

# **Flight Formation Control**

**Edited by**

**José A. Guerrero**  
**Rogelio Lozano**



**iSTE**

 **WILEY**



## Flight Formation Control



# **Flight Formation Control**

Edited by  
José A. Guerrero  
Rogelio Lozano



First published 2012 in Great Britain and the United States by ISTE Ltd and John Wiley & Sons, Inc.

Apart from any fair dealing for the purposes of research or private study, or criticism or review, as permitted under the Copyright, Designs and Patents Act 1988, this publication may only be reproduced, stored or transmitted, in any form or by any means, with the prior permission in writing of the publishers, or in the case of reprographic reproduction in accordance with the terms and licenses issued by the CLA. Enquiries concerning reproduction outside these terms should be sent to the publishers at the undermentioned address:

ISTE Ltd  
27-37 St George's Road  
London SW19 4EU  
UK

[www.iste.co.uk](http://www.iste.co.uk)

John Wiley & Sons, Inc.  
111 River Street  
Hoboken, NJ 07030  
USA

[www.wiley.com](http://www.wiley.com)

© ISTE Ltd 2012

The rights of José A. Guerrero & Rogelio Lozano to be identified as the author of this work have been asserted by them in accordance with the Copyright, Designs and Patents Act 1988.

---

Library of Congress Cataloging-in-Publication Data

Flight formation control / edited by Jose A. Guerrero, Rogelio Lozano.

p. cm.

Includes bibliographical references and index.

ISBN 978-1-84821-323-4

1. Airplanes--Control systems. 2. Airplanes--Automatic control. 3. Drone aircraft--Control systems. 4. Drone aircraft--Automatic control. 5. Stability of airplanes. I. Guerrero, Jose A. (Jose Alfredo), 1977- II. Lozano, R. (Rogelio), 1954-

TL589.4.F55 2012

629.132'6--dc23

2011052449

---

British Library Cataloguing-in-Publication Data

A CIP record for this book is available from the British Library

ISBN: 978-1-84821-323-4

---

Printed and bound in Great Britain by CPI Group (UK) Ltd., Croydon, Surrey CR0 4YY



## Table of Contents

<b>Chapter 1. Introduction</b>	1
J.A. GUERRERO	
1.1. Motivation . . . . .	1
1.2. Historical background . . . . .	4
1.2.1. Aviation history . . . . .	4
1.2.2. Evolution of UAVs . . . . .	7
1.2.3. UAV classification . . . . .	9
1.3. Flight control . . . . .	9
1.4. Flight formation control . . . . .	11
1.4.1. Multiple-input and multiple-output . . . . .	11
1.4.2. Leader/follower . . . . .	11
1.4.3. Virtual structure . . . . .	12
1.4.4. Behavior-based control . . . . .	12
1.4.5. Passivity-based control . . . . .	12
1.5. Outline of the book . . . . .	13
1.6. Bibliography . . . . .	15
<b>Chapter 2. Theoretical Preliminaries</b>	19
J.A. GUERRERO	
2.1. Passivity . . . . .	19
2.2. Graph theory . . . . .	20
2.3. Robustness problems . . . . .	21
2.3.1. Representation of the parametric uncertainty . . . . .	23
2.3.2. Families of polynomials . . . . .	24
2.4. Bibliography . . . . .	25

<b>Chapter 3. Multiagent Coordination Strategies . . . . .</b>	<b>27</b>
J.A. GUERRERO, R. LOZANO, M.W. SPONG, N. CHOPRA	
3.1. Introduction . . . . .	28
3.2. Controllability and observability of interconnections . . . . .	28
3.2.1. Cyclic topology . . . . .	29
3.2.2. Chain topology: input and output on agent 1 . . . . .	29
3.2.3. Chain topology: input and output on agent 2 . . . . .	29
3.2.4. Eigenvalues and eigenvectors of the system . . . . .	30
3.2.5. General case . . . . .	31
3.2.6. The cyclic topology in the general case . . . . .	32
3.2.6.1. Observability . . . . .	33
3.2.6.2. Controllability . . . . .	33
3.2.7. The chain topology in the general case . . . . .	34
3.2.7.1. Controllability . . . . .	34
3.2.7.2. Observability . . . . .	34
3.2.8. Combinations of chain and cyclic topologies . . . . .	34
3.2.8.1. Controllability . . . . .	35
3.2.8.2. Observability . . . . .	36
3.2.9. Simple configurations that are either non-controllable or non-observable . . . . .	36
3.2.9.1. Example 1 . . . . .	36
3.2.9.2. Example 2 . . . . .	36
3.2.9.3. Example 3 . . . . .	37
3.2.9.4. Example 4 . . . . .	37
3.2.9.5. Example 5 . . . . .	37
3.3. Formation leader tracking . . . . .	37
3.3.1. Formation leader tracking in the general case . . . . .	38
3.3.2. Observer design . . . . .	39
3.3.3. Simulations . . . . .	39
3.4. Time-varying trajectory tracking . . . . .	40
3.5. Linear high-order multiagent consensus . . . . .	44
3.5.1. Trajectory-tracking control . . . . .	48
3.6. Conclusion . . . . .	49
3.7. Bibliography . . . . .	50
<b>Chapter 4. Robust Control Design of Multiagent Systems with Parametric Uncertainty . . . . .</b>	<b>51</b>
J.A. GUERRERO, G. ROMERO	
4.1. Introduction . . . . .	52
4.2. Robust control design . . . . .	54
4.3. Robust stability analysis . . . . .	56

4.3.1. Robust strict positive realness . . . . .	57
4.3.2. Robust absolute stability . . . . .	60
4.4. Robust stability of time-delay systems . . . . .	61
4.5. Application to multiagent systems . . . . .	62
4.5.1. Cyclic topology . . . . .	62
4.5.2. Chain topology . . . . .	66
4.5.3. Balanced graph topology . . . . .	70
4.6. Conclusions . . . . .	73
4.7. Bibliography . . . . .	73
<b>Chapter 5. On Adaptive and Robust Controlled Synchronization of Networked Robotic Systems on Strongly Connected Graphs . . . . .</b>	<b>75</b>
Y.-C. LIU, N. CHOPRA	
5.1. Summary . . . . .	75
5.2. Introduction . . . . .	75
5.3. Problem formulation . . . . .	77
5.4. Adaptive controlled synchronization on strongly connected graphs . . . . .	79
5.4.1. Delay-free synchronization . . . . .	79
5.4.2. Synchronization with time delay . . . . .	81
5.5. Robust controlled synchronization on strongly connected graph . . . . .	83
5.5.1. Delay-free synchronization . . . . .	83
5.5.2. Synchronization with time delay . . . . .	86
5.6. Numerical examples . . . . .	87
5.6.1. Adaptive tracking algorithm . . . . .	88
5.6.2. Robust tracking algorithm . . . . .	90
5.6.3. Disturbances . . . . .	91
5.7. Conclusions . . . . .	93
5.8. Appendix . . . . .	94
5.8.1. Robotic system . . . . .	94
5.8.2. Graph theory . . . . .	95
5.9. Bibliography . . . . .	95
<b>Chapter 6. Modeling and Control of Mini UAV . . . . .</b>	<b>99</b>
G. FLORES COLUNGA, J.A. GUERRERO, J. ESCAREÑO, R. LOZANO	
6.1. Introduction . . . . .	99
6.2. General model . . . . .	101
6.2.1. Translational motion . . . . .	101
6.2.2. Angular motion . . . . .	102
6.2.3. Angular rate . . . . .	103
6.3. Control of a mini tailsitter . . . . .	103
6.3.1. Linear control strategy . . . . .	104
6.3.1.1. Roll subsystem . . . . .	104

6.3.1.2. Pitch subsystem . . . . .	107
6.3.1.3. Yaw subsystem . . . . .	108
6.3.2. Robust control considering parametric uncertainty . . . . .	109
6.3.2.1. Pitch subsystem . . . . .	109
6.3.2.2. Yaw subsystem . . . . .	109
6.3.2.3. Roll subsystem . . . . .	109
6.3.2.4. Time delay case . . . . .	111
6.3.3. Simulation results . . . . .	113
6.3.3.1. Linear controller . . . . .	113
6.3.3.2. Robust controller . . . . .	115
6.3.4. Experimental results . . . . .	115
6.4. Quad-tilting rotor convertible MAV . . . . .	117
6.4.1. Modeling . . . . .	118
6.4.1.1. Aerodynamics . . . . .	119
6.4.1.2. FFF mathematical model . . . . .	122
6.4.2. Transition . . . . .	123
6.4.3. Control strategy for hover flight mode . . . . .	125
6.4.4. Control strategy for forward flight mode . . . . .	127
6.4.5. Simulation results . . . . .	129
6.4.5.1. HF mode . . . . .	129
6.4.5.2. FFF mode . . . . .	131
6.5. Concluding remarks . . . . .	131
6.6. Bibliography . . . . .	132
<b>Chapter 7. Flight Formation Control Strategies for Mini UAVs . . . . .</b>	<b>135</b>
J.A. GUERRERO	
7.1. Introduction . . . . .	135
7.2. Formation geometry . . . . .	137
7.2.1. Triangular formation . . . . .	137
7.2.2. Line formation . . . . .	138
7.3. Communication network . . . . .	138
7.4. Dynamic model . . . . .	139
7.5. Formation flying control based on coordination . . . . .	142
7.5.1. Formation control . . . . .	147
7.6. Formation flying control based on nested saturations . . . . .	148
7.6.1. Formation control . . . . .	152
7.7. Trajectory-tracking control . . . . .	153
7.7.1. Time-varying reference tracking . . . . .	155
7.7.1.1. Chain topology . . . . .	156
7.7.1.2. Cyclic topology . . . . .	157
7.8. Simulation results . . . . .	158
7.8.1. High-order consensus-based formation . . . . .	158
7.8.2. Nested saturations based formation . . . . .	159

7.8.3. Time-varying tracking . . . . .	159
7.9. Conclusions . . . . .	162
7.10. Bibliography . . . . .	162
<b>Chapter 8. Formation Based on Potential Functions . . . . .</b>	<b>165</b>
L. GARCÍA, A. DZUL	
8.1. Introduction . . . . .	165
8.2. Dynamical model . . . . .	166
8.3. Formation control . . . . .	167
8.3.1. Interactive potential energy and force . . . . .	167
8.3.2. Collision avoidance . . . . .	169
8.3.3. Obstacle avoidance . . . . .	169
8.3.4. Total structural force . . . . .	170
8.4. Position control . . . . .	170
8.4.1. Altitude and yaw control . . . . .	171
8.4.2. Nested saturation control . . . . .	171
8.4.2.1. Change of variables for the nested saturation . . . . .	172
8.4.2.2. Nested saturation formation control . . . . .	174
8.4.3. Stability analysis . . . . .	175
8.4.4. Stability analysis for the interconnected system . . . . .	176
8.4.5. Bounded force . . . . .	181
8.4.6. Repulsive distance . . . . .	182
8.5. Simulation results . . . . .	183
8.5.1. Obstacle avoidance . . . . .	185
8.5.2. Multiple formations . . . . .	187
8.6. Conclusions . . . . .	189
8.7. Bibliography . . . . .	189
<b>Chapter 9. Quadrotor Vision-Based Control . . . . .</b>	<b>191</b>
J.E. GOMEZ-BALDERAS, J.A. GUERRERO, S. SALAZAR, R. LOZANO, P. CASTILLO	
9.1. Introduction . . . . .	191
9.2. Quadrotor dynamic model and control . . . . .	194
9.2.1. Dynamic model . . . . .	194
9.2.2. Nonlinear control . . . . .	194
9.2.3. Trajectory-tracking control . . . . .	195
9.3. Computer vision preliminaries . . . . .	197
9.3.1. Camera model . . . . .	197
9.3.2. Projective distortion removal . . . . .	199
9.3.3. Affine distortion removal . . . . .	200
9.4. Tracking of a visual target . . . . .	201
9.4.1. Edge-detection algorithm . . . . .	202
9.4.2. Polygons properties . . . . .	202

x Flight Formation Control

9.4.3. Square-detection algorithm . . . . .	204
9.4.4. Image rectification . . . . .	204
9.4.5. Solving the 3D localization problem . . . . .	205
9.4.6. OF measurement . . . . .	207
9.5. Tracking of a visual line . . . . .	209
9.5.1. Vanishing point detection . . . . .	209
9.6. Embedded architecture . . . . .	214
9.7. Experimental results . . . . .	214
9.7.1. Visual target position stabilization . . . . .	214
9.7.2. Tracking of a visual line with no marks . . . . .	218
9.8. Conclusions . . . . .	221
9.9. Bibliography . . . . .	221
<b>Chapter 10. Toward Vision-Based Coordination of Quadrotor Platoons . . . . .</b>	<b>225</b>
L.R. GARCÍA CARRILLO, J.A. GUERRERO, R. LOZANO	
10.1. Introduction . . . . .	225
10.2. Problem statement . . . . .	227
10.2.1. Description of the process . . . . .	227
10.2.2. Objective of our approach . . . . .	228
10.3. Dynamic model and control of a quadrotor . . . . .	228
10.3.1. Dynamic model . . . . .	228
10.3.2. Vehicle stabilization . . . . .	228
10.4. Vision-based position estimation . . . . .	229
10.4.1. Visual system setup . . . . .	230
10.4.2. Computing the 3D position . . . . .	231
10.4.3. Translational velocities . . . . .	233
10.4.4. Prediction of the landing pad position . . . . .	234
10.5. Coordination position control of two quadrotors . . . . .	235
10.6. Architecture of the experimental platforms . . . . .	237
10.6.1. Quadrotor system . . . . .	237
10.6.2. Ground station . . . . .	238
10.6.3. Monocular imaging system implementation . . . . .	239
10.7. Experimental results . . . . .	240
10.8. Conclusions and future work . . . . .	242
10.9. Bibliography . . . . .	243
<b>Chapter 11. Optimal Guidance for Rotorcraft Platoon Formation Flying in Wind Fields . . . . .</b>	<b>247</b>
J.A. GUERRERO, Y. BESTAOUI, R. LOZANO	
11.1. Introduction . . . . .	247
11.2. Preliminaries . . . . .	250
11.2.1. Dynamic model . . . . .	250

11.2.2. Vehicle control . . . . .	250
11.3. Path planning . . . . .	251
11.3.1. Center of mass of the platoon . . . . .	251
11.3.2. Zermelo navigation problem: case 2D . . . . .	251
11.3.2.1. Navigation equation . . . . .	252
11.3.2.2. One particular solution . . . . .	253
11.3.3. Zermelo navigation problem: case 3D . . . . .	254
11.3.3.1. Constant wind . . . . .	257
11.3.3.2. Linear variation of wind velocity . . . . .	257
11.4. Quadrotor formation control scheme . . . . .	258
11.5. Quadrotor trajectory-tracking control . . . . .	258
11.6. Simulation results . . . . .	259
11.6.1. Reference given to leader vehicle . . . . .	260
11.6.2. Reference given to all vehicles . . . . .	262
11.7. Conclusions and future work . . . . .	264
11.8. Bibliography . . . . .	264
<b>Chapter 12. Impact of Wireless Medium Access Protocol on the Quadrotor Formation Control</b> . . . . .	267
J.A. GUERRERO, Y. CHALLAL, P. CASTILLO	
12.1. Introduction . . . . .	267
12.2. Multi quadrotor consensus . . . . .	269
12.2.1. Quadrotor dynamic model and control . . . . .	269
12.2.2. From individual to collective behavior . . . . .	270
12.3. Multiagent consensus over wireless networks . . . . .	272
12.3.1. CSMA/CA . . . . .	272
12.3.2. TDMA . . . . .	273
12.3.3. Network analysis . . . . .	273
12.4. Quadrotor consensus over wireless networks . . . . .	275
12.5. Simulation results . . . . .	278
12.6. Conclusions and future work . . . . .	282
12.7. Bibliography . . . . .	282
<b>Chapter 13. MAC Protocol for Wireless Communications</b> . . . . .	285
A. MENDEZ, M. PANDURO, O. ELIZARRARAS, D. COVARRUBIAS	
13.1. Introduction . . . . .	285
13.2. Protocols of medium access control . . . . .	287
13.2.1. Slotted ALOHA . . . . .	287
13.2.1.1. Modeling of S-ALOHA . . . . .	287
13.2.2. Carrier sense multiple access . . . . .	290
13.2.2.1. Modeling of CSMA . . . . .	290
13.2.3. Inhibit sense multiple access . . . . .	292
13.2.3.1. Modeling of ISMA . . . . .	292

13.2.4. Results of performance evaluation . . . . .	294
13.3. Proposed MAC protocol . . . . .	296
13.4. Experimental setup and results . . . . .	299
13.5. Conclusions . . . . .	300
13.6. Acknowledgments . . . . .	301
13.7. Bibliography . . . . .	301
<b>Chapter 14. Optimization of a Scannable Pattern for Bidimensional Antenna Arrays to Provide Maximum Performance . . . . .</b>	<b>305</b>
A. REYNA, M.A. PANDURO, A. MENDEZ	
14.1. Introduction . . . . .	305
14.2. Design of planar antenna arrays . . . . .	306
14.2.1. Theoretical model . . . . .	306
14.2.2. Objective function used to optimize planar arrays . . . . .	307
14.2.3. Results obtained for the design of planar arrays . . . . .	308
14.3. Design of concentric ring arrays . . . . .	314
14.3.1. Theoretical model . . . . .	314
14.3.2. Results obtained for the design of concentric ring arrays . . . . .	316
14.4. Discussions and open problems . . . . .	319
14.5. Conclusions . . . . .	320
14.6. Acknowledgments . . . . .	320
14.7. Bibliography . . . . .	320
<b>List of Authors . . . . .</b>	<b>323</b>
<b>Index . . . . .</b>	<b>325</b>

### **1.1. Motivation**

Multiple spacecraft/aircraft flight formation and coordination control are topics that have received a lot of attention over the past decades. Also, the new developments powered by technological advances have spurred a broad interest in autonomous vehicles. The explosion in computation and communication capabilities as well as the advent of miniaturization technologies has increased the interest in a wide variety of research communities, including robotics, communications, automatic control, etc. On the one hand, cooperative and coordinated behavior of a group of unmanned aerial vehicles can cover a larger operational area than a single autonomous vehicle. On the other hand, the lifting of heavy and/or large structures, underway replenishment (fuel, munitions, goods, and personal transfer from one ship to another while under way) and aerial refueling are operations in which coordination is highly required. Thus, the main motivation of this work can be found in the wide variety of applications of multi-autonomous vehicle systems such as in the following examples:

**Formation flying** have been used in survey operations, homeland security, etc. During World War II the groups of B-17 bombers used to fly in a close formation and be escorted by P-51 Mustang fighters also flying in formation to gain better protection as a group. Piloting for many hours in a close formation and under the enemy fire has been proven to be tiring and stressful. Current fighters and bombers fly much faster than those during WWII which may increase the stress and induce nerve-racking

## 2 Flight Formation Control

experiences on pilots. In Figure 1.1 a group of nine aircraft is shown doing a fly pass during the French Bastille Day Military Parade.



**Figure 1.1.** Aircraft formation flying

**Heavy and/or large load transportation** vehicles such as the Helistat and the Skyhook projects that combine features of a blimp and a quadrotor helicopter. The Helistat was planned to be capable of carrying big loads for the US Forest Service. It consisted of a blimp and four Sikorsky Helicopters joined by a metallic structure. All the four helicopters were controlled by a human pilot. The Skyhook is planned to carry up to a 40 ton load with an operational range of 320 km without refueling. Figure 1.2 shows a virtual scene of a Skyhook carrying a heavy load in remote zone.



**Figure 1.2.** SkyHook heavy lifter vehicle. Courtesy of SkyHook Intl. Inc.

**Aerial refueling** is a task in which an aircraft (tanker) transfers fuel to another aircraft (receiver). This operation is used when an aircraft needs to take off with a greater payload of weapons, cargo, or personnel. It requires a good coordination

between the tanker and the receiver. It is a fact of history that a rescue mission helicopter – UH-60L has made more than 20 attempts to make contact with a tanker with no success. This gives an insight into the difficulty and importance of this type of operation. Figure 1.3 shows a USAF KC-135R Stratotanker, two F-15s and two F-16s, on an aerial refueling operation.



**Figure 1.3.** Aerial refueling of an F-15 Aircraft. Courtesy of the U.S. Air Force

**Spacecraft formation flying** is an important project of the National Aeronautics and Space Administration (NASA) in its search for Earth-like planets. Figure 1.4 shows a virtual image of a scheme of multiple spacecraft formation. A spacecraft formation requires a tighter level of precision, slower displacements, and automated control rather than human control.



**Figure 1.4.** NASA's formation flying for which the levels of precision are much tighter. Courtesy of NASA

## 1.2. Historical background

Man's dream of flying can be traced back to ancient times and illustrated by Daedalus' wings made of feathers and wax in Greek mythology. However, the idea of a device capable of horizontal and/or vertical flight was first developed in China. They made the first steps toward flight around 400 BC with the Chinese "tops", a toy made of feathers at the end of a stick which may be considered as one of the first unmanned aerial vehicles (UAV). A UAV can be defined as an aircraft with no onboard human pilot that can be reused and capable of controlled flight, carrying a payload, etc. The UAV has been a feature of aviation history for many years. The origin of the UAV is closely related to cruise missiles; the main difference is that a UAV has been designed to be used in multiple missions and a cruise missile has been designed to destroy itself along with its target. Therefore, a cruise missile cannot be considered as a UAV, while a UAV can be considered as an evolved form of an almost autonomous aircraft.

### 1.2.1. Aviation history

Throughout time, man-made flying machines have been evolving in many different ways such as balloons, dirigibles, autogyros, helicopters, airplanes, etc. [NEW 04, AIA 09]. A timeline documenting the evolution of aviation is shown in Table 1.1.

