp. 883–895, 2007.
- [107] K. J. Åström, *Introduction to stochastic control theory*. Courier Corporation, 2012.
- [108] R. Olfati-Saber, “Distributed kalman filtering for sensor networks,” in *Decision and Control, 2007 46th IEEE Conference on*, pp. 5492–5498, IEEE, 2007.
- [109] R. Olfati-Saber and J. S. Shamma, “Consensus filters for sensor networks and distributed sensor fusion,” in *Decision and Control, 2005 and 2005 European Control Conference. CDC-ECC’05. 44th IEEE Conference on*, pp. 6698–6703, IEEE, 2005.
- [110] W. Ren and R. W. Beard, “Consensus algorithms for double-integrator dynamics,” *Distributed Consensus in Multi-vehicle Cooperative Control: Theory and Applications*, pp. 77–104, 2008.
- [111] V. Kučera and C. De Souza, “A necessary and sufficient condition for output feedback stabilizability,” *Automatica*, vol. 31, no. 9, pp. 1357–1359, 1995.
- [112] Y.-Y. Cao, J. Lam, and Y.-X. Sun, “Static output feedback stabilization: an ilmi approach,” *Automatica*, vol. 34, no. 12, pp. 1641–1645, 1998.