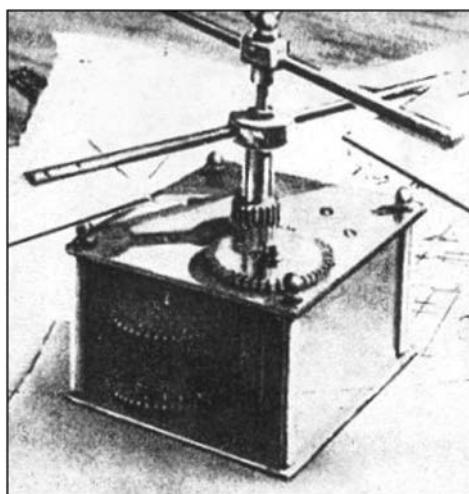
Early in 1754, Mikhail Lomonosov built a mechanical spring-based device, shown in Figure 1.5, capable of vertical takeoff and hover for few moments.

Although, man has been flying for centuries, perhaps the most important advances in aviation started in 1900s when the Wright brothers first successfully flew their glider in 1902 (see Figure 1.6). The Wright brothers' glider was based on the work of Sir George Cayley and other pioneers of 19th Century aviation. Other pioneers of aviation working in parallel were Gustave Whitehead, Samuel P. Langley, Lyman Gilmore, Richard Pearse, among others. Most of the airplanes developed during the 20th Century were based on the successful glider of the Wright brothers.

Another interesting moment in aviation history is the first flight of a manned helicopter, known to have risen from the ground in France in 1907. The Cornu helicopter, shown in Figure 1.7, was an experimental helicopter developed by Paul Cornu, and it was reported to have made several short hops, rising no more than 2 meters.

However, the first successful rotorcraft was not a true helicopter but an autogyro developed by Juan de la Cierva in 1919. Later, Sikorsky introduced several helicopter configurations from the early 1930s to the present (see Figure 1.8).

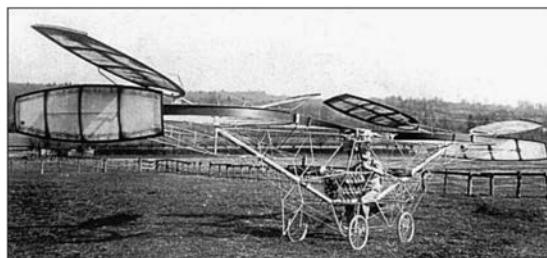
Year	Vehicle name	Designer	Type	Country
400 BC	Chinese "tops"		Helicopter	China
1483	Aerial screw	Leonardo Da Vinci	Helicopter	Italy
1670	Lighter than air	Francesco Lana de Terzi	Balloon	Italy
1709	Lighter than air	Bartolomeu de Gusmao	Balloon	Portugal
1754	Prototype	M. Lomonosov	Helicopter	Russia
1783	Lighter than air	Montgolfier brothers	Balloon	France
1843	Prototype	Sir G. Cayley	Helicopter	UK
1856	L'Albatros artificiel	Jean-Marie Le Bris	Airplane type	France
1861	Prototype	Bright	Helicopter	
1874	Monoplane	Felix du Temple	Airplane	France
1878	Prototype	Forlanini	Helicopter	Italy
1884	Monoplane type	A. Mozhaysky	Airplane	Russia
1890	Eole	Clément Ader	Airplane	France
1902–1905	Glider	Wright brothers	Glider	USA
1907	Hopper	P. Cornu	Helicopter	France
1913	Albatros		Airplane	
1915	MS type L	Morane-Saulnier	Airplane	France
1917	Dr I Triplane	Fokker	Airplane	Germany
1920–1924	Prototype	Pescara	Helicopter	
1923–1935	C1–C30	Juan de la Cierva	Autogyro	Spain
1926–present	Vega, Sirius, etc.	Lockheed	Airplanes	USA
1939–present	S-1, etc.	Sikorsky	Helicopters	Russia

**Table 1.1.** Aircraft evolution**Figure 1.5.** Lomonosov's helicopter. Courtesy of Aviastar [WOR 11]

## 6 Flight Formation Control



**Figure 1.6.** Wright brothers first flight. Courtesy of Wikipedia [WIK 11c]



**Figure 1.7.** Cornu's hopper



**Figure 1.8.** Sikorsky VS-300. Courtesy of Aviastar [WOR 11]

During and after World War I (WWI) there was an explosion in helicopter and airplane development all around the world. A more recent type of aircraft is the tailsitter which is an aircraft capable of vertical takeoff and landing (VTOL) as well as being capable of flying as a classic airplane. After WWII, in 1951, Lockheed and Convair were awarded the contract by the US Army and the US Navy to build the XFY (also referred as “Salmon”) and the XFY (also known as “Pogo”), tailsitters. Figure 1.9 shows a Convair XFY-1 tailsitter. This concept of VTOL was abandoned due to

many design and operational problems, e.g. the pilot had to look over his shoulder to properly stabilize the aircraft for landing. Also, it is considered that the XFV and the XFY VTOLs did not contribute to the development of modern VTOLs. Nowadays, there are many efforts to improve actual designs of helicopters and airplanes, to make them more stable, more reliable, more comfortable, etc.



**Figure 1.9.** Convair XFY "Pogo". Courtesy of Aviastar [WOR 11]

The advent of new technologies and miniaturization have spurred the design and development of manned and unmanned aerial vehicles. Military and civil aviation stepped up the development and production of aircraft and helicopters. In civil applications, man-piloted aircraft systems have been used to transport people and cargo; unmanned aerial vehicles have been used mainly for surveillance. In military applications, UAVs have been used in a wide variety of missions such as target and decoy, reconnaissance, surveillance, etc.

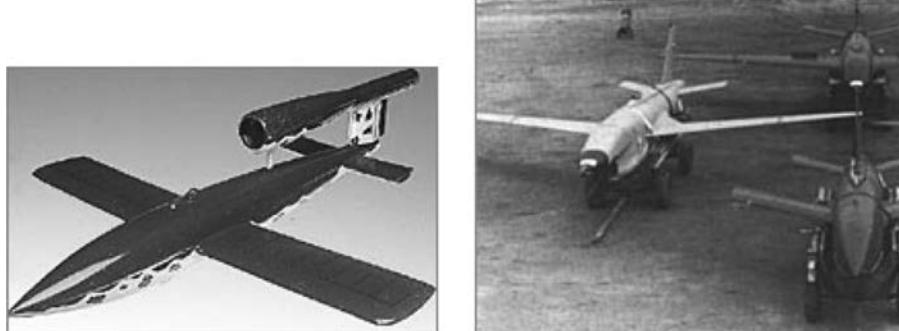
### 1.2.2. Evolution of UAVs

The history of unmanned aerial vehicles began around 1849. On August 22, the Austrians attacked the Italian city of Venice using unmanned balloons loaded with explosives. The next important advance in this domain happened during and after WWI. In November 1917, the US Army started the project to build the Kettering Bug that first flew in 1918 (see Figure 1.10). This unmanned aircraft was intended to be used as an aerial torpedo against Zeppelins.

The first French UAV was designed, built and tested in 1923. In the 1930s, the UK and the US developed the Radioplane OQ-2, a small teleoperated airplane. The German army, in 1938, started the development of a radio-controlled antiship flying bomb. The German V1 unmanned airplanes, shown in Figure 1.11, and the V2 missiles were flying bombs rather than UAVs. However, the V1 wing has been a base model for target drones.



**Figure 1.10.** Kettetring Bug. Courtesy of the National Museum of the US Air Force



**Figure 1.11.** German V1. Courtesy of Wikipedia [WIK 11b]

During the Korean and Vietnam wars, the development of UAVs made important advances. The Ryan Firebee was a well-proven platform for a target drone that led to other missions such as reconnaissance UAV. A modified version of the Ryan Firebee, called the Ryan Model 147 Lightning Bug, was used as a reconnaissance UAV to spy on Vietnam, China, and North Korea in the late 1960s and early 1970s. During the late 1970s and throughout the 1980s, the Israeli Air Force, an aggressive UAV developer, pioneered several important new UAVs that have been integrated into the UAV fleets of many other countries.

In the late 1990s, the American UAV RQ-1A predator, shown in Figure 1.12, offered real-time video imagery without the danger of aircrew losses. The predator RQ-1L was used in the Balkans in 1995, Iraq in 1996, and it proved to be very effective. UAVs have been used especially in risky missions to collect intelligence information. More recently, the trend for battlefield UAVs had been emerging before

the war in Afghanistan that began in 2001. An unmanned aerial system roadmap 2005–2030 has been published in [CAM 05].



**Figure 1.12.** American RQ-1A Predator. Courtesy of Wikipedia [WIK 11a]

### 1.2.3. UAV classification

UAV classification is usually determined by some criteria or features, e.g. use application, range, altitude, endurance, vehicle type, size, etc. We are interested in classifying UAVs due to their configuration as:

- fixed wing;
- rotary wing;
- free wing;
- tilt wing/rotor;
- tailsitter.

Based on this classification, we note that fixed wing conventional or hovering rotary-wing aircraft systems are the most commonly used vehicles. On the one hand, fixed wing conventional aircrafts have proven reliability, long flight time, and cruise efficiency, but they cannot hover or fly at low speeds. On the other hand, hovering platforms have the operational flexibility of being able to take off vertically, hover and land vertically, but they usually have limitations in forward flight, such as low speed and poor endurance. A relatively unexplored configuration is the tailsitter due to the awkward position of the pilot during takeoff, hover, and landing phases.

## 1.3. Flight control

Spacecrafts, aircrafts, and UAVs are dynamic systems that can be classified as underactuated mechanical systems. It is known that an underactuated mechanical

## 10 Flight Formation Control

system has fewer control inputs than degrees of freedom. Thus, the UAV represents an important challenge in automatic control. The UAV flight controller is designed to stabilize the altitude of an aircraft by holding a desired orientation and position. A flight controller also provides the means for an aircraft to navigate by tracking a desired trajectory. Different control techniques have been used to design flight controllers ranging from linear to nonlinear control algorithms.

In [HAU 92], an input–output linearization to stabilize a vertical/short takeoff and landing vehicle has been proposed. An extension and improvement of this work has been made in [MAR 96], in which the main idea was to find a flat output for the system.

In [BEN 96], a comparative analysis between different techniques has been presented. Here, the authors present techniques such as linearization, minimum phase, and sampled methods. Trajectory tracking for a Planar vertical takeoff and landing aircraft (PVTOL) has been also presented.

In [BOU 04], proportional-integral-derivative (PID) and linear-quadratic regulator (LQR) control schemes were used to control a mini rotorcraft with four rotors. A small experimental platform was developed and experimental results are provided. It is noted that the robustness of the control is not guaranteed against uncertainties and/or disturbances. In [BAR 07], a computer-vision-based algorithm is proposed and accomplished using several PID loops for altitude control.

In [MET 02], system identification modeling has been used to develop a parameterized model of a small helicopter. Unmodeled dynamics have been handled using an intuitive approach as in [GAV 01]. Also, robust control techniques have been used to stabilize small helicopters [LAC 03, MAR 02, LIN 99]. In [MET 02], a robust  $H_\infty$  loop-shaping controller has been developed and validated on an experimental helicopter platform performing a robust hover flight. In [MAR 02], an internal-model-based approach for autonomous landing of a VTOL vehicle on an oscillating landing platform on a ship has been presented. An internal-model-based error-feedback regulator has been developed ensuring the global convergence to the zero error manifold and the robustness against uncertainties affecting the system.

In the last decade, UAV altitude stabilization and autonomous hover using bounded input strategies were developed. Several nonlinear saturated flight controllers have been proposed in [FAN 02], [CAS 05], [LOZ 07], and [LOZ 03]. Nested saturations and saturated state techniques have been successfully implemented on real-time platforms to stabilize the PVTOL aircraft and mini rotorcrafts with four rotors.

Nonlinear methods such as sliding modes and backstepping have been proposed in [MAH 04], [OLF 01], [BOU 05], and [ISI 03]. In [ISI 03], a nonlinear adaptive

output regulation and robust stabilization of system in feedforward form has been applied.

#### **1.4. Flight formation control**

Cooperative control and multiple spacecraft formation control have been intensively investigated during the past decades. Multiple spacecraft formation flying has been identified by NASA as an enabling technology for 21st Century missions such as terrestrial planet finding and deep space exploration. Multiple aerial, ground, or underwater vehicles working cooperatively or in coordination have important applications. The applications of multi-autonomous vehicles is currently progressing in multiple fields, e.g. industrial, military, and in the study of biological systems. Missions for these type of systems include exploration and map building, military operations, traffic control, entertainment, biological systems, transport of heavy or large loads, search and rescue operations, surveillance, and aerospace and ocean exploration. In this section, a discussion of the different approaches that have been proposed in the literature for coordinating multiple robot systems is presented. In Scharf's survey [SCH 04], five approaches have been identified for spacecraft formation flying: multiple-input and multiple-output (MIMO), leader/follower, virtual structure, cyclic, and behavioral. In following sections, a state-of-the-art on multiple spacecraft formation flying is discussed.

##### **1.4.1. *Multiple-input and multiple-output***

In the MIMO architecture, the formation problem is treated as a MIMO system where a dynamic model of the formation was used to develop a formation controller. MIMO approaches are described in [LAW 00], [HAD 00], [SMI 02], and [DUN 02]. In [HAD 00], an LQR controller is designed using a minimal state realization of the relative error states. In [DUN 02], a model predictive control was derived to solve the nonlinear and constrained model predictive control (MPC) problem for multiple vehicle formation to a set of equilibria.

##### **1.4.2. *Leader/follower***

In the leader/follower architecture, one agent is designated as leader, while the others are designated as followers that should track the orientation and position of the leader with some offset. Leader/follower approaches are described in [HAD 98], [DES 98], [CHE 06], and [KRI 06]. In [CHE 06], an input-to-state stability (ISS) concept has been used as a tool to develop a formation control. In this approach, saturated controls enforce ISS of the dynamics, thereby avoiding the problem of dealing with locally asymptotically stable zero dynamics. In [HAD 98], an adaptive control strategy was developed considering the presence of constant, but unknown disturbances.

#### **1.4.3. Virtual structure**

The virtual structure approach considers every agent as an element of a larger structure [LEO 01, BEA 99, LAW 99]. Usually, the motion of the virtual structure is done through controlling the individual spacecrafts by tracking their reference trajectories. In [BEA 99], a constellation template was proposed to solve the problem of the coordinated motion of space-based interferometers. A constellation template is a virtual structure that defines the desired position and orientation of each spacecraft within the constellation. In [LAW 99], an adaptive control approach was adopted to design a controller that includes saturation constraints.

#### **1.4.4. Behavior-based control**

The behavioral control in [BAL 98] and [ARR 06] is based on the decomposition of the main control goal into tasks or behaviors. This approach also deals with behaviors such as collision avoidance, flock centering, obstacle avoidance, and barycenter. In [BEA 01], [TAN 03a], [TAN 03b], and [OLF 06], the authors have used algebraic graph theory in order to model the information exchange between vehicles. By using this technique, several control strategies have been developed. In [OLF 06], a coordination control composed of a velocity consensus term, a gradient-based term was proposed. The gradient term helps the cohesion of the group, while the velocity consensus term synchronizes the velocities of the agents. An extension of this approach to include navigational feedback has also been presented in [OLF 06]. The navigational term is used to change the orientation of the group or to move the formation to a given reference position. Ren [REN 07a] presents a new strategy for consensus in multiagent systems with a time-varying reference. Several cases are presented, such as: all agents have access to the reference, several agents have access to the reference, etc. The analysis presented assumes that agent dynamics are represented by a first-order integrator. A state of the art in consensus algorithms can be found in [REN 07b].

#### **1.4.5. Passivity-based control**

In [LEE 03] and [LEE 06], an analysis of multiple agent coordination using a passivity approach to decompose the system into two passive subsystems is presented. The first subsystem, called “shape”, maintains the formation of the group of agents, while the second subsystem, called “lock”, represents the translational dynamics of the group. In [LEE 06], the convergence of velocity and relative position of the agents via passive decomposition is shown. A bilateral teleoperation approach has been used in [HOK 07] to teleoperate a group of agents. The authors provide results to achieve a bilateral teleoperation one-to-many (i.e. one master and many slaves in a leader/follower architecture). The center of mass is used as a virtual master robot

which is used to coordinate the slave robots. Trajectory tracking is also considered using an input to state stability analysis. Consensus algorithms allow the coordination of velocities and/or positions of multiple agents. They have been the object of extensive analysis and development [BEA 01, REN 07b, TAN 03a, TAN 03b]. Trajectory tracking of flocks has recently been studied in [REN 07a] and [HOK 07].

## 1.5. Outline of the book

### **Chapter 2: Theoretical Preliminaries**

In this chapter, some useful results on passivity, graph theory, and robust control are presented. These results will be used through the first half of the book.

### **Chapter 3: Multiagent Coordination Strategies**

In this chapter, a contribution to controllability and observability of multiagent systems is presented. Several approaches to velocity and position forced consensus are presented. It is shown that formation tracking to a time-varying reference can be achieved by using a feedback control based on the center of mass of the multiagent system.

### **Chapter 4: Robust Control Design for Multiagent Systems with Parametric Uncertainty**

In this chapter, we develop an algorithm for robust control design for dynamical systems assuming parametric uncertainty and control input time delay. A robust absolute stability analysis is presented with application to multiagent systems.

### **Chapter 5: On Adaptive and Robust Controlled Synchronization of Networked Robotic Systems on Strongly Connected Graphs**

In this chapter, a controlled synchronization of networked robotic systems communicating on strongly connected graphs is presented. Adaptive and robust tracking control algorithms are utilized to synchronize heterogeneous robotic systems (with dynamic uncertainty) while following a desired trajectory. The robustness of the control algorithms to constant delays in communication is also demonstrated.

### **Chapter 6: Modeling and Control of Mini UAV**

In this chapter, we present the general dynamic model for mini UAVs considering the aerodynamic moments and forces. The dynamic model of two prototypes are developed, a bi-rotor tailsitter and a convertible quadrotor UAV are studied. The main contribution of this chapter is the modeling of two new designs of mini UAV, a tailsitter using variable pitch propellers and a convertible quadrotor using tilting rotors. The stabilization on vertical mode using linear and nonlinear control laws for stabilizing the attitude and position.

### **Chapter 7: Flight Formation Control Strategies for Mini UAVs**

In this chapter, we introduce two approaches to flight formation control such as nested saturation based nonlinear control and high-order consensus nonlinear control.

### **Chapter 8: Formation Based on Potential Functions**

In this chapter, we address a 2D formation control, using simple potential functions that generate the desired forces and a nested saturation controller to move the vehicles to their goal positions.

### **Chapter 9: Quadrotor Vision-Based Control**

In this chapter, a vision-based control scheme for autonomous hovering and trajectory tracking of a miniature quadrotor is presented. Vanishing points techniques are used to estimate the rotation matrix and translation vector of the camera mounted on the quadrotor. These methods have been tested using real images. The analytic results are supported by experimental tests.

### **Chapter 10: Toward Vision-Based Coordination of Quadrotor Platoons**

This chapter presents a vision-based scheme for position coordination of two camera-equipped quadrotors in hover flight. Applying a homography estimation technique, the aircrafts are capable of estimating their relative position with respect to their corresponding target. Simulations and real-time experiments illustrate the performance of this method.

### **Chapter 11: Optimal Guidance for Rotorcraft Platoon Formation Flying in Wind Fields**

In this chapter, a time-optimal guidance for a platoon of rotorcraft flying in formation through a region of strong winds fields is presented. The main goal is to program the heading for the virtual center of mass in such way as to minimize the flight time between two-way points. The heading program is obtained by using a Zermelo navigation approach.

### **Chapter 12: Impact of Wireless Medium Access Protocol on the Quadrotor Formation Control**

This chapter presents an overview of the medium access protocols' impact on the average consensus problem over wireless networks for a group of quadrotors. The analysis considers groups of quadrotors communicating over a wireless network considering both directed and undirected graphs of information flow.

### **Chapter 13: MAC Protocol for Wireless Communications**

This chapter deals with the design of a wireless MAC protocol for UAV communication applications. A discussion on the protocols that define and control

access to the wireless channel is provided. A new protocol based on carrier sense multiple access–code division multiple access (CSMA–CDMA) is presented.

### **Chapter 14: Optimization of a Scannable Pattern for Bidimensional Antenna Arrays to Provide Maximum Performance**

This chapter presents an antenna array design for multirobot systems. The main objective of this chapter is to show the behavior of radiation for the design of antenna arrays in a uniform rectangular and concentric ring geometry, considering the optimization of a scannable pattern in a wide scanning range.

### **1.6. Bibliography**

- [AIA 09] AIAA, *Aerospace America*, Out of the Past, AIAA, 2009.
- [ARR 06] ARRICHIELLO F., CHIAVERINI S., FOSSEN T., “Formation Control of Marine Vessels using the Null-Space-Based Behavioral Control”, *LNCIS 336 Group Coordination and Cooperative Control*, Springer-Verlag, 2006.
- [BAL 98] BALCH T., ARKIN A., “Behavior-based formation control for multirobot teams”, *IEEE Transactions on Robotics and Automation*, vol. 14, no. 6, pp. 926–939, 1998.
- [BAR 07] BARBER D., GRIFFITHS S., McLAIN T., BEARD R., “Autonomous landing of miniature aerial vehicles”, *AIAA Journal of Aerospace computing Information and Communication*, vol. 4, no. 5, pp. 770–784, 2007.
- [BEA 99] BEARD R., HADAEGH F., “Finite thrust control for satellite formation flying with state constraints”, *Proceedings of the IEEE American Control Conference*, San Diego, USA, pp. 4383–4387, 1999.
- [BEA 01] BEARD R., LAWTON J., HADAEGH F., “A coordination architecture for spacecraft formation control”, *IEEE Transactions on Control Systems Technology*, vol. 9, no. 6, pp. 777–790, 2001.
- [BEN 96] BENVENUTI L., DI GIAMBERARDINO P., FARINA L., “Trajectory tracking for PVTOL aircraft: a comparative analysis”, *Proceedings of the 35<sup>th</sup> CDC*, Kobe, Japan, 1996.
- [BOU 04] BOUABDALLAH S., NOTH A., SIEGWART R., “PID vs LQ control techniques applied to an indoor micro quadrotor”, *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, Sendri, Japan, pp. 2451–2456, 2004.
- [BOU 05] BOUABDALLAH S., SIEGWART R., “Backstepping and sliding modes techniques applied to an indoor micro quadrotor”, *Proceedings of the IEEE International Conference on Robotics and Automation*, Barcelona, Spain, pp. 2259–2264, 2005.
- [CAM 05] CAMBONE S., KRIEG K., PACE P., WELLS II L., Unmanned Aerial Systems Roadmap 2005–2030, US Department of Defense, 2005.

## 16 Flight Formation Control

- [CAS 05] CASTILLO P., LOZANO R., DZUL A., *Modelling and Control of Mini-Flying Machines*, Springer-Verlag, 2005.
- [CHE 06] CHEN X., SERRANI A., “ISS-based robust leader/follower trailing control”, *LNCIS 336 Group Coordination and Cooperative Control*, Springer-Verlag, 2006.
- [DES 98] DESAI J., Motion planing and control of formations of nonholonomic mobile robots, PhD Thesis, University of Pennsylvania, 1998.
- [DUN 02] DUNBAR W., MURRAY R., “Model predictive control of coordinated multi-vehicle formations”, *Proceedings of the IEEE Conference on Decision and Control*, pp. 4631–4636, 2002.
- [FAN 02] FANTONI I., LOZANO R., *Nonlinear Control for Underactuated Mechanical Systems*, Communications and Control Engineering Series, Springer-Verlag, 2002.
- [GAV 01] GAVRILETS V., FRAZZOLI E., METTLER B., PIEDMONTE M., FERON E., “Aggresive manouvering of small autonomous helicopters: A human centered approach”, *The International Journal of Robotics Research*, vol. 20, no. 10, pp. 795–807, 2001.
- [HAD 98] HADAEGH F., LU W., WANG P., “Adaptive control of formation flying spacecraft for interferometry”, *IFAC*, Patras, Greece, 1998.
- [HAD 00] HADAEGH F., GHAVIMI A., SINGH G., QUADRELLI M., “A centralized optimal controller for formation flying spacecraft”, *International Conference on Intel Technologies*, Bangkok, Thailand, 2000.
- [HAU 92] HAUSER J., SASTRY S., MEYER G., “Nonlinear control design for slightly nonminimum phase system: application to V/STOL aircraft”, *Automatica*, vol. 28, no. 4, pp. 665–679, 1992.
- [HOK 07] HOKAYEM P., STIPANOVIC D., SPONG M., “Reliable control of multi-agent formations”, *Proceedings of the IEEE American Control Conference*, New York, USA, 2007.
- [ISI 03] ISIDORI A., MARCONI L., SERRANI A., “Robust nonlinear motion control of a helicopter”, *IEEE Transactions on Automatic Control*, vol. 48, no. 3, pp. 413–426, 2003.
- [KRI 06] KRISTIANSEN R., LORÍA A., CHAILLET A., NICKLASSON P., “Output feedback control of relative translation in a leader–follower spacecraft formation”, *LNCIS 336 Group Coordination and Cooperative Control*, Springer-Verlag, 2006.
- [LAC 03] LA CIVITA M., PAPAGEORGIOU G., MESSNER W., KANADE T., “Design and flight testing of a gain-scheduled  $H_\infty$  loop shaping controller for wide-envelope flight of a robotic helicopter”, *Proceedings of the 2003 American Control Conference*, Denver, USA, pp. 4195–4200, 2003.
- [LAW 99] LAWTON J., BEARD R., HADAEGH F., “An adaptive control approach to satellite formation flying with relative distance constraints”, *Proceedings of the IEEE American Control Conference*, pp. 1545–1549, 1999.
- [LAW 00] LAWTON J., A behavior-based approach to multiple spacecraft formation flying, PhD Thesis, Brigham Young University, 2000.
- [LEE 03] LEE D., LI P., “Formation and maneuver control of multiple spacecraft”, *Proceedings of the American Control Conference*, Denver, USA, 2003.

- [LEE 06] LEE D., SPONG M., “Flocking of multiple inertial agents on balanced graph”, *Proceedings of the American Control Conference*, Minneapolis, USA, 2006.
- [LEO 01] LEONARD N., FIORELLI E., “Virtual leaders, artificial potentials and coordinated control of groups”, *Proceedings of the 40<sup>th</sup> IEEE Conference on Decision and Control*, Orlando, USA, 2001.
- [LIN 99] LIN F., ZHANG W., BRANDT R., “Robust hovering control of a PVTOL aircraft”, *IEEE Transactions on Control Systems Technology*, vol. 7, no. 3, pp. 343–351, 1999.
- [LOZ 03] LOZANO R., CASTILLO P., DZUL A., “Global stabilization of the PVTOL: real-time application to a mini-aircraft”, *International Journal of Control*, vol. 76, no. 18, pp. 1833–1844, 2003.
- [LOZ 07] LOZANO R., *Objets volants miniatures: Modélisation et commande embarquée*, Hermès-Lavoisier, 2007.
- [MAH 04] MAHONEY R., HAMEL T., “Robust trajectory tracking for a scale model autonomous helicopter”, *International Journal of Robust and Nonlinear Control*, vol. 14, pp. 1035–1059, 2004.
- [MAR 96] MARTIN P., DEVASIA S., PRADEN B., “A different look at output tracking: Control of a VTOL aircraft”, *Automatica*, vol. 32, no. 1, pp. 101–107, 1996.
- [MAR 02] MARCONI L., ISIDORI A., SERRANI A., “Autonomous vertical landing on an oscillating platform: an internal-model based approach”, *Automatica*, vol. 38, no. 1, pp. 21–32, 2002.
- [MET 02] METTLER B., TISHCLER M., KANADE T., “System identification modeling of a small-scale unmanned rotorcraft for flight control design”, *Journal of American Helicopter Society*, vol. 47, pp. 50–63, 2002.
- [NEW 04] NEWCOME L., *Unmanned Aviation: A Brief History of Unmanned Aerial Vehicles*, AIAA, 2004.
- [OLF 01] OLFATI-SABER R., Nonlinear control of underactuated mechanical systems with application to robotics and aerospace vehicles, PhD Thesis, Department of Electrical Engineering and Computer Science, MIT, February 2001.
- [OLF 06] OLFATI-SABER R., “Flocking for Multi-Agent Dynamic Systems: Algorithms and Theory”, *IEEE Transactions on Automatic Control*, vol. 51, no. 3, 2006.
- [REN 07a] REN W., “Consensus seeking in multi-vehicle systems with a time varying reference state”, *Proceedings of the 2007 American Control Conference*, New York, USA, 2007.
- [REN 07b] REN W., BEARD R., ATKINS E., “Information consensus in multivehicle cooperative control”, *IEEE Control Systems Magazine*, vol. 27, no. 2, pp. 71–82, 2007.
- [SCH 04] SCHARFT D., HADAEGH F., PLOEN S., “A survey of spacecraft formation flying guidance and control. Part II: Control”, *Proceedings of the IEEE American Control Conference*, Los Angeles, USA, pp. 2976–2985, 2004.
- [SMI 02] SMITH R., HADAEGH F., “Control topologies for deep space formation flying spacecraft”, *Proceedings of the IEEE American Control Conference*, Anchorage, USA, 2002.

## 18 Flight Formation Control

- [TAN 03a] TANNER H., JADBABAIE A., PAPPAS G., “Stable flocking of mobile agents, part I: fixed topology”, *Proceedings of the 42nd IEEE Conference on Decision and Control*, Maui, USA, 2003.
- [TAN 03b] TANNER H., JADBABAIE A., PAPPAS G., “Stable flocking of mobile agents, part II: dynamic topology”, *Proceedings of the 42nd IEEE Conference on Decision and Control*, Maui, USA, 2003.
- [WIK 11a] WIKIPEDIA, RQ-1 Predator, [http://fr.wikipedia.org/wiki/RQ-1\\_Predator](http://fr.wikipedia.org/wiki/RQ-1_Predator), September 2011.
- [WIK 11b] WIKIPEDIA, V-1 flying bomb, [http://en.wikipedia.org/wiki/V-1\\_flying\\_bomb](http://en.wikipedia.org/wiki/V-1_flying_bomb), September 2011.
- [WIK 11c] WIKIPEDIA, Wright brothers, [http://en.wikipedia.org/wiki/Wright\\_brothers](http://en.wikipedia.org/wiki/Wright_brothers), September 2011.
- [WOR 11] THE WORLD’S ROTORCRAFT A., Sikorsky VS-300, <http://www.aviastar.org>, September 2011.

## Chapter 2

# Theoretical Preliminaries

Most of the papers in the literature dealing with multiple agent coordination consider fully actuated agents capable of movement in all directions.

In the following sections, we provide a background on passivity-based control, graph theory, and parametric uncertainty that we will use throughout this book.

### 2.1. Passivity

Consider a dynamical system represented by the state space model:

$$\dot{x} = f(x, u) \quad [2.1]$$

$$y = h(x, u) \quad [2.2]$$

where  $f$  is locally Lipschitz,  $h$  is continuous, and the system has the same number of inputs and outputs.

In this thesis, we limit ourselves to control affine passive systems. To set the background for what follows, consider the following affine nonlinear system:

$$\begin{aligned} \dot{x} &= f(x) + g(x)u \\ \Sigma \quad y &= h(x) \end{aligned} \quad [2.3]$$

where  $x \in \mathbb{R}^n$ ,  $u \in \mathbb{R}^m$ , and  $y \in \mathbb{R}^m$ . The functions  $f(\cdot) \in \mathbb{R}^n$ ,  $g(\cdot) \in \mathbb{R}^{n \times m}$ , and  $h(\cdot) \in \mathbb{R}^m$  are assumed to be sufficiently smooth. It is assumed, for simplicity, that  $f(0) = 0$  and  $h(0) = 0$ . Now, we introduce the following result [BRO 00]:

---

Chapter written by J.A. GUERRERO.

LEMMA 2.1.– Consider the nonlinear system  $\Sigma$ . The following statements are equivalent.

– There exists a  $C^1$  storage function  $V(x) \geq 0$ ,  $V(0) = 0$  and a function  $\mathcal{S}(x) \geq 0$  such that for all  $t \geq 0$ :

$$V(x(t)) - V(x(0)) = \int_0^t y^T(s)u(s)ds - \int_0^t \mathcal{S}(x(s))ds \quad [2.4]$$

The system is strictly passive for  $\mathcal{S}(x) > 0$ , passive for  $\mathcal{S}(x) \geq 0$  and lossless for  $\mathcal{S}(x) = 0$ .

– There exists a  $C^1$  non-negative function  $V: x \rightarrow \mathbb{R}$  with  $V(0) = 0$ , such that

$$L_f V(x) = -\mathcal{S}(x) \quad [2.5]$$

$$L_g V(x) = h^T(x) \quad [2.6]$$

where  $L_g V(x) = \partial V(x)/\partial x g(x)$ .

## 2.2. Graph theory

A natural way to analyze the relationship and communication between agents is using either directed or undirected graphs. A multiagent dynamic system can be modeled as a group of dynamical systems which has an information exchange topology represented by information graphs. Then, we introduce some basic definitions and results from graph theory [GOD 07].

DEFINITION 2.1.– A graph  $\mathcal{G}$  is a pair  $\mathcal{G}(\mathcal{N}, \mathcal{E})$  consisting of a set of nodes  $\mathcal{N} = \{n_i : n_i \in \mathcal{N}, \forall i = 1, \dots, n\}$  together with their interconnections  $\mathcal{E}$  on  $\mathcal{N}$ . Each pair  $(n_1, n_2)$  is called an edge  $e \in \mathcal{E}$

If a pair  $(n_1, n_2)$  is an edge then it is said that  $n_2$  is a neighbor of  $n_1$ . We use directed graphs or digraphs in order to model asymmetric relations between agents or nodes. A digraph consists of a node set  $\mathcal{N}$  and an edge set  $\mathcal{E}$  where every directed edge is an ordered pair of distinct nodes, i.e. the  $i$ th node can get information from the  $j$ th node but not necessarily vice versa. Directed edges are represented by arrows indicating the direction of the edge. An undirected graph is a graph where the  $i$ th and the  $j$ th nodes can get information from each other. In this case, a mechanical approach can be adopted since every node (agent) in the graph can be considered as a mass  $M$  and the edges can be considered as springs and dampers when states and its first derivative are shared with their neighbors. A more complicated problem is to consider unidirectional communication between agents. We then introduce some important definitions on graph theory.

DEFINITION 2.2.– A subgraph  $\mathcal{G}^\circledast$  of a graph  $\mathcal{G}$  is such that  $\mathcal{N}(\mathcal{G}^\circledast) \subseteq \mathcal{N}(\mathcal{G})$  and  $\mathcal{E}(\mathcal{G}^\circledast) \subseteq \mathcal{E}(\mathcal{G})$ .

When  $\mathcal{N}(\mathcal{G}^{\otimes}) = \mathcal{N}(\mathcal{G})$ , it is said that  $\mathcal{G}^{\otimes}$  is a spanning graph of  $\mathcal{G}$ .

The valency of a node  $n_i$  is the number of neighbors of  $n_i$ . When working with directed graphs, the in-valency of  $n_i$  is the number of edges ending on  $n_i$  and the out-valency is defined analogously.

**DEFINITION 2.3.–** *Let  $\mathcal{G}$  be a directed graph, it is said to be balanced, if its in-valency (number of communication links arriving at the node) is equal to its out-valency (number of communication links leaving the node) for all  $e \in \mathcal{E}$ .*

A path in a directed graph is a sequence  $n_1, \dots, n_r + 1$  of  $r$  distinct nodes starting at  $n_1$  and ending on  $n_r + 1$  such that consecutive nodes are neighbors, i.e. for every  $i \in \{1, \dots, r+1\}$ ,  $(n_i, n_{i+1})$  is an edge. A weak path is a sequence of  $n_1, \dots, n_r + 1$  of distinct nodes such that for  $i = 1, \dots, r$ , either  $(n_i, n_{i+1})$  or  $(n_{i+1}, n_i)$  is an edge.

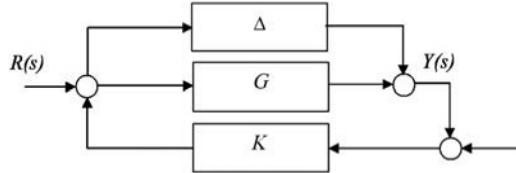
A graph is connected if for every pair  $(n_1, n_2)$  of distinct vertices there is a path from  $n_1$  to  $n_2$ . A connected graph allows the communication between all agents through the network. A digraph is said to be strongly connected if any two nodes can be joined by a path and is weakly connected if any two nodes can be joined by a weak path.

A cycle is a graph where every node has exactly two neighbors. Then, a cycle in a graph is a subgraph that is a cycle. Similarly, an acyclic graph is a graph with no cycles. A connected acyclic graph is called a tree and a spanning subgraph with no cycles is called a spanning tree.

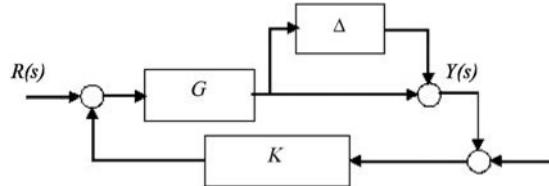
### 2.3. Robustness problems

Many modern control problems involve uncertainty which refers to the difference between a real system and the model that describes its behavior. The robust control problem appeared with the objective of reducing the difference between the dynamic behavior of real systems and their mathematical models. Robust control considers diverse approaches; however, for linear time-invariant systems, there are two fundamental approaches: the robust control considering dynamic uncertainty and the robust control considering parametric uncertainty. Looking back at the literature, control of dynamic systems involving dynamic or parametric uncertainty has been studied in recent decades. Much of the work dealing with dynamic uncertainty can be found in [HEL 98], [ZHO 96], and [GRE 95]. Also, much of the work on control of dynamic systems considering parametric uncertainty can be found in [BAR 94] and [ACK 93]. Although we are interested in dynamic systems considering parametric uncertainty, we first give a brief introduction to dynamic systems involving dynamic uncertainty.

– *Dynamic uncertainty*: For dynamical systems involving dynamic uncertainty represented in input–output form, the analysis is done in the frequency domain. There are two important approaches for the analysis of this type of system: the analysis of dynamical systems considering additive uncertainty and the analysis of dynamical system involving multiplicative uncertainty. These approaches are shown in Figures 2.1 and 2.2, respectively,



**Figure 2.1.** Dynamical system with additive uncertainty



**Figure 2.2.** Dynamical system with multiplicative uncertainty

where  $G$  represents the mathematical model of the physical process,  $K$  is the controller acting on the process,  $\Delta$  represents the dynamic uncertainty, and  $U(s)$  and  $Y(s)$  represent the system's input and output, respectively.

Generally, when dynamic uncertainty is considered, the problem statement is as follows: to find a controller  $K$  that satisfies certain performance specifications for the family of plants obtained from considering the uncertainty  $\Delta$  in the system as well as to optimize in the sense that it guarantees the desired performance for the biggest possible uncertainty, measured in function of its infinity norm. This problem is known as  $H^\infty$  control problem (see [ROM 97] and [GRE 95]).

– *Parametric uncertainty*: The uncertainty in a dynamical system can also be expressed as uncertainty in the system parameters that define its structure as follows:

$$\dot{x} = A(\mathbf{q})x - b(\mathbf{q})\psi(y) \quad [2.7]$$

$$y = c(\mathbf{q})^T x$$

where the uncertainty is considered in the matrices  $A(\mathbf{q})$ ,  $b(\mathbf{q})$ , and  $c(\mathbf{q})$ .

For dynamic systems involving parametric uncertainty, the analysis can also be done in the frequency domain. In this case, the analysis consists of determining

polynomial function properties which can be obtained from the characteristic equation of the dynamical system represented in input–output form. The robust stability property is determined in the function of the family of polynomials expressed in the following form:

$$p(s, \mathbf{q}) = \sum_{i=0}^n a_i(\mathbf{q}) s^i \quad [2.8]$$

where  $\mathbf{q} \in \mathbb{R}^n$  is the vector of uncertainty parameters in the dynamical system, which is reflected in the coefficients of the polynomial through  $a_i(\mathbf{q})$ . The parametric uncertainty is classified in diverse forms. In the following section, we will describe the different classes of parametric uncertainty.

### 2.3.1. Representation of the parametric uncertainty

Parametric uncertainty on a dynamical system is represented as a vector,  $\mathbf{q}$ , in which every element represents a parameter whose nominal value is considered uncertain. The vector of uncertainty is delimited by the uncertainty conditions such that it forms a region in a vectorial space. There are three types of parametric uncertainty: box, sphere, and diamond [ROM 97].

**DEFINITION 2.4.–** *The uncertainty type “box” is defined as:*

$$\mathcal{Q} = \{\mathbf{q} \in \mathbb{R}^l : q_i^- \leq q_i \leq q_i^+ \quad \forall i = 1, 2, \dots, l\} \quad [2.9]$$

**DEFINITION 2.5.–** *Uncertainty type “sphere” is defined as:*

$$\mathcal{Q} = \{\mathbf{q} \in \mathbb{R}^l : \|\mathbf{q}\|_2 \leq r\} \quad [2.10]$$

**DEFINITION 2.6.–** *Uncertainty type “diamond” is defined as:*

$$\mathcal{Q} = \{\mathbf{q} \in \mathbb{R}^l : \|\mathbf{q}\|_1 \leq r\} \quad [2.11]$$

We are interested in the analysis of dynamical systems involving box-type parametric uncertainty. Depending on the quantity of uncertain parameters, we call uncertainty box a region formed by two uncertain parameters, we call uncertainty cube a region formed by three uncertain parameters, and finally, we call uncertainty hypercube a region formed by more than three uncertain parameters.

Generally, uncertainty is defined as a set and is important for this set to be a connected set such that we are able to apply the robust stability results that will be presented later in this work. The definition of a connected set is as follows.

**DEFINITION 2.7.** – A set  $\mathcal{Q} \subseteq \mathbb{R}^n$  is said to be connected, if the following condition is satisfied: given any two points  $q_0, q_1 \in \mathcal{Q}$ , there exists a continuous function  $\Phi: [0, 1] \rightarrow \mathcal{Q}$  such that  $\Phi(0) = q_0$  and  $\Phi(1) = q_1$ .

This definition can be graphically represented by Figure 2.3.

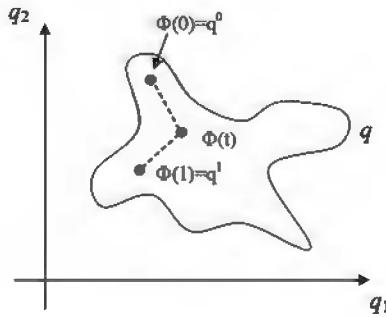


Figure 2.3. Connected set in  $\mathbb{R}^2$

### 2.3.2. Families of polynomials

The union of an uncertain polynomial and its set of uncertainty is called family of polynomials, which is denoted by:

$$\mathcal{P} = \{p(s, \mathbf{q}) : \mathbf{q} \in \mathcal{Q}\}$$

where  $p(s, \mathbf{q})$  is the uncertain polynomial and  $\mathcal{Q}$  is the set of all possible values that the uncertainty can take, usually called *uncertainty box*.

Basically, a family of polynomials is the union of an uncertain polynomial and its uncertainty box that results in a set of polynomials with the same structure called family of polynomials. This family of polynomials can be associated with the stability property as follows [BAR 94].

**DEFINITION 2.8.** – A family of polynomials  $\mathcal{P} = \{p(\cdot, \mathbf{q}) : \mathbf{q} \in \mathcal{Q}\}$  is said to be robustly stable if, for all  $\mathbf{q} \in \mathcal{Q}$ ,  $p(s, \mathbf{q})$  is stable, i.e. for all  $\mathbf{q} \in \mathcal{Q}$ , all the roots of  $p(s, \mathbf{q})$  are placed on left half plane of the complex plane.

The previous definition implies that, if we have a family of polynomials, we have to make all the possible combinations of the uncertain polynomial with the uncertainty to show that every element of the family of polynomials has its roots in the left half plane of the complex plane. This means that we have to evaluate an infinite number of polynomials to show that the family of polynomials is robustly stable. It is impossible to evaluate an infinite number of polynomials; hence, Kharitonov's

result, see [KHA 79], allows us to evaluate the robust stability property of a family of polynomials by evaluating four uncertain polynomials that are created based on the Kharitonov's theorem. In [KHA 79], it is possible to find the demonstration of the theorem as well as the procedure to evaluate families of polynomials; however, Kharitonov's theorem can be used only on families of interval polynomials which are known to have independent uncertainty structure. However, the uncertainty structure in typical applications is more complicated than the independent uncertainty structure. Then, the hierarchy of the structure of uncertainty is as follows:

$$\mathcal{P}_{ind} \subset \mathcal{P}_{af} \subset \mathcal{P}_{mul} \subset \mathcal{P}_{pol}$$

where  $\mathcal{P}_{ind}$  = independent uncertainty structure,  $\mathcal{P}_{af}$  = affine uncertainty structure,  $\mathcal{P}_{mul}$  = multilinear uncertainty structure, and  $\mathcal{P}_{pol}$  = polynomic uncertainty structure.

In [BAR 94], it is possible to find the definitions of the different uncertainty structures. In this section, we include the definition of the polynomic uncertainty structure because this type of uncertainty is considered in our problem, and it is defined as follows.

**DEFINITION 2.9.–** *It is said that  $p(s, \mathbf{q}) = \sum_{i=1}^n a_i(\mathbf{q})s^i$  has polynomic uncertainty structure if each coefficient function  $a_i(\mathbf{q})$  is a multivariable polynomial in the components of  $\mathbf{q}$ .*

## 2.4. Bibliography

- [ACK 93] ACKERMANN J., *Robust Control*, Springer-Verlag, 1993.
- [BAR 94] BARMISH B., *New Tools for Robustness of Linear Systems*, Macmillan, 1994.
- [BRO 00] BROGLIATO B., LOZANO R., MASCHKE B., EGELAND O., *Dissipative Systems Analysis and Control*, Springer, 2000.
- [GOD 07] GODSIL C., ROYLE G., *Algebraic Graph Theory*, Springer, 2007.
- [GRE 95] GREEN M., LINEBEER D., *Linear Robust Control*, Prentice Hall, 1995.
- [HEL 98] HELTON J. W., MERINO O., *Classical Control Using  $H^\infty$  Methods*, SIAM, 1998.
- [KHA 79] KHARITONOV V., “Asymptotic stability of an equilibrium point position of a family of systems of linear differential equations”, *Differential Equations*, vol. 14, pp. 1483–1485, 1979.
- [ROM 97] ROMERO G., Análisis de Estabilidad Robusta para Sistemas Dinámicos con Retardo, PhD Thesis, FIME-UANL, June 1997.
- [ZHO 96] ZHOU K., DOYLE J., GLOVER K., *Robust and Optimal Control*, Prentice Hall, 1996.



## Chapter 3

# Multiagent Coordination Strategies

This chapter introduces an analysis of the controllability and observability properties of several interconnection configurations for multiagent systems such as the chain topology and cyclic topology, as well as the combinations of these two topologies. A leader/follower control strategy is proposed to control the center of mass of the multiple agent (multiagent) system. We show that the trajectory tracking for a multiagent system converges to the constant input reference given only to the leader. Also, it is shown that choosing an appropriate gain, the agents achieve consensus for constant input reference. It is assumed that the  $i$ th agent can be represented as a passive system in the control affine form introduced in Chapter 2 with radially unbounded  $\mathcal{C}^1$  positive-definite storage function  $V_i$ .

The remainder of this chapter is organized as follows. In section 3.1, we present results on controllability and observability properties of multiagent systems. We show that the center of mass of every multiagent system with balanced information-exchange graph corresponds to controllable and observable modes. In section 3.3, we demonstrate trajectory-tracking convergence for leader-based multiagent systems. We use the case where the leader is not assumed to have direct information from all the agents, the state is observed from the input and output of the leader. We therefore require observability and controllability (or at least detectability and stabilizability) of the agents' network from the leader input and output. Also, a high-order consensus control scheme is presented.

---

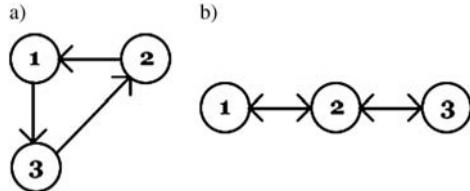
Chapter written by J.A. GUERRERO, R. LOZANO, M.W. SPONG and N. CHOPRA.

### 3.1. Introduction

A natural way to analyze the relationship and communication between agents (or autonomous vehicles, e.g. unmanned aerial vehicles (UAVs)) is by using directed or undirected graphs. Every node in a graph is considered as an agent, which can have information exchange with all or several agents. Multiagent consensus algorithms allow velocity and/or position coordination. A state-of-the-art in consensus algorithms can be found in [REN 07].

We are interested in a passive approach for multiple vehicle coordination and trajectory-tracking control. We propose a control strategy composed of two terms. The first control term is used for agent consensus and the second control term is used for achieving a desired position in the formation, center of mass. We use the output synchronization control scheme in [CHO 06] for agent coordination. This type of strategy is such that the control input for each agent depends only on the information coming from its neighbors. Tracking of the center of mass is achieved by using a full state feedback control on the leader.

Information from all agents is necessary when working with leader-based control schemes. Therefore, observability and controllability from the input and output of the leader are required. Thus, observability and controllability of multiagent formations are studied for the case of cyclic and chain topologies using a coordinating controller. Controllability and observability of leader-based multiagent coordination have been studied in [JI 06], [RAH 06], and [RAH 07]. Here, the authors provide sufficient conditions for multiagent coordination controllability and observability using multiple leaders, and show how interconnection graph topology determines controllability and observability properties.



**Figure 3.1.** Information flow configuration: (a) cyclic topology (left), (b) chain topology (right)

### 3.2. Controllability and observability of interconnections

We will first consider the case of three agents and study the controllability and observability of the system using only the input and output of a single agent. It is

shown that for the cyclic topology the system is observable and controllable from any agent. It is also shown that for the chain topology with undirected communication, the system is controllable and observable for agents 1 and 3, but only stabilizable and detectable for agent 2.

### 3.2.1. Cyclic topology

In the case of three agents, the cyclic topology with input and output on the first agent, is represented as

$$\begin{bmatrix} \ddot{x}_1 \\ \ddot{x}_2 \\ \ddot{x}_3 \end{bmatrix} = \begin{bmatrix} -1 & 1 & 0 \\ 0 & -1 & 1 \\ 1 & 0 & -1 \end{bmatrix} \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} u_1 \quad [3.1]$$

$$y_1 = [1 \ 0 \ 0] \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix} \quad [3.2]$$

Let  $\mathcal{O}$  and  $\mathcal{C}$  be, respectively, the observability and controllability matrices. It is clear that the system is observable since  $\det \mathcal{O} = 1$ . Also the system is controllable since  $\det \mathcal{C} = -1$ .

### 3.2.2. Chain topology: input and output on agent 1

In the chain topology with input and output of the first agent, the system is represented by

$$\begin{bmatrix} \ddot{x}_1 \\ \ddot{x}_2 \\ \ddot{x}_3 \end{bmatrix} = \begin{bmatrix} -1 & 1 & 0 \\ 1 & -2 & 1 \\ 0 & 1 & -1 \end{bmatrix} \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} u_1 \quad [3.3]$$

$$y_1 = [1 \ 0 \ 0] \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix}$$

The system is observable since  $\det \mathcal{O} = 1$ . The system is also controllable since  $\det \mathcal{C} = 1$ . Due to symmetry, the same result holds for agent 3.

### 3.2.3. Chain topology: input and output on agent 2

Let us consider the same chain configuration as before but considering this time the input and output of agent 2

$$\begin{bmatrix} \ddot{x}_1 \\ \ddot{x}_2 \\ \ddot{x}_3 \end{bmatrix} = \begin{bmatrix} -1 & 1 & 0 \\ 1 & -2 & 1 \\ 0 & 1 & -1 \end{bmatrix} \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} u_2 \quad [3.4]$$

$$y_1 = [0 \ 1 \ 0] \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix} \quad [3.5]$$

The system is not observable since  $\det \mathcal{O} = 0$ . Note, however, that we assumed that agent 2 measures all the state. The system is not controllable since  $\det \mathfrak{C} = 0$ .

### 3.2.4. Eigenvalues and eigenvectors of the system

Consider the case of three agents in the chain topology described in [3.3]. The eigenvalues of  $-\mathcal{L}$  in [3.3] are 0, -1, and -3. Their corresponding eigenvectors are

$$\begin{aligned} \mathbf{w}_1^T &= [1 \ 1 \ 1] \\ \mathbf{w}_2^T &= [1 \ 0 \ -1] \\ \mathbf{w}_3^T &= [1 \ -2 \ 1] \end{aligned} \quad [3.6]$$

Premultiplying [3.4] by  $\mathbf{w}_1^T$ ,  $\mathbf{w}_2^T$ , and  $\mathbf{w}_3^T$ , we obtain

$$\begin{aligned} \ddot{x}_1 + \ddot{x}_2 + \ddot{x}_3 &= u_2 \\ \ddot{x}_1 - 2\ddot{x}_2 + \ddot{x}_3 &= -3(\dot{x}_1 - 2\dot{x}_2 + \dot{x}_3) + u_2 \\ \ddot{x}_1 - \ddot{x}_3 &= -(\dot{x}_1 - \dot{x}_3) \end{aligned}$$

Equation [3.4] can also be rewritten as

$$\begin{aligned} \ddot{x}_2 &= -2\dot{x}_2 + (\dot{x}_1 + \dot{x}_3) + u_2 \\ \ddot{x}_1 + \ddot{x}_3 &= 2\dot{x}_2 - (\dot{x}_1 + \dot{x}_3) \\ \ddot{x}_1 - \ddot{x}_3 &= -(\dot{x}_1 - \dot{x}_3) \end{aligned} \quad [3.7]$$

The last equation represents a stable uncontrollable mode. It follows that  $\dot{x}_1 - \dot{x}_3 = e(0) \exp^{-t}$ , with  $e(0) = \dot{x}_1(0) - \dot{x}_3(0)$ . Thus, defining  $\xi = \dot{x}_1 + \dot{x}_3$ , [3.7] can be reduced to:

$$\begin{aligned} \ddot{x}_2 &= -2\dot{x}_2 + \xi + u_2 \\ \dot{\xi} &= 2\dot{x}_2 - \xi \end{aligned}$$

The reduced system is controllable since  $\det \mathbf{C} = 2$ . Also, the reduced system is observable since  $\det \mathcal{O} = 1$ .

### 3.2.5. General case

The examples given above suggest that the interconnection of agents using the coordinating control strategy leads to systems that are controllable and observable from the input and output of every agent, or that they are at least stabilizable and detectable. We will prove next that this is true in the general case when the system is represented by

$$\begin{aligned}\ddot{\mathbf{x}} &= -\mathcal{L}\dot{\mathbf{x}} + \mathbf{b}\mathbf{u} \\ y &= \mathbf{c}^T \dot{\mathbf{x}}\end{aligned}\quad [3.8]$$

where  $\mathcal{L}$  is the Laplacian matrix having the following properties:

- 1)  $\mathcal{L}$  has a single eigenvalue at 0,  $\lambda_1(\mathcal{L}) = 0$  with right eigenvector  $\mathbf{w}_1^T = [1 \ 1 \ \dots \ 1]$ , i.e.  $\mathcal{L}\mathbf{w}_1 = 0$ .
- 2) The remaining eigenvalues have all positive real part, i.e.  $\text{Re}[\lambda_i(\mathcal{L})] > 0$  and  $\mathcal{L}\mathbf{w}_i = \lambda_i \mathbf{w}_i$ , for  $i = 2, \dots, n$ , and  $\mathbf{w}_i \in \mathbb{R}^n$ . If  $\mathcal{L}$  is symmetric then the remaining eigenvalues are all positive, i.e.  $\lambda_i(\mathcal{L}) > 0$ .

We assume that the information exchange graph is balanced. Let us assume also that in the coordinating controller the gains multiplying the signals in between agents are all equal to 1. For the  $i$ th row of  $\mathcal{L}$ , the entries  $l_{ij} = -1$ , for  $i \neq j$ , correspond to the gains multiplying the signals from other agents coming to agent  $i$ . For the  $i$ th column of  $\mathcal{L}$ , the entries  $l_{ji} = -1$ , for  $i \neq j$ , correspond to the gains multiplying the signals going out of agent  $i$  toward the other agents. We then have the following property.

- 3)  $\mathbf{w}_1$  defined above is also the left eigenvalue of  $\mathcal{L}$  corresponding to the eigenvalue 0, i.e.  $\mathbf{w}_1^T \mathcal{L} = 0$ .

Let us study the controllability and observability of the system from the input and output of agent  $k$ , i.e.  $\mathbf{c} = \mathbf{b}^T = [0 \ \dots \ 0 \ 1 \ 0 \ \dots \ 0]$  where only the  $k$ th element is different from zero.

**PROPOSITION 3.1.** – Consider the multiple agent system whose evolution is described by [3.8]. This system is unobservable if there exists a right eigenvector  $\mathbf{w}_i$  of  $\mathcal{L}$  such that  $\mathbf{c}^T \mathbf{w}_i = 0$ .

**PROOF.** – Let  $\mathbf{w}_1$  be the right eigenvector of  $\mathcal{L}$  corresponding to the zero eigenvalue. Note that  $\mathbf{c}^T \mathbf{w}_1 \neq 0$ , and thus, the mode corresponding to  $(\lambda_1, \mathbf{w}_1)$  is observable. If  $\mathbf{c}^T \mathbf{w}_i = 0$  for some  $i = 2, \dots, n$ , then the system will have non-observable modes,

but such modes will be asymptotically stable, i.e. they will converge to zero, because  $\text{Re}[\lambda_i(\mathcal{L})] > 0$ , for  $i = 2, \dots, n$ .

**PROPOSITION 3.2.**— *Consider the multiple agent system whose evolution is described by [3.8]. This system is uncontrollable if there exists an eigenvector  $\mathbf{v}_i$  of  $\mathcal{L}^T$  such that  $\mathbf{v}_i^T \mathbf{b} = 0$ .*

**PROOF.**— Note that  $\mathcal{L}$  and  $\mathcal{L}^T$  have the same eigenvalues. Let  $\mathbf{v}_i$  be the right eigenvectors of  $\mathcal{L}^T$  (or the left eigenvectors of  $\mathcal{L}$ ), i.e.  $\mathcal{L}^T \mathbf{v}_i = \lambda_i \mathbf{v}_i$  or  $\mathbf{v}_i^T \mathcal{L} = \lambda_i \mathbf{v}_i^T$ . Premultiplying [3.8] by  $\mathbf{v}_i^T$ , we get

$$\mathbf{v}_i^T \ddot{\mathbf{x}} = -\mathbf{v}_i^T \mathcal{L} \dot{\mathbf{x}} + \mathbf{v}_i^T \mathbf{b} \mathbf{u}$$

or

$$\mathbf{v}_i^T \ddot{\mathbf{x}} = -\lambda_i \mathbf{v}_i^T \dot{\mathbf{x}} + \mathbf{v}_i^T \mathbf{b} \mathbf{u} \quad [3.9]$$

If  $\mathbf{v}_i^T \mathbf{b} = 0$  for some  $i$ , then

$$\frac{d}{dt} (\mathbf{v}_i^T \dot{\mathbf{x}}) = -\lambda_i (\mathbf{v}_i^T \dot{\mathbf{x}})$$

which means that the corresponding mode is not controllable. Nevertheless, for  $i = 2, \dots, n$ , such mode is asymptotically stable and converges to zero. Recall that for  $i = 1$ , we have  $\mathbf{v}_1 = \mathbf{w}_1$ , and therefore,  $\mathbf{v}_1^T \mathbf{b} \neq 0$ . Thus, the mode corresponding to  $(\lambda_1, \mathbf{v}_1)$  is controllable.

As a consequence of the above propositions, it is possible to present the following lemma.

**LEMMA 3.1.**— *The center of mass of multiagent system [3.8] corresponds to controllable and observable modes.*

### 3.2.6. The cyclic topology in the general case

In this section, it will be proved that the cyclic topology is controllable and observable in the general case. The Laplacian matrix of the cyclic topology is as follows:

$$-\mathcal{L}_r = \begin{bmatrix} -1 & 1 & & & \\ & -1 & 1 & & \\ & & \ddots & \ddots & \\ & & & -1 & 1 \\ 1 & & & & -1 \end{bmatrix} \quad [3.10]$$

and  $\mathbf{c} = \mathbf{b}^T = [0 \ \cdots \ 1 \ \cdots \ 0]$ , where only the  $k$ th element is different from zero. For simplicity, this system will be rewritten as

$$\ddot{\mathbf{x}} = -\mathcal{L}_r \dot{\mathbf{x}} + \mathbf{b} \mathbf{u}_l$$

$$y_l = \mathbf{c}^T \dot{\mathbf{x}}$$

### 3.2.6.1. Observability

The system will be non-observable if there exists a vector  $v$  such that

$$\mathbf{v}^T \mathbf{b} = 0 \quad [3.11]$$

and

$$\mathcal{L}_r \mathbf{v} = \boldsymbol{\lambda} \mathbf{v} \quad [3.12]$$

with  $\mathbf{v}^T = [v_1 \ v_2 \ \cdots \ v_n]$  and a real  $\boldsymbol{\lambda}$ .

For  $\boldsymbol{\lambda} = 0$ ,  $\mathbf{v}^T = [1 \ 1 \ \cdots \ 1]$ , and thus,  $\mathbf{v}^T \mathbf{b} \neq 0$ , which is a contradiction.

For  $\boldsymbol{\lambda} \neq 0$ , it follows from [3.11] that  $v_1 = 0$ . In view of the structure of  $\mathcal{L}_r$  in [3.12] it follows that  $v_2 = 0$ . Iterating it follows that  $\mathbf{v} = 0$ . We conclude that the system is observable.

### 3.2.6.2. Controllability

The system will be non-controllable if there exists a vector  $v$  such that

$$\mathbf{v}^T \mathbf{b} = 0 \quad [3.13]$$

and

$$\mathbf{v}^T \mathcal{L}_r = \boldsymbol{\lambda} \mathbf{v}^T \quad [3.14]$$

with  $\mathbf{v}^T = [\mathbf{v}_1 \ \mathbf{v}_2 \ \cdots \ \mathbf{v}_n]$  and a real  $\boldsymbol{\lambda}$ .

For  $\boldsymbol{\lambda} = 0$ , it follows from [3.13] that  $\mathbf{v}_1 = 0$ . In view of [3.14] it follows that  $\mathbf{v}_2 = 0$ . Iterating it follows that  $\mathbf{v} = 0$ .

For  $\boldsymbol{\lambda} \neq 0$ , since  $\mathbf{v}_1 = 0$  and in view of the structure of  $\mathcal{L}_r$  in [3.12], it follows that  $\mathbf{v}_n = 0$ . Substituting into [3.12] gives  $\mathbf{v}_{n-1} = 0$ . Iterating it follows  $\mathbf{v} = 0$ . We conclude that the system is controllable.

### 3.2.7. The chain topology in the general case

In this section, it will be proved that the chain topology is controllable and observable in the general case. The Laplacian matrix of the chain topology is as follows:

$$-\mathcal{L}_c = \begin{bmatrix} -1 & 1 & & & \\ 1 & -2 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & 1 & -2 & 1 \\ & & & 1 & -1 \end{bmatrix} \quad [3.15]$$

and  $\mathbf{c}^T = \mathbf{b} = [0 \ \cdots \ 1 \ \cdots \ 0]$ , where only the  $k$ th element is different from zero. For simplicity, this system will be rewritten as

$$\begin{aligned} \ddot{\mathbf{x}} &= -\mathcal{L}_c \dot{\mathbf{x}} + \mathbf{b} u_l \\ y_l &= \mathbf{c}^T \dot{\mathbf{x}} \end{aligned}$$

#### 3.2.7.1. Controllability

The system will be non-controllable if there exists a vector  $\mathbf{v}$  such that

$$\mathbf{v}^T \mathbf{b} = 0 \quad [3.16]$$

and

$$\mathbf{v}^T \mathcal{L}_c = \lambda \mathbf{v}^T \quad [3.17]$$

with  $\mathbf{v}^T = [v_1 \ v_2 \ \cdots \ v_n]$  and a real  $\lambda$ .

For  $\lambda = 0$ , since  $v_1 = 0$ , from [3.17] it follows that  $v_2 = 0$ . Iterating it follows that  $\mathbf{v} = 0$ .

For  $\lambda \neq 0$ , in view of the structure of  $\mathcal{L}_c$  in [3.15] it follows from [3.17] that  $v_2 = 0$ . Iterating it follows that  $\mathbf{v} = 0$ . We conclude that the system is controllable.

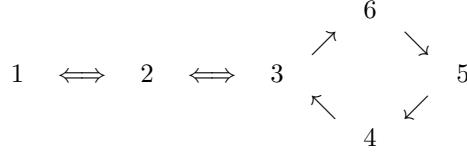
#### 3.2.7.2. Observability

Given that  $\mathcal{L}_c = \mathcal{L}_c^T$  and  $\mathbf{b} = \mathbf{c}^T$ , the system is also observable.

### 3.2.8. Combinations of chain and cyclic topologies

In this section, we show that a network of agents obtained by appropriately combining the cyclic and chain topologies is controllable and observable. Let us

consider the following topology:



Considering the input and output of agent 1, the state space representation is given by

$$\ddot{\mathbf{x}} = -\mathcal{L}_{co}\dot{\mathbf{x}} + \mathbf{b}\mathbf{u}_1$$

$$y_1 = \mathbf{c}^T \dot{\mathbf{x}}$$

where

$$-\mathcal{L}_{co} = \begin{bmatrix} -1 & 1 & 0 & 0 & 0 & 0 \\ 1 & -2 & 1 & 0 & 0 & 0 \\ 0 & 1 & -2 & 1 & 0 & 0 \\ 0 & 0 & -1 & -1 & 1 & 0 \\ 0 & 0 & 0 & -1 & -1 & 1 \\ 0 & 0 & 0 & 0 & -1 & -1 \end{bmatrix} \quad [3.18]$$

and  $\mathbf{c} = \mathbf{b}^T = [1 \ 0 \ \dots \ 0]$ .

### 3.2.8.1. Controllability

The system will be non-controllable if there exists a vector  $\mathbf{v}$  such that

$$\mathbf{v}^T \mathbf{b} = 0$$

and

$$\mathbf{v}^T \mathcal{L}_{co} = \boldsymbol{\lambda} \mathbf{v}^T \quad [3.19]$$

with  $\mathbf{v}^T = [v_1 \ v_2 \ \dots \ v_6]$  and  $\boldsymbol{\lambda}$  real.

*Case  $\boldsymbol{\lambda} = 0$ :* In this case, the only vector satisfying [3.19] is  $\mathbf{v}^T = [1 \ 1 \ \dots \ 1]$ , which implies that  $\mathbf{v}^T \mathbf{b} \neq 0$ .

*Case  $\lambda \neq 0$ :* In view of the structure of  $\mathcal{L}_{co}$  in [3.18], it follows that  $v_1 = 0 \implies v_2 = 0$ . Similarly, it follows that  $v_3 = 0, v_6 = 0, v_4 = 0$ , and  $v_5 = 0$ .

We conclude that the system is controllable.

### 3.2.8.2. Observability

The system will be non-observable if there exists a vector  $\mathbf{v}$  such that

$$\mathbf{v}^T \mathbf{b} = 0 \quad [3.20]$$

and

$$\mathcal{L}_{co} \mathbf{v} = \lambda \mathbf{v} \quad [3.21]$$

with  $\mathbf{v}^T = [v_1 \ v_2 \ \cdots \ v_6]$  and a real  $\lambda$ .

*Case  $\lambda = 0$ :* The only vector satisfying [3.21] is  $v^T = [1 \ 1 \ \cdots \ 1]$ , and thus,  $\mathbf{v}^T \mathbf{b} \neq 0$ , which is a contradiction.

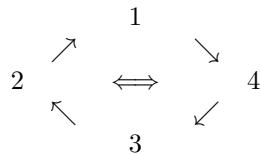
*Case  $\lambda \neq 0$ :* In view of the structure of  $\mathcal{L}_{co}$  in [3.19] and since  $v_1 = 0$ , it follows that  $v_2 = 0$ . Iterating it follows that  $v = 0$ . We conclude that the system is observable.

REMARK.– In view of the structure of  $\mathcal{L}_{co}$  in [3.19] and the arguments described above, it follows that the controllability and the observability properties are preserved if the chain and the cyclic topologies in [3.18] have arbitrary length.

### 3.2.9. Simple configurations that are either non-controllable or non-observable

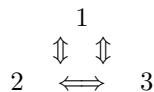
In order to help characterizing the configurations that are both controllable and observable, we present, in this section, a series of illustrative simple configurations that are not in the form of [3.18] and that fail to be controllable and observable. We will denote by  $\mathcal{L}$ ,  $\mathfrak{C}_i$ , and  $\mathcal{O}_i$  the Laplacian, the controllability, and the observability matrices, respectively, when using the input and output of the  $i$ th agent.

#### 3.2.9.1. Example 1



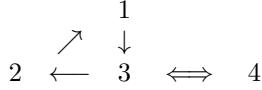
Note that  $\det \mathfrak{C}_1 = \det \mathfrak{C}_2 = 0$ . We have that  $\det \mathcal{O}_1 = \det \mathcal{O}_2 = 0$ .

#### 3.2.9.2. Example 2



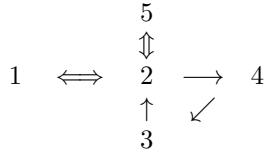
Note that  $\det \mathfrak{C}_1 = \det \mathcal{O}_1 = 0$ .

### 3.2.9.3. Example 3



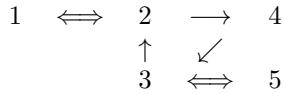
$\det \mathbf{C}_1 = 0$ ,  $\det \mathcal{O}_1 = 1$ ,  $\det \mathbf{C}_2 = -1$ ,  $\det \mathcal{O}_2 = 0$ ,  $\det \mathbf{C}_3 = 0$ ,  $\det \mathcal{O}_3 = 0$ ,  $\det \mathbf{C}_4 = 1$ , and  $\det \mathcal{O}_4 = -1$ . Controllability and observability from agent 4 can also be proved as it was done for [3.18].

### 3.2.9.4. Example 4



$\det \mathbf{C}_1 = 0$ ,  $\det \mathcal{O}_1 = 0$ .

### 3.2.9.5. Example 5



$\det \mathbf{C}_1 = -1$ ,  $\det \mathcal{O}_1 = 0$ .

## 3.3. Formation leader tracking

In this section, we introduce a coordination control for flock tracking. We first study the case of three agents, and then we extend the method to the general case.

Consider the case of three agents in chain topology described in [3.3]. Premultiplying [3.3] by  $\mathbf{w}_1^T$ ,  $\mathbf{w}_2^T$ , and  $\mathbf{w}_3^T$  above, we obtain

$$\ddot{x}_1 + \ddot{x}_2 + \ddot{x}_3 = u_1 \quad [3.22]$$

$$\ddot{x}_1 - \ddot{x}_3 = -(\dot{x}_1 - \dot{x}_3) + u_1 \quad [3.23]$$

$$\ddot{x}_1 - 2\ddot{x}_2 + \ddot{x}_3 = -3(\dot{x}_1 - 2\dot{x}_2 + \dot{x}_3) + u_1 \quad [3.24]$$

Let us define  $\dot{x}_{CM} \triangleq 1/3 \sum_{i=1}^3 \dot{x}_i$  as the center of mass of system [3.22] and let  $\dot{x}_{CM}^d$  be the desired velocity for the center of mass. Let us consider the tracking control law

$$u_1 \triangleq 3k_{CM}\sigma(\dot{x}_{CM}^d - \dot{x}_{CM}) \quad [3.25]$$

where  $\sigma(\cdot)$  represents the saturation function and  $k_{CM}$  is a constant positive gain. Note that  $\dot{x}_{CM}$  is not directly measurable by the leader (agent 1), but the state can be observed from the input and output of agent 1. Defining  $e_{CM} \triangleq \dot{x}_{CM}^d - \dot{x}_{CM}$  and introducing [3.25] into [3.22], we get

$$\begin{aligned} \ddot{x}_{CM} &= k_{CM}\sigma(e_{CM}) \\ \ddot{x}_1 - \ddot{x}_3 &= -(\dot{x}_1 - \dot{x}_3) + 3k_{CM}\sigma(e_{CM}) \\ \ddot{x}_1 - 2\ddot{x}_2 + \ddot{x}_3 &= -3(\dot{x}_1 - 2\dot{x}_2 + \dot{x}_3) + 3k_{CM}\sigma(e_{CM}) \end{aligned}$$

If the desired velocity value  $\dot{x}_{CM}^d$  is constant, then  $\dot{x}_{CM} \rightarrow \dot{x}_{CM}^d$  as  $t \rightarrow \infty$ , which implies that  $u_1 \rightarrow 0$  and  $(\dot{x}_1 - \dot{x}_3), (\dot{x}_1 - \dot{x}_2) \rightarrow 0$ . Note that for small values of  $k_{CM}$ , the convergence speed of  $\dot{x}_{CM}$  is slow, but the transient in the errors  $(\dot{x}_1 - \dot{x}_3), (\dot{x}_1 - \dot{x}_2)$  will be smaller.

From the previous analysis, we can state the following lemma.

**LEMMA 3.2.-** Consider a multiagent system of the form [3.8] with coordinating control law [3.25]. If  $\dot{x}_{CM}^d$  is constant, then  $\dot{x}_{CM} \rightarrow \dot{x}_{CM}^d$  as  $t \rightarrow \infty$ , which implies that  $u_1 \rightarrow 0$  and  $(\dot{x}_i - \dot{x}_j) \rightarrow 0$ .

### 3.3.1. Formation leader tracking in the general case

Define  $\dot{x}_{CM} \triangleq 1/N \sum_{i=1}^N \dot{x}_i$ , where  $N$  is the number of agents in the formation. Let  $\dot{x}_{CM}^d$  be the desired value for  $\dot{x}_{CM}$ . Assume, for simplicity, that agent 1 is the leader, i.e.  $\mathbf{c}^T = \mathbf{b}^T = [1 \ 0 \ \dots \ 0]$  and that the control law is

$$\mathbf{u}_1 \triangleq \begin{bmatrix} Nk_{CM}\sigma(\dot{x}_{CM}^d - \dot{x}_{CM}) \\ 0 \\ 0 \end{bmatrix} \quad [3.26]$$

Note that  $\dot{x}_{CM}$  may not be directly measurable for the leader (agent 1). We assume the system is observable from the input and output of the leader. The state can, therefore, be observed from the input and output of agent 1. Substituting [3.25] into [3.9], we get

$$\begin{aligned} \ddot{x}_{CM} &= k_{CM}\sigma(\dot{x}_{CM}^d - \dot{x}_{CM}) \\ \frac{d}{dt}(\mathbf{v}_i^T \dot{\mathbf{x}}) &= -\boldsymbol{\lambda}_i(\mathbf{v}_i^T \dot{\mathbf{x}}) + \mathbf{v}_i^T \mathbf{b} \mathbf{u}_1, \quad i = 2, \dots, N \end{aligned} \quad [3.27]$$

The modes in the last equation above are all stable. When  $\mathbf{u}_1 = \mathbf{0}$ , these modes converge to zero, which means that  $(\dot{x}_i - \dot{x}_j) \rightarrow 0$ , for  $i \neq j$ . This property is obtained by using the coordinating control algorithm that leads to system [3.8]. These modes are uncontrollable when  $\mathbf{v}_i^T \mathbf{b} = 0$ . There is a tradeoff in the choice of gain  $k_{CM}$  in [3.26]. For smaller values of  $k_{CM}$ , the speed of convergence of  $\dot{x}_{CM}$  is slower, but the transient in the errors  $(\dot{x}_i - \dot{x}_j)$ , for  $i \neq j$ , will be smaller.

### 3.3.2. Observer design

Due to the nature of information flow between agents, full state is, in general, not available. Thus, we have developed a coordination control based on the center of mass of the multiagent system. Full state is needed in order to compute the tracking control [3.26]. In order to obtain the full state, we propose a Luenberger observer of the form:

$$\begin{aligned}\dot{\hat{x}} &= \mathbf{Ax} - \mathbf{Bu}(\hat{x}) \\ \dot{\hat{x}} &= \bar{\mathbf{L}}\mathbf{Cx} + (\mathbf{A} - \bar{\mathbf{L}}\mathbf{C})\hat{x} - \mathbf{u}(\hat{x}) \\ y &= \mathbf{Cx}\end{aligned}$$

where  $x$  is the state vector,  $\hat{x}$  is the observed state vector, and  $\bar{\mathbf{L}}$  is the Luenberger gain vector.

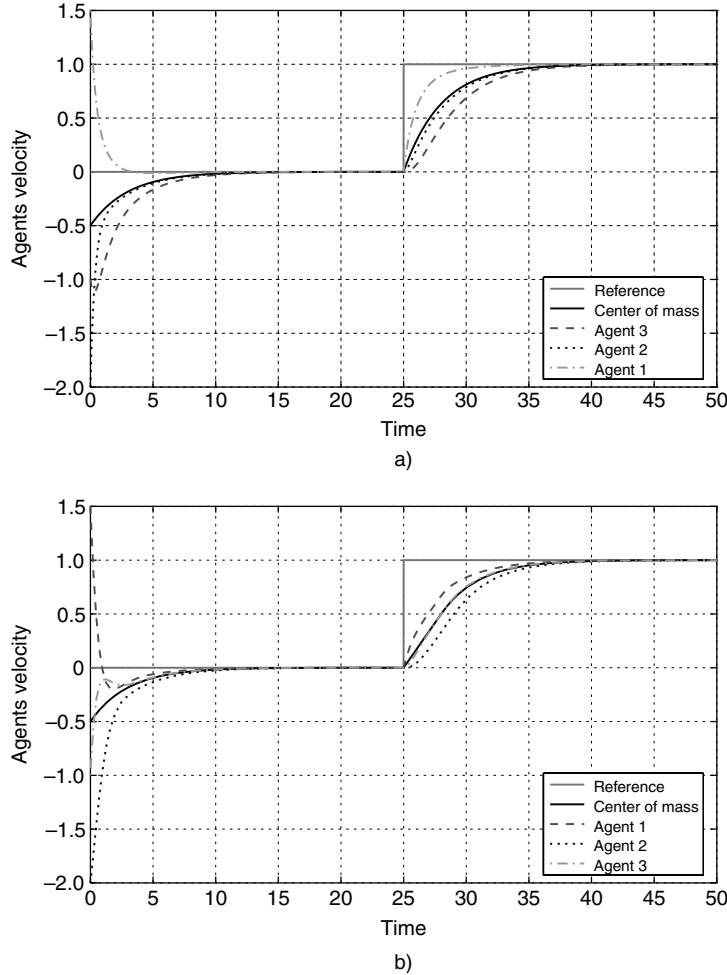
### 3.3.3. Simulations

To validate the previous results, simulations were run considering both the case when partial and full state of the multiagent system is available from measurement. When the state is not available from measurement, a Luenberger observer is used to estimate the state.

In Figures 3.2(a) and (b), it is possible to see that multiagent systems with chain and cyclic topology of information exchange synchronize in velocity to the desired velocity of the center of mass using the combination of coordinating and tracking control.

When Luenberger observer is used in the multiagent system, simulation results (Figures 3.3(a) and (b)) show the observer state convergence to the actual state which implies a convergence of the center of mass to the input reference. If the reference is constant, agents' state also converges to the center-of-mass reference.

Although synchronization of the center of mass is achieved with respect to a constant reference, when a continuous time-varying reference is given, there is a small bias in the center of mass with respect to the input reference. Agents' synchronization is not completely achieved either with respect to each other or with respect to the center of mass (see Figure 3.4).



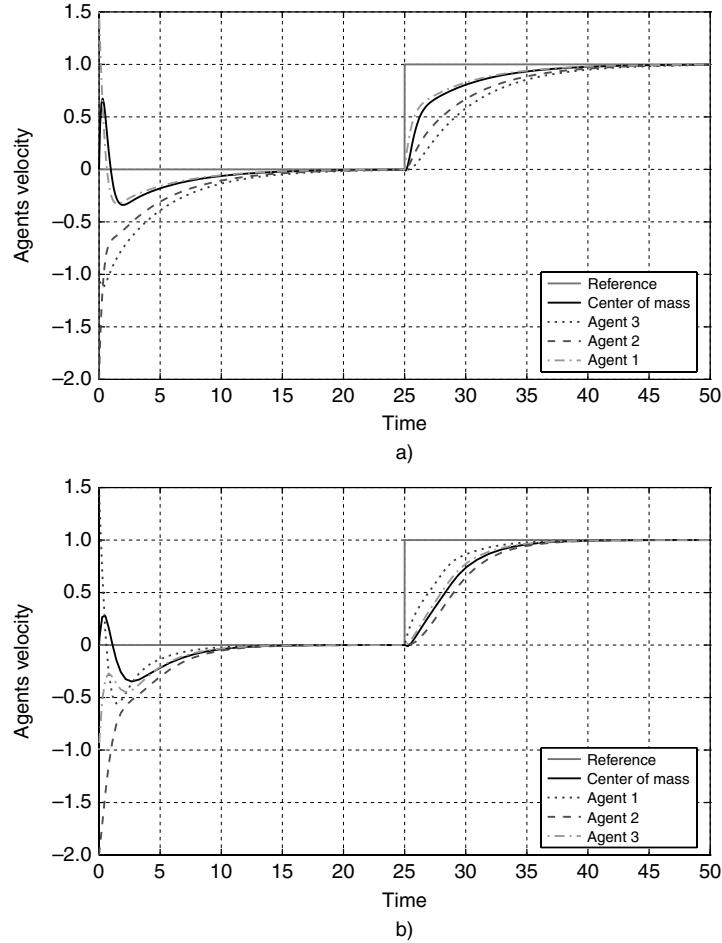
**Figure 3.2.** Velocity consensus and tracking considering full state available for: (a) a three-agent chain configuration and (b) a three-agent cyclic configuration

### 3.4. Time-varying trajectory tracking

It has been shown in previous sections that control law [3.26] ensures the convergence of the center of mass of a multiagent reference to a constant reference given to the leader. However, when the reference is varying in time, there is a small bias in agents coordination. In this section, we consider the case of multiagent trajectory tracking of a time-varying reference. We will prove that a double-integrator multiagent system converges to the position time-varying reference given only to the

leader. Again, we are interested in the chain and cyclic topologies of information exchange. Let us consider the double integrator multiagent system of the form

$$\ddot{x}_i = u_i \quad [3.28]$$



**Figure 3.3.** Velocity consensus and tracking using Luenberger observer for:  
(a) a three-agent chain configuration and (b) a three-agent cyclic configuration

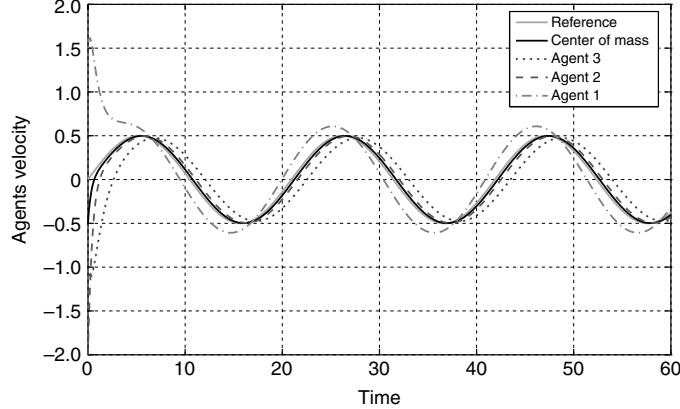
We define a change of variable

$$\dot{\xi}_i \triangleq \dot{x}_i + \kappa x_i \quad [3.29]$$

$$\dot{\xi}_{CM}^d \triangleq \dot{x}_{CM}^d + \kappa x_{CM}^d \quad [3.30]$$

where  $\kappa$  is a positive constant. Control input  $u_i$  is defined as

$$u_i \triangleq \bar{u}_i - \kappa \dot{x}_i \quad [3.31]$$



**Figure 3.4.** Velocity consensus to a time-varying reference for a three-agent chain configuration

Differentiating [3.29] and using control [3.31], we obtain

$$\ddot{\xi}_i = \bar{u}_i \quad [3.32]$$

We define the coordinating control  $\bar{u}_i$  as

$$\bar{u}_i \triangleq - \sum_{j \in \mathcal{N}_i} (\dot{\xi}_i - \dot{\xi}_j) + b \tilde{u} \quad [3.33]$$

Thus, we obtain the following multiagent system:

$$\ddot{\xi} = -\mathcal{L}\dot{\xi} + b\tilde{u} \quad [3.34]$$

where  $\mathcal{L}$  is the Laplacian matrix and the control law

$$\tilde{u}_l \triangleq \tilde{u}_{CM} \quad [3.35]$$

$$\tilde{u}_i \triangleq \dot{\xi}_j \text{ for some } j \in \mathcal{N}_i \quad [3.36]$$

where  $\tilde{u}_{CM} = k_{CM} \text{ sat}\{\dot{\xi}_{CM} - \dot{\xi}_{CM}^d\}$ .

Thus, the case of three agents with chain topology of information exchange with agent 1 acting as the leader of the group is represented as

$$\dot{\xi}_1 = (\xi_2 - \xi_1) + \tilde{u}_{CM} \quad [3.37]$$

$$\dot{\xi}_2 = (\xi_1 - \xi_2) + (\xi_3 - \xi_2) + \dot{\xi}_1 \quad [3.38]$$

$$\dot{\xi}_3 = (\xi_2 - \xi_3) + \dot{\xi}_2 \quad [3.39]$$

which can also be represented as

$$\begin{bmatrix} \dot{\xi}_1 \\ \dot{\xi}_2 \\ \dot{\xi}_3 \end{bmatrix} = \begin{bmatrix} -1 & 1 & 0 \\ 1 & -2 & 1 \\ 0 & 1 & -1 \end{bmatrix} \begin{bmatrix} \xi_1 \\ \xi_2 \\ \xi_3 \end{bmatrix} + \begin{bmatrix} \tilde{u}_{CM} \\ \dot{\xi}_1 \\ \dot{\xi}_2 \end{bmatrix} \quad [3.40]$$

From [3.39], we get

$$(\dot{\xi}_3 - \dot{\xi}_2) = -(\xi_3 - \xi_2) \quad [3.41]$$

then, from [3.41] we see that  $(\xi_3 - \xi_2) \rightarrow 0$ .

From [3.38], we get

$$(\dot{\xi}_2 - \dot{\xi}_1) = (\xi_1 - \xi_2) + (\xi_3 - \xi_2) \quad [3.42]$$

since  $(\xi_3 - \xi_2) \rightarrow 0$ , then [3.42] is reduced to

$$(\dot{\xi}_2 - \dot{\xi}_1) = -(\xi_2 - \xi_1) \quad [3.43]$$

which implies that  $(\xi_2 - \xi_1) \rightarrow 0$ .

Premultiplying [3.40] by 1 eigenvector, we get

$$\dot{\xi}_1 + \dot{\xi}_2 + \dot{\xi}_3 = \tilde{u}_{CM} + \dot{\xi}_1 + \dot{\xi}_2 \quad [3.44]$$

$$\dot{\xi}_3 = \tilde{u}_{CM} \quad [3.45]$$

Now, we define

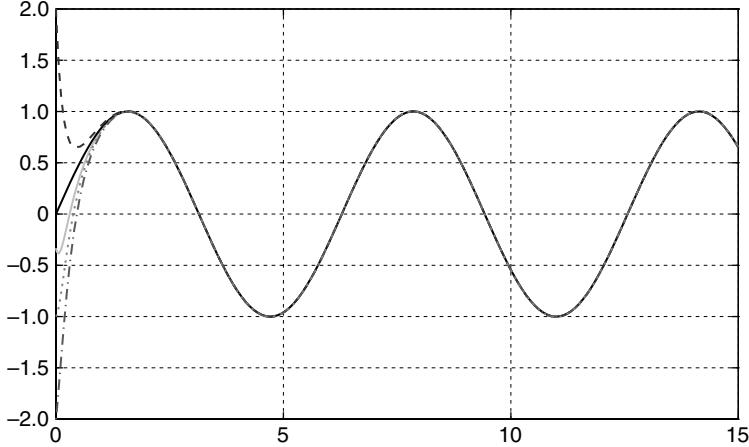
$$\tilde{u}_{CM} \triangleq -(\xi_{CM} - \xi_{CM}^d) + \dot{\xi}_{CM}^d \quad [3.46]$$

and then introducing [3.46] into [3.45], we obtain

$$(\dot{\xi}_3 - \dot{\xi}_{CM}^d) = -(\xi_{CM} - \xi_{CM}^d) \quad [3.47]$$

which implies that  $(\xi_3 - \xi_{CM}^d) \rightarrow 0$ . Assume that  $\xi_{CM}^d = x_{CM}^d$  and  $\dot{x}_{CM}^d = 0$ , then  $(\xi_3 - x_{CM}^d) \rightarrow 0$  implies that  $(x_3 - x_{CM}^d) \rightarrow 0$ . Since  $(x_i - x_j) \rightarrow 0$  and  $(x_3 - x_{CM}^d) \rightarrow 0$  then  $(x_i - x_{CM}^d) \rightarrow 0$ .

**LEMMA 3.3.-** Consider a multiagent system of the form [3.34] with coordinating control laws [3.35] and [3.36]. Then,  $\dot{x}_{CM} \rightarrow \dot{x}_{CM}^d$  as  $t \rightarrow \infty$ ; moreover,  $(\dot{x}_i - \dot{x}_{CM}^d) \rightarrow 0$ .



**Figure 3.5.** Position consensus to a time-varying reference for a three-agent chain configuration

### 3.5. Linear high-order multiagent consensus

As it has been proven, multiagent consensus can be applied not only to a single-integrator multiagent system. Now we want to show that by following the same recursive algorithm presented in the previous section it is possible to get a consensus for  $n$ -order integrator multiagent system. Then, let us consider a linear high-order multiagent system to show consensus and forced consensus convergence. To illustrate the algorithm, we will use a fourth-order integrator multiagent system. Consider the following multiagent system

$$x_i^{(iv)} = u_i \quad [3.48]$$

for all  $i = 1, \dots, 4$ .

A first change of variable is proposed

$$\xi_i \triangleq \dot{x}_i + \kappa x_i \quad [3.49]$$

where  $\kappa$  is a positive constant. The third derivative of the above equation is

$$\xi_i^{(iii)} = x_i^{(iv)} + \kappa x_i^{(iii)} \quad [3.50]$$

$$\xi_i^{(iii)} = u_i + \kappa x_i^{(iii)} \quad [3.51]$$

The control  $u_i$  is defined as

$$u_i \triangleq u_i' - \kappa x_i^{(iii)} \quad [3.52]$$

Equation [3.51] can be rewritten as

$$\xi_i^{(iii)} = u'_i$$

By following the iterative algorithm, we define a new variable

$$\bar{\xi}_i \triangleq \dot{\xi}_i + \kappa \xi_i \quad [3.53]$$

the second derivative w.r.t. is given by

$$\ddot{\xi}_i = \xi_i^{(iii)} + \kappa \ddot{\xi}_i \quad [3.54]$$

$$\ddot{\xi}_i = u'_i + \kappa \ddot{\xi}_i \quad [3.55]$$

The control  $u'_i$  is defined as

$$u'_i \triangleq \tilde{u}_i - \kappa \ddot{\xi}_i \quad [3.56]$$

Equation [3.55] can be rewritten as

$$\ddot{\xi}_i = \tilde{u}_i$$

By following the iterative algorithm, we define the last change of variable

$$\tilde{\xi}_i \triangleq \dot{\xi}_i + \kappa \bar{\xi}_i \quad [3.57]$$

the first derivative w.r.t. is given by

$$\dot{\tilde{\xi}}_i = \ddot{\xi}_i + \kappa \dot{\bar{\xi}}_i \quad [3.58]$$

$$\dot{\tilde{\xi}}_i = \tilde{u}_i + \kappa \dot{\bar{\xi}}_i \quad [3.59]$$

The control  $\tilde{u}_i$  is defined as

$$\tilde{u}_i \triangleq \bar{u}_i - \kappa \dot{\bar{\xi}}_i \quad [3.60]$$

Equation [3.59] can be rewritten as

$$\dot{\tilde{\xi}}_i = \bar{u}_i \quad [3.61]$$

We define the controller  $\bar{u}_i$  as

$$\bar{u}_i \triangleq - \sum_{j \in \mathcal{N}_i} (\tilde{\xi}_i - \tilde{\xi}_j) \quad [3.62]$$

Introducing [3.62] into [3.60], we have

$$\tilde{u}_i = - \sum_{j \in \mathcal{N}_i} (\tilde{\xi}_i - \tilde{\xi}_j) - \kappa \dot{\bar{\xi}}_i \quad [3.63]$$

Introducing [3.63] into [3.56], we have

$$u_i' = - \sum_{j \in \mathcal{N}_i} (\tilde{\xi}_i - \tilde{\xi}_j) - \kappa \dot{\tilde{\xi}}_i - \kappa \ddot{\xi}_i \quad [3.64]$$

Introducing [3.64] into [3.52], we have

$$u_i = - \sum_{j \in \mathcal{N}_i} (\tilde{\xi}_i - \tilde{\xi}_j) - \kappa \dot{\tilde{\xi}}_i - \kappa \ddot{\xi}_i - \kappa x_i^{(iii)} \quad [3.65]$$

Introducing [3.65] into [3.48], we have

$$x_i^{(iv)} = - \sum_{j \in \mathcal{N}_i} (\tilde{\xi}_i - \tilde{\xi}_j) - \kappa \dot{\tilde{\xi}}_i - \kappa \ddot{\xi}_i - \kappa x_i^{(iii)} \quad [3.66]$$

where

$$\ddot{\xi}_i = x_i^{(iii)} + \kappa \ddot{x}_i \quad [3.67]$$

$$\dot{\tilde{\xi}}_i = x_i^{(iii)} + 2\kappa \ddot{x}_i + \kappa^2 \dot{x}_i \quad [3.68]$$

Then, [3.66] can be rewritten as

$$x_i^{(iv)} = - \sum_{j \in \mathcal{N}_i} (\tilde{\xi}_i - \tilde{\xi}_j) - 3\kappa x_i^{(iii)} - 3\kappa^2 \ddot{x}_i - \kappa^3 \dot{x}_i \quad [3.69]$$

A positive-definite Lyapunov function is proposed as given in [CHO 06]

$$V = 2(V_1 + V_2) \quad [3.70]$$

where  $V_i$  is the storage function for each dynamical subsystem, i.e. for all  $i = 1, 2$ . Then, using Lemma 2.1,  $\dot{V}$  is given by

$$\dot{V} = -S_i(\tilde{\xi}_i) - K \sum_{i=1}^N \sum_{j \in \mathcal{N}_i} (\tilde{\xi}_i - \tilde{\xi}_j)^T (\tilde{\xi}_i - \tilde{\xi}_j) \quad [3.71]$$

where  $V_i$  is the storage function of the system. Noting that  $S_i(\tilde{\xi}_i) = 0 \forall i$  and integrating the above equation we can see that  $(\tilde{\xi}_i - \tilde{\xi}_j) \in \mathcal{L}_2$ . Using [3.57], we get

$$(\tilde{\xi}_i - \tilde{\xi}_j) = (\dot{\tilde{\xi}}_i - \dot{\tilde{\xi}}_j) + \kappa(\bar{\xi}_i - \bar{\xi}_j) \quad [3.72]$$

Defining  $e_{ij} = \bar{\xi}_i - \bar{\xi}_j$  and differentiating with respect to time, we get  $\dot{e}_{ij} = \dot{\tilde{\xi}}_i - \dot{\tilde{\xi}}_j$ , then [3.72] is rewritten as

$$(\tilde{\xi}_i - \tilde{\xi}_j) = \dot{e}_{ij} + \kappa e_{ij} \quad [3.73]$$

Then,  $\dot{e}_{ij} + \kappa e_{ij} \in \mathcal{L}_2$ , which guarantees the exponential convergence of  $e_{ij}$  to the origin. Assuming that the information exchange graph is strongly connected, then

$$\lim_{t \rightarrow \infty} |\bar{\xi}_i - \bar{\xi}_j| = 0$$

Since  $(\bar{\xi}_i - \bar{\xi}_j) \in \mathcal{L}_2$ , using [3.53], we have

$$(\bar{\xi}_i - \bar{\xi}_j) = (\dot{\xi}_i - \dot{\xi}_j) + \kappa(\xi_i - \xi_j) \quad [3.74]$$

Defining  $e'_{ij} = \xi_i - \xi_j$  and differentiating with respect to time, we get  $\dot{e}'_{ij} = \dot{\xi}_i - \dot{\xi}_j$ , then [3.75] is rewritten as

$$(\bar{\xi}_i - \bar{\xi}_j) = \dot{e}'_{ij} + \kappa e'_{ij} \quad [3.75]$$

Then,  $\dot{e}'_{ij} + \kappa e'_{ij} \in \mathcal{L}_2$ , which guarantees the exponential convergence of  $e'_{ij}$  to the origin. Assuming that the information exchange graph is strongly connected, then

$$\lim_{t \rightarrow \infty} |\xi_i - \xi_j| = 0$$

We can see that  $(\xi_i - \xi_j) \in \mathcal{L}_2$ . Using [3.49], we have

$$(\xi_i - \xi_j) = (\dot{x}_i - \dot{x}_j) + \kappa(x_i - x_j) \quad [3.76]$$

Defining  $\tilde{e}_{ij} = x_i - x_j$  and differentiating with respect to time, we get  $\dot{\tilde{e}}_{ij} = \dot{x}_i - \dot{x}_j$ , then [3.76] is rewritten as

$$(\xi_i - \xi_j) = \dot{\tilde{e}}_{ij} + \kappa \tilde{e}_{ij} \quad [3.77]$$

Then,  $\dot{\tilde{e}}_{ij} + \kappa \tilde{e}_{ij} \in \mathcal{L}_2$ , which guarantees the exponential convergence of  $\tilde{e}_{ij}$  to the origin. Assuming that the information exchange graph is strongly connected then

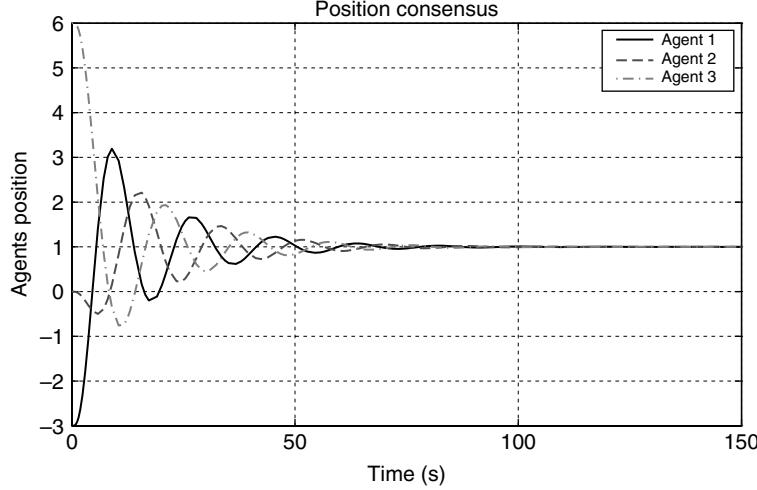
$$\lim_{t \rightarrow \infty} |x_i - x_j| = 0$$

After a time  $T > 0$ ,  $\sum_{j \in N_i} (\tilde{\xi}_i - \tilde{\xi}_j) \rightarrow 0$ ; therefore, from [3.69] it can be seen that  $x_i^{(iii)}, \ddot{x}_i, \dot{x}_i \rightarrow 0$ .

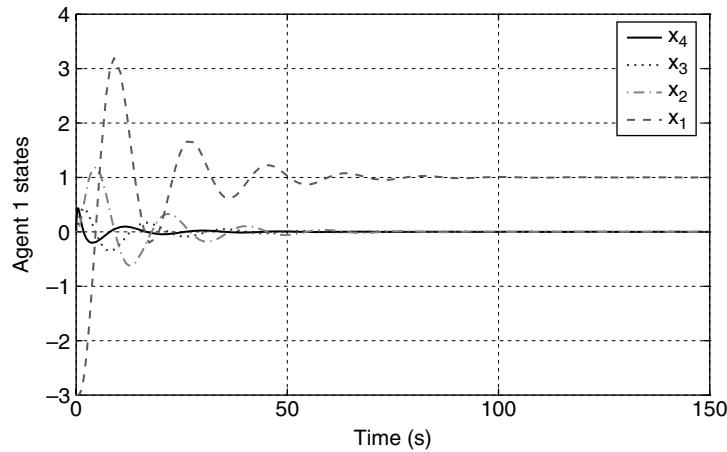
The control law [3.69] guarantees the coordination of lateral and longitudinal subsystems. It should be noticed that the lateral and longitudinal systems will reach consensus in the sense that

$$\lim_{t \rightarrow \infty} |\tilde{\xi}_i - \tilde{\xi}_j| = 0 \quad \forall i, j. \quad [3.78]$$

In Figure 3.6, the position consensus of a three-agent system with cyclic topology of information exchange is shown.



**Figure 3.6.** Position consensus for a three-agent cyclic configuration



**Figure 3.7.** Leader agent states for a three-agent cyclic configuration

### 3.5.1. Trajectory-tracking control

Now, we will consider the problem of the trajectory tracking. It is assumed that each agent has an access to the desired trajectory. Then,  $\bar{u}_i$  is rewritten as

$$\bar{u}_i = - \sum_{j \in \mathcal{N}_i} (\tilde{\xi}_i - \tilde{\xi}_j) + u_{CM} \quad [3.79]$$

where  $u_{CM}$  is the reference input. Define  $\tilde{\xi}_{CM} = 1/N \sum_{i=1}^N \tilde{\xi}_i$ , where  $N$  is the number of agents in the formation. Let  $\tilde{\xi}_{CM}^d$  be the desired value for  $\tilde{\xi}_{CM}$ . To control the center of mass, it is proposed that

$$u_{CM}(\tilde{\xi}) \triangleq Nk_{CM}\sigma\left(\tilde{\xi}_{CM}^d - \tilde{\xi}_{CM}\right) \quad [3.80]$$

Trajectory-tracking control law is such that  $\tilde{\xi}_{CM} \rightarrow \tilde{\xi}_{CM}^d$  as  $t \rightarrow \infty$ , which implies that  $z \rightarrow \tilde{\xi}_{CM}^d/\kappa$ , which, in turn, implies that  $y \rightarrow \tilde{\xi}_{CM}^d/\kappa^2$  and  $x \rightarrow \tilde{\xi}_{CM}^d/\kappa^3$ . When  $\dot{\tilde{\xi}}_{CM}^d = \kappa^3 x_{CM}^{(d)}$  implies that  $|x_i - x_i^d| \rightarrow x_{CM}^d$ .

### 3.6. Conclusion

In this chapter, we have proposed a control method for synchronizing and tracking multiagent systems. We have used the coordinating control in [CHO 06] for which the input of every agent depends only on its neighbors information.

Tracking of the center of mass of the agents' formation has been achieved by using state feedback control applied to the leader. Since the leader is not assumed to have direct information from all the agents, the state is observed from the input and output of the leader. This approach requires observability and controllability (or at least detectability and stabilizability) of the agents' network from the leader input and output. The observability/controllability properties for chain and cyclic topologies has been studied in this chapter.

It has been shown that the interconnection of agents using the coordinating control strategy and leading to systems as [3.8] are stabilizable and detectable from the input and output of any agent.

It has been proved that, after applying a coordinating control, the cyclic topology is observable and controllable from any agent and the chain topology is observable and controllable from the first or the last agent of the chain. A cyclic topology and a chain topology can be interconnected in cascade to obtain a new configuration that is controllable and observable provided that the input and output are taken from the first agent of the chain topology as in [3.18]. Several simple counter examples show that adding any extra interconnection to the cyclic topology, the chain topology, or the combination of both, as in [3.18], leads to either uncontrollable or unobservable modes. This means that the most general combination of chain and cyclic topologies that is both controllable and observable from one agent (the first agent of the chain) is the cascade interconnection of a chain topology and a cyclic topology as in [3.18].

It has been shown that a tracking of time-varying reference is achieved using control laws [3.35] and [3.36]. Also it has been proved that linear high-order multiagent consensus is possible using a recursive change of variable.

### 3.7. Bibliography

- [CHO 06] CHOPRA N., Output synchronization of mechanical systems, PhD Thesis, University of Illinois at Urbana-Champaign, 2006.
- [JI 06] JI M., MUHAMMAD A., EGERSTEDT M., “Leader-based multiagent coordination: controllability and optimal control”, *IEEE American Control Conference*, pp. 1358–1363, 2006.
- [RAH 06] RAHMANI A., MESBAHI M., “On the controlled agreement problem”, *IEEE American Control Conference*, pp. 1376–1381, 2006.
- [RAH 07] RAHMANI A., MESBAHI M., “Pulling the strings on agreement: anchoring, controllability and graph automorphisms”, *IEEE American Control Conference*, pp. 2738–2743, 2007.
- [REN 07] REN W., BEARD R., ATKINS E., “Information consensus in multivehicle cooperative control”, *IEEE Control Systems Magazine*, vol. 27, no. 2, pp. 71–82, 2007.

## Chapter 4

# Robust Control Design of Multiagent Systems with Parametric Uncertainty

This chapter introduces a robust control design technique for linear systems with application to multiple agent (multiagent) systems. Chain, cyclic, and irregular topologies of information exchange are analyzed. The interactions of a multiple robot system are modeled as a linear system, and the forced consensus is possible by using a state feedback control. Our approach is based on a leader/follower structure of the multiple robot systems. A Lyapunov approach to robust control design based on optimal control design techniques is developed. The value set characterization and the robust strict positive realness (SPR) property are used as tools for analysis to verify the stability of the dynamical system. The novelty of this approach is to combine optimal control and the value set characterization to robust control design for multiagent systems with time delay.

The remainder of this chapter is organized as follows. In section 4.1, a general overview of the robust control design by means of optimal control is introduced. Section 4.2 presents an approach to robust optimal control. Here, the goal is to design a state feedback to stabilize the forced consensus of a multiagent system considering uncertainty in the system. A multiagent system can be considered as a linear system and thereby linear optimal control is used to design a controller that minimizes the trajectory tracking error as well as the energy. Section 4.3 introduces concepts and results to analyze the robust stability of multiagent systems. We propose to analyze two cases: the robust absolute stability property of Lur'e systems and the robust stability property in presence of time delays in the input. In the first case, we

---

Chapter written by J.A. GUERRERO and G. ROMERO.

propose to transform the multiagent system into a Lur'e system to analyze the robust absolute stability property. In the second case, using the value set, a graphical tool, we show that the multiagent system is robustly stable in presence of time delays. In section 4.5, application of robust control to multiagent systems with chain topology, cyclic topology, and irregular topology is shown.

#### 4.1. Introduction

When stabilizing a dynamical system, it is usual to design a controller using an inaccurate mathematical model. Thus, the following fundamental question arises: Will the controller developed for an inaccurate mathematical model drive, stabilize the physical system? It is important to develop robust controllers due to the fact that modern dynamical systems become more complex everyday. The importance of developing robust controllers is reflected in the literature. Robust and optimal control are branches of modern control that provides tools for analysis and synthesis of control systems. We identify several approaches to robust and optimal control:  $H^\infty$  approach [HEL 98], [ZHO 96], parametric approach [BAR 94], [KHA 79], [FU 89a], [FU 89b], and Lyapunov approach [KRA 88], [LIN 07], among others.

As discussed in Chapter 3, multiagent systems have important applications in engineering. Then, we focus our attention on the synthesis of controllers for multiagent systems with parametric uncertainty and the stability analysis for such systems. An extension for time-delay multiagent systems is also considered. Cyclic topology and chain topology of information exchange are considered in both these cases.

It is clear that one of the first challenges in control is to stabilize a dynamical system using either a linear controller or a nonlinear controller. After we stabilize a system, for instance a multiagent system, the following challenge is to optimize the system's performance. The optimal control considers the problem of minimizing a cost or maximizing a utility (or minimizing a negative cost) for general nonlinear time-invariant systems which are described as:

$$\dot{\mathbf{x}} = f(\mathbf{x}, \mathbf{u}), \quad [4.1]$$

where  $\mathbf{x} \in \mathbb{R}^n$  and  $\mathbf{u} \in \mathbb{R}^m$  are the state variables and control inputs, respectively, and  $f(\cdot, \cdot)$  satisfies the conditions for the existence of the solution. The cost function, also called performance index, may comprise a wide variety of costs including the time needed for a control to perform a desired objective, the energy needed to perform the desired task, combination of cost functions, etc. As described in [LIN 07], the goal of the optimal control is to minimize a cost:

$$J(\mathbf{x}, t) = \int_t^{t_f} D(\mathbf{x}, \mathbf{u}) d\tau, \quad [4.2]$$

where  $D(\mathbf{x}, \mathbf{u})$  is the cost objective to minimize. As mentioned before, the cost objective may take different functions or objectives in control, such as minimal tracking error, minimal energy or combined minimal tracking error, and minimal energy functional cost.

*Minimal tracking error:* In this case, the goal for the system is to converge to a desired value  $\mathbf{x}_d$ . Then, it is usual to take the cost function as  $D(\mathbf{x}, \mathbf{u}) = \|\mathbf{x} - \mathbf{x}_d\|$  or  $D(\mathbf{x}, \mathbf{u}) = (\mathbf{x} - \mathbf{x}_d)^T \mathbf{H}(\mathbf{x} - \mathbf{x}_d)$ , where  $\mathbf{H} = \mathbf{H}^T \geq 0$  is a symmetric and positive-definite matrix. An example of an application is the case of a group of coordinated agents displacing at cruise speed where the goal is to maintain the speed as constant as possible for every agent.

*Minimal energy:* In this case, the goal is to use the minimal energy to control the system. Then, the cost function takes the form  $D(\mathbf{x}, \mathbf{u}) = \|\mathbf{u}\|$  or  $D(\mathbf{x}, \mathbf{u}) = \mathbf{u}^T \mathbf{R} \mathbf{u}$ , where  $\mathbf{R} = \mathbf{R}^T \geq 0$  is a symmetric and positive-definite matrix. The cost for this case takes an input quadratic form because usually the input is related to the energy consumed by the system.

*Minimal tracking error and minimal energy:* Here, the goal for the system is to converge to a desired value  $\mathbf{x}_d$  using minimal energy. Then, the cost function is defined as  $D(\mathbf{x}, \mathbf{u}) = (\mathbf{x} - \mathbf{x}_d)^T \mathbf{H}(\mathbf{x} - \mathbf{x}_d) + \mathbf{u}^T \mathbf{R} \mathbf{u}$ , where  $\mathbf{H}$  and  $\mathbf{R}$  are the weights on the state and control, respectively.

It is known that solving the Hamilton–Jacobi–Bellman equation is not easy, in particular, for nonlinear systems, but if the functional cost is quadratic, e.g. minimal tracking error and minimal energy and the dynamic system is linear, then the Hamilton–Jacobi–Bellman is reduced to the following algebraic Riccati equation:

$$\mathbf{S}\mathbf{A} + \mathbf{A}^T\mathbf{S} + \mathbf{H} - \mathbf{S}\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\mathbf{S} = 0. \quad [4.3]$$

The solution for  $\mathbf{S}$  represents the solution of the optimal control problem, which is also called linear quadratic regulator problem:

$$\mathbf{u}^* = -\mathbf{R}^{-1}\mathbf{B}^T\mathbf{S}\mathbf{x}. \quad [4.4]$$

We consider the problem of robust control for linear systems of the form:

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{A}(\mathbf{q})\mathbf{x} + \mathbf{B}(\mathbf{q})\mathbf{u}, \\ \mathbf{y} &= \mathbf{C}(\mathbf{q})\mathbf{x}, \end{aligned} \quad [4.5]$$

where  $\mathbf{q}$  represents the uncertainty in the parameters of the system. Our approach considers the noise in the signals as uncertainty in parameters of the multiagent system, i.e. every non-zero entry in the Laplacian matrix is uncertain with a nominal value (typically  $l(i, j) \in \mathbb{I}$  depending on the information exchange structure of the multiagent system) and bounded by inferior and superior limits, e.g. let the nominal value of the signal  $x_i = 1$  for some  $i$  with minimum and maximum values between

0.9 and 1.1, then the signal  $x_i$  can be considered as  $x_i(q)$ , where  $q \in [0.9, 1.1]$ . Then, the Laplacian matrix represents the exchange of information between agents, and we are interested in modeling the noise of those signals as uncertainty in the entries of the Laplacian matrix.

Then, the objective is to find a state feedback that effectively stabilizes a forced consensus for multiagent systems to a given reference for all possible  $\mathbf{q}$  within the given bounds. We combine an optimal control approach to design a controller that robustly stabilizes the system and a robust absolute stability approach to analyze and verify if the resulting system is robustly stable.

**REMARK 4.1.**— It is worth mentioning that the optimal control is a design tool that is used to design a controller that robustly stabilizes the system of interest while the robust tools such as the value set are inherently the tools for analysis that are used to analyze and verify if the system is robustly stable.

## 4.2. Robust control design

Now our goal is to show that there is an optimal feedback that robustly stabilizes the forced coordination of multiple agent systems to a given reference. It should be noted that under a similarity transformation the system's characteristic equation, transfer function, eigenvalues, and eigenvectors are all preserved. Let us consider the following multiagent system:

$$\begin{aligned}\dot{\mathbf{x}} &= -\mathcal{L}\mathbf{x} + \mathbf{bu}, \\ y &= \mathbf{c}^T \mathbf{x},\end{aligned}\tag{4.6}$$

which can be rewritten in the canonical form

$$\dot{\boldsymbol{\eta}}(t) = \begin{bmatrix} 0 & 1 & \dots & 0 \\ 0 & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & 1 \\ -q_0 & -q_1 & \dots & -q_{n-1} \end{bmatrix} \boldsymbol{\eta}(t) + \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 1 \end{bmatrix} \mathbf{u}(t),\tag{4.7}$$

where  $\boldsymbol{\eta} \in \mathbb{R}^n$  and  $\mathbf{u} \in \mathbb{R}^m$  are the state variables and control inputs, respectively.  $q_i \in \mathcal{Q}$  are the parametric uncertain values.  $\mathcal{Q}$  is a set that represents the parametric uncertainty defined as:

$$\mathcal{Q} \triangleq \left\{ \mathbf{q} = [q_0 \ \dots \ q_{n-1}]^T : q_i^- \leq q_i \leq q_i^+ \right\}.\tag{4.8}$$

These type of sets are known as boxes.

Let us consider the following representation of the multiagent system [4.5] as given in [LIN 07]:

$$\Sigma_{un} \triangleq \begin{cases} \dot{\boldsymbol{\eta}} = \mathbf{A}(\mathbf{q}^-)\boldsymbol{\eta} + \mathbf{B}\mathbf{u} + \mathbf{B}\Gamma(r)\boldsymbol{\eta}, \\ \mathbf{y} = \mathbf{C}\boldsymbol{\eta}. \end{cases} \quad [4.9]$$

Now, the  $\mathbf{F}$  matrix is defined in such a way that the following condition is satisfied:

$$\Gamma(r)^T \Gamma(r) \leq \mathbf{F} \quad \forall r \in \mathbb{R}, \quad [4.10]$$

where  $\Gamma(r)$  is defined as follows:

$$\Gamma(r) = [q_0^- - q_0 \quad q_1^- - q_1 \quad \dots \quad q_{n-1}^- - q_{n-1}]^T, \quad [4.11]$$

and the nominal system is defined as follows:

$$\Sigma_{nom} \triangleq \begin{cases} \dot{\boldsymbol{\eta}} = \mathbf{A}(\mathbf{q}^-)\boldsymbol{\eta} + \mathbf{B}\mathbf{u}, \\ \mathbf{y} = \mathbf{C}\boldsymbol{\eta}. \end{cases} \quad [4.12]$$

With the above definitions, it is possible to present the following result.

**THEOREM 4.1.–** Consider the system [4.5], and the following control law:

$$\mathbf{u} = -\mathbf{B}^T \mathbf{S} \boldsymbol{\eta}, \quad [4.13]$$

where

$$\mathbf{S}\mathbf{A}(\mathbf{q}^-) + \mathbf{A}(\mathbf{q}^-)^T \mathbf{S} + \mathbf{F} + \mathbf{I} - \mathbf{S}\mathbf{B}\mathbf{B}^T \mathbf{S} = 0, \mathbf{S} > 0, \quad [4.14]$$

then the closed-loop system is robustly stable.

**PROOF.–** By considering the following Lyapunov candidate function:

$$V(\boldsymbol{\eta}) = \min_{\mathbf{u} \in \mathbb{R}} \int_0^\infty (\boldsymbol{\eta}^T \mathbf{F} \boldsymbol{\eta} + \boldsymbol{\eta}^T \boldsymbol{\eta} + \mathbf{u}^T \mathbf{u}) dt, \quad [4.15]$$

it is possible to verify that the proposed control law [4.13] corresponds to the solution of the LQR optimal control problem for the  $\Sigma_{nom}$  system [4.12], considering the cost functional  $V(\boldsymbol{\eta})$ , and the relative weights matrices  $\mathbf{H} = \mathbf{F} + \mathbf{I}$  and  $\mathbf{R} = 1$ . Obviously, the above control law stabilizes the nominal system  $\Sigma_{nom}$ . Next, a proof that the same control law also stabilizes the uncertain system  $\Sigma_{un}$  will be presented. Using the results of the LQR optimal control problem, it is possible to obtain the following solution to the problem in [4.15]:

$$V^*(\boldsymbol{\eta}) = \int_0^\infty (\boldsymbol{\eta}^T F \boldsymbol{\eta} + \boldsymbol{\eta}^T \boldsymbol{\eta} + \boldsymbol{\eta}^T S B B^T S \boldsymbol{\eta}) dt.$$

By definition,  $V^*(\boldsymbol{\eta})$  must satisfy the Hamilton–Jacobi–Bellman equations (see [AND 90], [LEW 95], and [LIN 07]):

$$\boldsymbol{\eta}^T \mathbf{F}\boldsymbol{\eta} + \boldsymbol{\eta}^T \boldsymbol{\eta} + \boldsymbol{\eta}^T \mathbf{S}\mathbf{B}\mathbf{B}^T \mathbf{S}\boldsymbol{\eta} + \left[ \frac{\partial V(\boldsymbol{\eta})}{\partial \boldsymbol{\eta}} \right]^T (\mathbf{A}(\mathbf{q}^-)\boldsymbol{\eta} + \mathbf{B}\mathbf{B}^T \mathbf{S}\boldsymbol{\eta}) = 0, \quad [4.16]$$

$$2\boldsymbol{\eta}^T \mathbf{S}\mathbf{B} + \left[ \frac{\partial V(\boldsymbol{\eta})}{\partial \boldsymbol{\eta}} \right]^T \mathbf{B} = 0. \quad [4.17]$$

Along the trajectories of the system  $\Sigma_{\text{un}}$  [4.9], we have

$$\begin{aligned} \dot{V}(\boldsymbol{\eta}) &= \left[ \frac{\partial V(\boldsymbol{\eta})}{\partial \boldsymbol{\eta}} \right]^T \dot{\boldsymbol{\eta}}, \\ &= \left[ \frac{\partial V(\boldsymbol{\eta})}{\partial \boldsymbol{\eta}} \right]^T [\mathbf{A}(\mathbf{q}^-)\boldsymbol{\eta} + \mathbf{B}\mathbf{u} + \mathbf{B}\Gamma(r)\boldsymbol{\eta}], \\ &= \left[ \frac{\partial V(\boldsymbol{\eta})}{\partial \boldsymbol{\eta}} \right]^T [\mathbf{A}(\mathbf{q}^-)\boldsymbol{\eta} + \mathbf{B}\mathbf{B}^T \mathbf{S}\boldsymbol{\eta}] + \left[ \frac{\partial V(\boldsymbol{\eta})}{\partial \boldsymbol{\eta}} \right]^T \mathbf{B}\Gamma(r)\boldsymbol{\eta}. \end{aligned}$$

Then, [4.16] and [4.17] lead to

$$\begin{aligned} \dot{V}(\boldsymbol{\eta}) &= -\boldsymbol{\eta}^T \mathbf{F}\boldsymbol{\eta} - \boldsymbol{\eta}^T \boldsymbol{\eta} - \boldsymbol{\eta}^T \mathbf{S}\mathbf{B}\mathbf{B}^T \mathbf{S}\boldsymbol{\eta} - 2\boldsymbol{\eta}^T \mathbf{S}\mathbf{B}\Gamma(r)\boldsymbol{\eta}, \\ &= -\boldsymbol{\eta}^T \mathbf{F}\boldsymbol{\eta} - \boldsymbol{\eta}^T \boldsymbol{\eta} - \boldsymbol{\eta}^T \mathbf{S}\mathbf{B}\mathbf{B}^T \mathbf{S}\boldsymbol{\eta} - 2\boldsymbol{\eta}^T \mathbf{S}\mathbf{B}\Gamma(r)\boldsymbol{\eta} \pm \boldsymbol{\eta}^T \mathbf{B}^T(r)\mathbf{B}(r)\boldsymbol{\eta}, \\ &= -\boldsymbol{\eta}^T [\mathbf{F} - \mathbf{B}^T(r)\mathbf{B}(r)] \boldsymbol{\eta} - \boldsymbol{\eta}^T \boldsymbol{\eta} - \boldsymbol{\eta}^T [\mathbf{B}^T \mathbf{S} + \mathbf{B}(r)]^T [\mathbf{B}^T \mathbf{S} + \mathbf{B}(r)] \boldsymbol{\eta}, \end{aligned}$$

from condition [4.10] it follows

$$\dot{V}(\boldsymbol{\eta}) \leq -\boldsymbol{\eta}^T \boldsymbol{\eta}. \quad [4.18]$$

Then,  $\dot{V}(\boldsymbol{\eta}) < 0$  for all  $\boldsymbol{\eta} \neq 0$  and  $\dot{V}(\boldsymbol{\eta}) = 0$  if and only if  $\boldsymbol{\eta} = 0$ , which ends the proof.

### 4.3. Robust stability analysis

In engineering problems, it is quite often to find ourselves facing the problem that actuators are always saturated. This makes sense since there is no actuator that can produce infinite force, torque, etc. Then, it is necessary to introduce a saturation in the control action to protect physical actuators from damage. When a saturation function is included, a new question arise: Will the dynamical system still be stable when saturated control is used? More precisely, we are interested in verifying the robust absolute stability of multiagent systems of the form:

$$\Upsilon: \dot{\mathbf{x}} = \mathbf{A}(\mathbf{q})\mathbf{x} + \mathbf{B}(\mathbf{q})\mathbf{u}, \quad [4.19]$$

$$\mathbf{y} = \mathbf{C}(\mathbf{q})\mathbf{x}, \quad [4.20]$$

$$\mathbf{u} = -\varpi(t, \mathbf{y}), \quad [4.21]$$

with uncertainty in the linear part represented by  $\mathbf{q}$  and  $\varpi(t, \mathbf{y})$  being a memoryless nonlinear function contained in a region called sector  $[0, k]$ . It is important to note that the information exchange graph of a multiagent system is represented by a Laplacian matrix denoted as  $\mathbf{A}(\mathbf{q}) = -\mathcal{L} = (\mathbf{q})$  in [4.19].  $\varpi(t, \mathbf{y})$  represents a nonlinear saturation on the feedback control. We recall that a leader-based multiagent system structure is adopted. This leads to the Lur'e system form given by  $\Upsilon$ -system.  $G(s, \mathbf{q}) = \mathbf{C}(\mathbf{q})(s\mathbf{I} - \mathbf{A}(\mathbf{q}))^{-1}\mathbf{B}(\mathbf{q})$  represents the transfer function of the multiagent system in  $\Upsilon$  considering polynomic parametric uncertainty that is defined as follows [BAR 94]:

**DEFINITION 4.1.**— *A polynomic plant is a transfer function with parametric uncertainty that has the following structure:*

$$G(s, \mathbf{q}) = \frac{n(s, \mathbf{q})}{d(s, \mathbf{q})} = \frac{\sum_{i=0}^m a_i(\mathbf{q})s^i}{\sum_{i=0}^n b_i(\mathbf{q})s^i} \forall \mathbf{q} \in \mathcal{Q} \quad [4.22]$$

where  $a_i(\mathbf{q})$  and  $b_i(\mathbf{q})$  are polynomic functions of the vector  $\mathbf{q}$ .

Without loss of generality, it will be assumed that  $q_i^- \geq 0$ . Nevertheless, if this condition is not satisfied, it is always possible to make a linear transformation to make this condition hold. The name of polynomic plants is used because the coefficients of the transfer function are uncertain values that have polynomic structures. The absolute stability property when the linear part has uncertainty is called robust absolute stability and is satisfied if the system, shown in the equation [4.19], is absolutely stable for all members of the interval plant  $G(s, \mathbf{q})$ .

**REMARK 4.2.**— It is important to mention that a multiagent system is composed of multiple independent dynamical systems, e.g. manipulator robots, mobile robots, etc., interconnected using a static or dynamic communication topology. Our idea is to transform the original system into the form  $\Upsilon$  to verify the robust absolute stability property of the multiagent system.

#### 4.3.1. Robust strict positive realness

The concepts of robust SPR, robust absolute stability, and its application to multiagent systems are the basis of this analysis; thus, we elaborate on them next. To introduce the definition of SPR functions, it is necessary to first present the definition of PR functions [IOA 96].

**DEFINITION 4.2.**— *A rational function  $G(s)$  of the complex variable  $s = \sigma + j\omega$  is called PR if*

- $G(s)$  is real for real  $s$ ;
- $\text{Re}[G(s)] \geq 0$  for all  $\text{Re}[s] > 0$ .

Now, it is possible to present the following definition:

**DEFINITION 4.3.**— Assume that  $G(s)$  is not identically zero for all  $s$ . Then,  $G(s)$  is SPR if  $G(s - \epsilon)$  is PR for some  $\epsilon > 0$ .

The robust version of the SPR property is defined as follows:

**DEFINITION 4.4.**— A polynomic plant is robustly SPR if  $G(s, \mathbf{q})$  is SPR for all  $\mathbf{q} \in \mathcal{Q}$ .

It is clear that the polynomic plant defined in [4.22] represents an infinite number of transfer functions. Therefore, the robust SPR property results are impossible to be verified directly.

However, the objective is to find a method to verify this property using finite tests. Before introducing a finding that helps to build our main result, it is necessary to present the following definition:

$$\mathfrak{h}(\omega^2, \mathbf{q}) = |d(j\omega, \mathbf{q})|^2, \quad [4.23]$$

$$\mathfrak{g}(\omega^2, \mathbf{q}) = n(j\omega, \mathbf{q})d(-j\omega, \mathbf{q}) + n(-j\omega, \mathbf{q})d(j\omega, \mathbf{q}). \quad [4.24]$$

It can be seen that the functions  $\mathfrak{h}(\omega, \mathbf{q})$  and  $\mathfrak{g}(\omega, \mathbf{q})$  are polynomic functions of  $\omega$  with uncertain coefficients which are again polynomic functions of  $q_i$  in the vector  $\mathbf{q}$ . Hence, the following theorem presents necessary and sufficient conditions to verify the robust SPR property [STI 00]:

**THEOREM 4.2.**— A polynomic plant is robustly SPR if and only if  $G(s, \mathbf{q})$  is stable for some  $\mathbf{q} \in \mathcal{Q}$ , and

- 1)  $\mathfrak{h}(\omega, \mathbf{q})$  is positive for all  $\mathbf{q} \in \mathcal{Q}$  and  $\omega \in (0, \infty)$ ,
- 2)  $\mathfrak{g}(\omega, \mathbf{q})$  is positive for all  $\mathbf{q} \in \mathcal{Q}$  and  $\omega \in (0, \infty)$ .

This theorem presents the necessary and sufficient conditions to verify the robust SPR property. To verify the positivity of these functions, a sign decomposition technique is used. This method analyzes the positivity of a multivariable real polynomic function by its decomposition into this positive and negative parts (see [ELI 99]). The sign decomposition is defined as follows:

**DEFINITION 4.5.**— Let  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  be a continuous function and  $\mathcal{Q} \subset P \subset \mathbb{R}^n$  a convex subset,  $f(\cdot)$  has sign decomposition in  $\mathcal{Q}$  if there exist two bounded non-growing functions  $f_n(\cdot) \geq 0$ ,  $f_p(\cdot) \geq 0$ , such that  $f(\mathbf{q}) = f_p(\mathbf{q}) - f_n(\mathbf{q})$  for all  $\mathbf{q} \in \mathcal{Q}$ . These functions will be called positive part of the function  $f_p(\cdot)$  and the negative part of the function  $f_n(\cdot)$ .

Here, the  $P$  set is considered to be a positive convex cone (see [ELI 99]). Now, the maximum ( $v_{\max}$ ) and minimum ( $v_{\min}$ ) vertices of the uncertainty set  $\mathcal{Q}$  will be defined as  $\|v_{\min}\|_2 = \min_{\mathbf{q} \in \mathcal{Q}} \|\mathbf{q}\|_2$  and  $\|v_{\max}\|_2 = \max_{\mathbf{q} \in \mathcal{Q}} \|\mathbf{q}\|_2$ . We can see that the  $v_{\min}$  and  $v_{\max}$  vertices are the minimum and maximum Euclidian of the uncertain set  $\mathcal{Q}$ . Then, to determine if a function with sign decomposition is positive, we need to verify if  $\mu = f_p(v_{\min}) - f_n(v_{\max})$  is greater than zero, this is only a sufficient condition that is relaxed with the following theorem [ELI 01]:

**THEOREM 4.3.** – *Let  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  be a continuous function with sign decomposition in  $\mathcal{Q}$  such that  $\mathcal{Q} \subset P \subset \mathbb{R}^n$  is a box with minimum and maximum Euclidian vertices  $v_{\min}, v_{\max}$ , then the function  $f(\mathbf{q}) > 0$  in  $\mathcal{Q}$  if and only if there exist some  $\Gamma^j$  box sets, such that  $\mathcal{Q} = \bigcup_j \Gamma^j$  and  $\mu_j = f_p(v_{\min}^j) - f_n(v_{\max}^j)$  are greater than zero for each  $\Gamma^j$ .*

A more general result of this theorem is presented in [ELI 99], where a graphical version is included. However, for our purposes we only need this simplified version.

It was mentioned that the robust SPR condition of transfer functions will be verified using the sign decomposition approach. However, it is important to mention that this approach needs full knowledge of the low and high boundaries of the uncertainty, and as was seen in Theorem 4.3,  $\omega$  is an unbounded parameter. Therefore, it is necessary to undertake a previous special operation before applying the sign decomposition method. This operation consists of determining the limits for the bound of  $\omega$ . To do this, we will define the next function:

$$\mathfrak{h}_{\min}(\omega) = \mathfrak{h}_p(\omega, \mathbf{q}^-) - \mathfrak{h}_n(\omega, \mathbf{q}^+), \quad [4.25]$$

$$\mathfrak{g}_{\min}(\omega) = \mathfrak{g}_p(\omega, \mathbf{q}^-) - \mathfrak{g}_n(\omega, \mathbf{q}^+), \quad [4.26]$$

where  $\mathfrak{h}_p(\cdot)$ ,  $\mathfrak{h}_n(\cdot)$ ,  $\mathfrak{g}_p(\cdot)$ , and  $\mathfrak{g}_n(\cdot)$  are the negative and positive parts of  $\mathfrak{h}(\omega, \mathbf{q})$  and  $\mathfrak{g}(\omega, \mathbf{q})$ , respectively, as it was defined in the Definition 4.5:  $\mathbf{q}^- = [q_1^- \ \cdots \ q_m^-]^T$  and  $\mathbf{q}^+ = [q_1^+ \ \cdots \ q_m^+]^T$ . In the previous definition, it is clear that the following conditions are satisfied:

$$\mathfrak{h}_{\min}(\omega) \leq \mathfrak{h}(\omega, \mathbf{q}) \ \forall \omega \in (0, \infty), \ \mathbf{q} \in \mathcal{Q}, \quad [4.27]$$

$$\mathfrak{g}_{\min}(\omega) \leq \mathfrak{g}(\omega, \mathbf{q}) \ \forall \omega \in (0, \infty), \ \mathbf{q} \in \mathcal{Q}, \quad [4.28]$$

and therefore if  $\mathfrak{h}_{\min}(\omega)$  and  $\mathfrak{g}_{\min}(\omega)$  are greater than zero, also  $\mathfrak{h}(\omega, \mathbf{q})$  and  $\mathfrak{g}(\omega, \mathbf{q})$  will be greater than zero. It is important to note that due to the shape of  $\mathfrak{h}_{\min}(\omega)$  and  $\mathfrak{g}_{\min}(\omega)$  it is possible to set a minimum value  $\omega^-$  and maximum value  $\omega^+$  so that the functions may take negative values, and thus the function only has the possibility to be negative when they are within these intervals. This allows us to obtain the limits we were looking for the  $\omega$  parameter in both the functions. These values usually correspond to some roots of  $\mathfrak{h}_{\min}(\omega)$  and  $\mathfrak{g}_{\min}(\omega)$ , respectively, and can be gotten

graphically. With the limits of  $\omega$ , it is possible to define the following sets in order to use the sign decomposition approach:

$$\mathcal{V} = [\omega_{\mathfrak{h}}^-, \omega_{\mathfrak{h}}^+] \times \mathcal{Q}, \quad [4.29]$$

$$\mathcal{U} = [\omega_{\mathfrak{g}}^-, \omega_{\mathfrak{g}}^+] \times \mathcal{Q}. \quad [4.30]$$

Applying Theorems 4.2 and 4.3, we can present the next result [GUE 05], [GUE 06].

**THEOREM 4.4.** – *The polynomic plant of system  $\Upsilon$  is robustly SPR if and only if:*

- 1) *there exist some  $\Omega^j$  box sets, such that  $\mathcal{V} = \bigcup_j \Omega^j$  and  $\omega_j = \mathfrak{h}_p(v_{\min}^j) - \mathfrak{h}_n(v_{\max}^j)$  are greater than zero for each  $\mu_j$ ,*
- 2) *there exist some  $\Phi^j$  box sets, such that  $\mathcal{U} = \bigcup_j \Phi^j$  and  $\phi_j = \mathfrak{g}_p(v_{\min}^j) - \mathfrak{g}_n(v_{\max}^j)$  are greater than zero for each  $\sigma_j$ ,*

for all  $\nu \in \mathcal{V}$  and  $v \in \mathcal{U}$  where  $\mathcal{V}$  and  $\mathcal{U}$  are the sets defined in [4.29]. This theorem can be applied to obtain the robust SPR property using an algorithm codified in C language.

#### 4.3.2. Robust absolute stability

To verify the robust absolute stability of the Lur'e system, we present the following lemma:

**LEMMA 4.1.** – *Consider the system  $\Upsilon$ , where  $\varpi(t, \mathbf{y})$  satisfies the sector  $[0, k]$  condition. Then, the system  $\Upsilon$  is robustly absolutely stable if  $z_f(s, \mathbf{q}) = 1 + kG(s, \mathbf{q})$  is SPR for all  $\mathbf{q} \in \mathcal{Q}$ .*

It is important to note that this condition implies verifying the robust SPR condition of the fictitious transfer function  $z_f(s, \mathbf{q})$ . Hence, the robust absolute stability problem is transformed to determine if  $z_f(s, \mathbf{q})$  is robustly SPR.

From Lemma 4.1 it is seen that the condition to ensure the robust absolute stability consists of determining if the  $z_f(s, \mathbf{q})$  is a robust SPR transfer function. In addition, when the relative degree of the polynomic plant  $G(s, \mathbf{q})$  is known to be equal to 1, we assume that the  $z_f(s, \mathbf{q})$  is also a polynomic plant with relative degree equal to zero. Then, it is possible to use the previous result presented in Theorem 4.4. Before presenting the result, it is necessary to introduce the following definitions:

$$z_f(s, \mathbf{q}) = \frac{\vartheta(s, \mathbf{q})}{\delta(s, \mathbf{q})}, \quad [4.31]$$

$$\alpha(\omega, \mathbf{q}) = |\delta(j\omega, \mathbf{q})|^2, \quad [4.32]$$

$$\beta(\omega, \mathbf{q}) = \vartheta(j\omega, \mathbf{q})\delta(-j\omega, \mathbf{q}) + \vartheta(-j\omega, \mathbf{q})\delta(j\omega, \mathbf{q}). \quad [4.33]$$

The sets  $\mathcal{V}$  and  $\mathcal{U}$  are defined as in [4.29]. Now, the result of robust absolute stability with polynomic uncertainty will be presented in the next theorem [GUE 05], [GUE 06].

**THEOREM 4.5.** – Consider the nonlinear system from  $\Upsilon$ , where  $\varpi(t, y)$  satisfies the sector  $[0, k]$  condition and  $z_f(s, q) = 1 + kG(s, \mathbf{q}) = \vartheta(s, \mathbf{q})/\delta(s, \mathbf{q})$  is analytic in  $\text{Re}[s] \geq 0$ . Then, the Lur'e system is robustly absolutely stable if

- 1) there exist some  $\Lambda^j$  box sets, such that  $\mathcal{V} = \bigcup_j \Lambda^j$  and  $\lambda_j = \alpha_p(v_{\min}^j) - \alpha_n(v_{\max}^j)$  are greater than zero for each  $\lambda_j$ ,
- 2) there exist some  $\Theta^j$  box sets, such that  $\mathcal{U} = \bigcup_j \Theta^j$  and  $\theta_j = \beta_p(v_{\min}^j) - \beta_n(v_{\max}^j)$  are greater than zero for each  $\theta_j$ .

#### 4.4. Robust stability of time-delay systems

Now, we consider the case of time delays due to sensor information process, actuator time delay, etc. Considering a time delay  $\tau$  in the input, the system [4.5] can be rewritten in the canonical form as:

$$\dot{\boldsymbol{\eta}}(t) = \begin{bmatrix} 0 & 1 & \dots & 0 \\ 0 & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & 1 \\ -q_0 & -q_1 & \dots & -q_{n-1} \end{bmatrix} \boldsymbol{\eta}(t) + \begin{bmatrix} 0 \\ 0 \\ \vdots \\ r_0 \end{bmatrix} \mathbf{u}(t - \tau). \quad [4.34]$$

Then, this system is robustly stable if the same optimal control law  $\mathbf{u} = \mathbf{K}\boldsymbol{\eta}$  is used considering a maximum time delay  $\tau_{\max}$ .

The uncertain time-delay system [4.34] has the following characteristic equation:

$$p(s, \mathbf{q}, \mathbf{r}, e^{-\tau s}) = s^n + [q_{n-1}^-, q_{n-1}^+]s^{n-1} + \dots + [q^-, q^+]s + [q_0^-, q_0^+] \quad [4.35] \\ + (k_{n-1}s^{n-1} + \dots + k_1s + k_0)e^{-[0, \tau_{\max}]},$$

where  $\mathbf{q}$  is defined as in [4.8] and  $\mathbf{r}$  is defined as follows:

$$\mathcal{R} \triangleq \left\{ \mathbf{r} = [r_0 \ \dots \ r_{n-1}]^T : r_i^- \leq r_i \leq r_i^+ \right\}. \quad [4.36]$$

These kind of functions are known as quasi-polynomials. It is clear that the above characteristic equation [4.35] represents an infinite number of quasi-polynomials that have to be considered to verify the robust stability property. This family is defined as follows:

$$\mathcal{P}_\tau \triangleq \left\{ p(s, \mathbf{q}, \mathbf{r}, e^{-\tau s}) : \mathbf{q} \in \mathcal{Q}, \right. \\ \left. \mathbf{r} \in \mathcal{R}; \tau \in [0, \tau_{\max}] \right\} \quad [4.37]$$

where  $\mathcal{Q}$  and  $\mathcal{R}$  represent the set of uncertainty (see [BAR 94]). Now, we define the value set as follows:

**DEFINITION 4.6.– (Value set):** The value set of  $\mathcal{P}_\tau$ , denoted by  $V_\tau(\omega)$ , is the graph in the complex plane of  $p(s, \mathbf{q}, \mathbf{r}, e^{-\tau s})$  when  $s = j\omega$

$$V_\tau(\omega) \triangleq \left\{ p(s, \mathbf{q}, \mathbf{r}, e^{-j\omega\tau}) : \mathbf{q} \in \mathcal{Q}, \begin{array}{l} \mathbf{r} \in \mathcal{R}; \tau \in [0, \tau_{\max}]; \omega \in \mathbb{R}. \end{array} \right\} \quad [4.38]$$

It is clear that the value set of  $\mathcal{P}_\tau$  is a set of complex numbers plotted on the complex plane for values of  $\mathbf{q}$ ,  $\mathbf{r}$ ,  $\omega$ , and  $\tau$  inside the defined boundaries. Next, the zero exclusion principle is presented in order to verify the robust condition [BAR 94].

**LEMMA 4.2.–** Consider the characteristic equation [4.35], also called quasi-polynomials. Suppose that [4.35] has at least one stable member. Then the robust stability property of the control system is guaranteed if and only if

$$0 \notin V_\tau(\omega) \quad \forall \omega \geq 0. \quad [4.39]$$

#### 4.5. Application to multiagent systems

To illustrate the application of the previous results, numerical simulations of several multiagent dynamic models have been run using Matlab Simulink™. Coupling gain between agents has been set to unit gain; as noted in previous sections, 10% uncertainty was considered in every signal. A representative Gaussian noise is considered in measured signals.

The proposed algorithm finds a constant output feedback that effectively synchronizes every agent (signal). Also, it allows the multiagent system to follow a reference signal or trajectory. In this case, we propose a constant reference.

##### 4.5.1. Cyclic topology

The multiagent system with cyclic topology of information exchange is one of the most frequently used applications in coordination and consensus on multiagent systems. Let us consider a three-agent system, with cyclic topology of information exchange given by:

$$\dot{\mathbf{x}}(t) = \begin{bmatrix} -1 & 1 & 0 \\ 0 & -1 & 0 \\ 1 & 0 & -1 \end{bmatrix} \mathbf{x}(t) + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \mathbf{u}(t). \quad [4.40]$$

The multiagent system in the canonical form is given by:

$$\dot{\boldsymbol{\eta}}(t) = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & -3 & -3 \end{bmatrix} \boldsymbol{\eta}(t) + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \mathbf{u}(t). \quad [4.41]$$

We will consider a  $[10\%, 10\%, 10\%]$  uncertainty in the last row coefficients of the state matrix. As a result of this assumption, the following matrix is obtained:

$$\mathbf{A}(\mathbf{q}) = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -q_0 & -q_1 & -q_2 \end{bmatrix}, \quad [4.42]$$

where  $q_0 = 0$ ,  $q_1 = [-3.3, -2.7]$ , and  $q_2 = [-3.3, -2.7]$ .

Despite the fact that the structure of the uncertainty in [4.42] may be multilinear, polynomial, etc., it is always possible to lump the uncertainty such that uncertainty structure in [4.42] becomes independent interval uncertainty. Then,

$$\mathbf{A}(\mathbf{q}^-) = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & -3.3 & -3.3 \end{bmatrix},$$

$$\mathbf{r}^T(r) = [r_1 \ r_2 \ r_3],$$

where  $r_1 = [0, 0]$ ,  $r_2 = [0, 0.6]$ , and  $r_3 = [0, 0.6]$ .

The  $\mathbf{F}$  matrix is given by:

$$\mathbf{F} = \begin{bmatrix} 0.00 & 0.00 & 0.00 \\ 0.00 & 0.36 & 0.36 \\ 0.00 & 0.36 & 0.36 \end{bmatrix}. \quad [4.43]$$

Then, the optimal control law is given by:

$$\mathbf{K} = [1.0000 \ 1.1519 \ 0.5168]. \quad [4.44]$$

After a simple inverse transformation of the optimal control law [4.44], the resulting optimal control law is applied to the multiagent system [4.40]. The performance of the robust optimal control is shown in Figure 4.1.

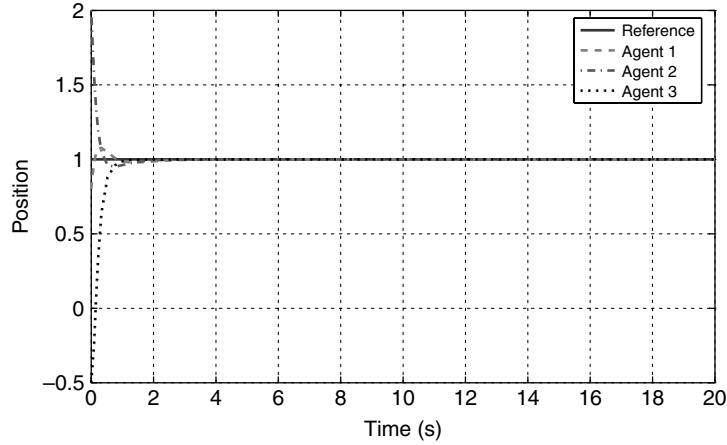
Then, the uncertain time-delay system of the form [4.34] has the following characteristic equation:

$$p(s, \mathbf{q}, \mathbf{r}, e^{-\tau s}) = s^3 + [2.7, 3.3]s^2 + [2.7, 3.3]s + (0.5168s^2 + 1.1519s + 1)e^{-[0, 4.3]}. \quad [4.45]$$

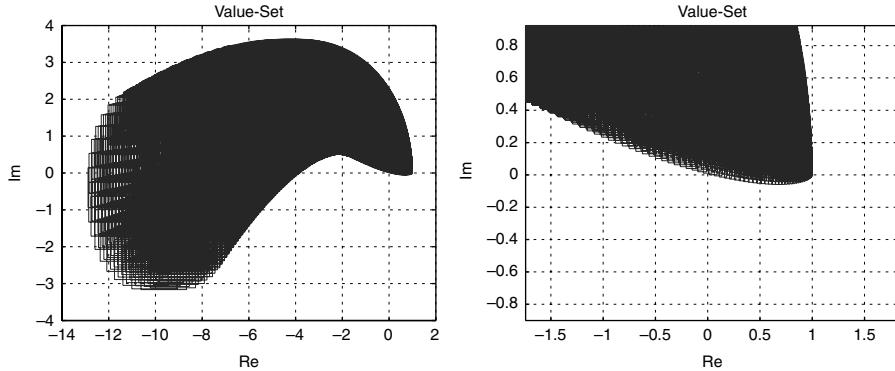
It is clear that the value set of  $\mathcal{P}_\tau$  is a set of complex numbers plotted on the complex plane for values of  $\mathbf{q}$ ,  $\mathbf{r}$ ,  $\omega$ , and  $\tau$  inside the defined boundaries. The results presented in [ROM 95] and [ROM 97] permit building the value set  $V_\tau(\omega)$  for the characteristic equation [4.45] and is presented in Figures 4.2(a) and (b). It can be noted that the zero is not included in the value set  $V_\tau(\omega)$ . Then, the multiagent system with cyclic topology of information exchange [4.40] and  $\tau < 4.3$  is robustly stable.

## 64 Flight Formation Control

The behavior of the multiagent system with cyclic topology is shown in Figures 4.3 and 4.4.



**Figure 4.1.** Multiagent system

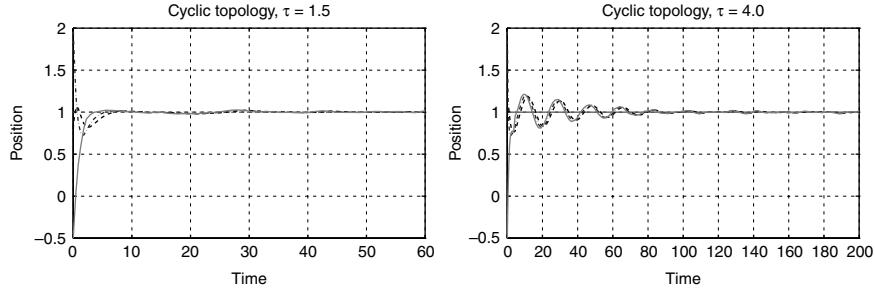


**Figure 4.2.** (a) Value set of the uncertain time-delay system [4.40];  
(b) Value set zoom

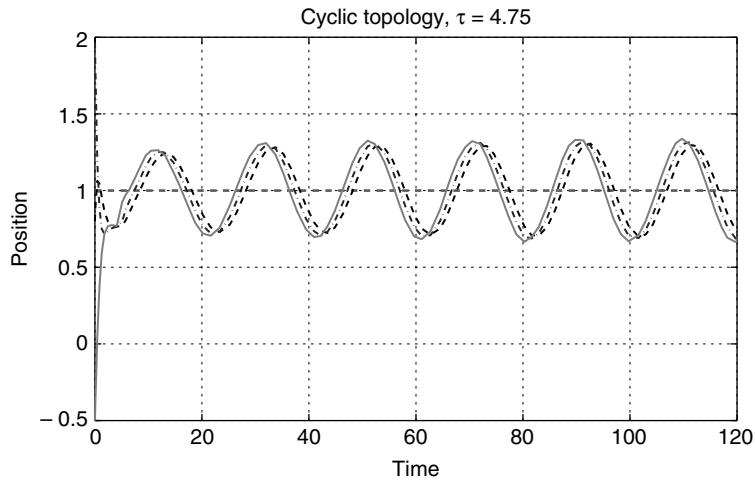
Now, let us consider that the input is saturated and the saturation function satisfies the sector condition. We will consider the case of robust absolute stability of the multiagent system as follows. First, let us consider the transfer function of the above system

$$G(s, \mathbf{q}) = \frac{s^2 + s + 1}{s^3 + \bar{q}_1 s^2 + \bar{q}_2 s + 1}, \quad [4.46]$$

where  $\bar{q}_1 = [3.2168, 3.8168]$  and  $\bar{q}_2 = [3.8591, 4.4591]$ .



**Figure 4.3.** (a) Cyclic topology of information exchange with  $\tau = 1.5$ ; (b) Cyclic topology of information exchange with  $\tau = 4.0$



**Figure 4.4.** Cyclic topology of information exchange with  $\tau = 4.75$

The fictitious transfer function has the following form:

$$z_f(s, \mathbf{q}) = \frac{s^3 + (\bar{q}_1 + 1)s^2 + (\bar{q}_2 + 1)s + 2}{s^3 + \bar{q}_1 s^2 + \bar{q}_2 s + 1}, \quad [4.47]$$

where the positive and negative parts of  $\mathfrak{h}(\omega, \mathbf{q})$  and  $\mathfrak{g}(\omega, \mathbf{q})$  functions are the following:

$$\mathfrak{h}_p(\omega, \mathbf{q}) = \omega^3 + \bar{q}_1^2 \omega^2 + \bar{q}_2^2 \omega + 1, \quad [4.48]$$

$$\mathfrak{h}_n(\omega, \mathbf{q}) = 2\bar{q}_2^2 \omega^2 + 2\bar{q}_1^2 \omega, \quad [4.49]$$

$$\mathfrak{g}_p(\omega, \mathbf{q}) = 2\omega^3 + (2\bar{q}_1^2 + 2\bar{q}_1)\omega^2 + (2\bar{q}_2^2 + 2\bar{q}_2)\omega + 4, \quad [4.50]$$

$$\mathfrak{g}_n(\omega, \mathbf{q}) = (4\bar{q}_2 + 2)\omega^2 + (6\bar{q}_1 + 2)\omega. \quad [4.51]$$

Using the equations defined in [4.25] and [4.26], we get

$$\mathfrak{h}_{\min}(\omega) = \omega^3 - 29.4176\omega^2 - 14.2433\omega + 1, \quad [4.52]$$

$$\mathfrak{g}_{\min}(\omega) = 2\omega^3 + 7.2932\omega^2 + 12.5873\omega + 1. \quad [4.53]$$

On the one hand, it is clear that the  $\mathfrak{g}_{\min}(\omega)$  function is positive, because all of its coefficients are positive and the frequency  $\omega$  belongs to the interval  $(0, \infty)$ . On the other hand, from  $\mathfrak{h}_{\min}(\omega)$  function it is possible to get the interval  $[0.062, 29.8929]$ . Then, the set  $\mathcal{V}$  is defined as:

$$\mathcal{V} = [0.062, 29.8929] \times [3.2168, 3.8168] \times [3.8591, 4.4591]. \quad [4.54]$$

Now, applying Theorem 4.5 we obtain the following result:

$$\mathfrak{h}(\omega, \mathbf{q}) > 102.2151. \quad [4.55]$$

With this, we can conclude that the three-agent system with cyclic topology of information exchange is robustly absolutely stable.

#### 4.5.2. Chain topology

A three-agent system with chain topology of information exchange is given by:

$$\dot{\mathbf{x}}(t) = \begin{bmatrix} -1 & 1 & 0 \\ 1 & -2 & 1 \\ 0 & 1 & -1 \end{bmatrix} \mathbf{x}(t) + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \mathbf{u}(t). \quad [4.56]$$

The multiagent system, in terms of the canonical form, is given by:

$$\dot{\boldsymbol{\eta}}(t) = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & -3 & -4 \end{bmatrix} \boldsymbol{\eta}(t) + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \mathbf{u}(t). \quad [4.57]$$

Let us consider a  $[10\%, 10\%, 10\%]$  uncertainty in the last row coefficients of the state matrix. As a result of this assumption, the following matrix is obtained:

$$\mathbf{A}(\mathbf{q}) = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -q_0 & -q_1 & -q_2 \end{bmatrix}, \quad [4.58]$$

where  $q_0 = [0, 0]$ ,  $q_1 = [-3.3, -2.7]$ , and  $q_2 = [-4.4, -3.6]$ .

Then,

$$\mathbf{A}(\mathbf{q}^-) = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & -3.3 & -4.4 \end{bmatrix},$$

$$\mathbf{\Gamma}^T(r) = [r_1 \ r_2 \ r_3],$$

where  $r_1 = 0$ ,  $r_2 = [0, 0.6]$ , and  $r_3 = [0, 0.8]$ .

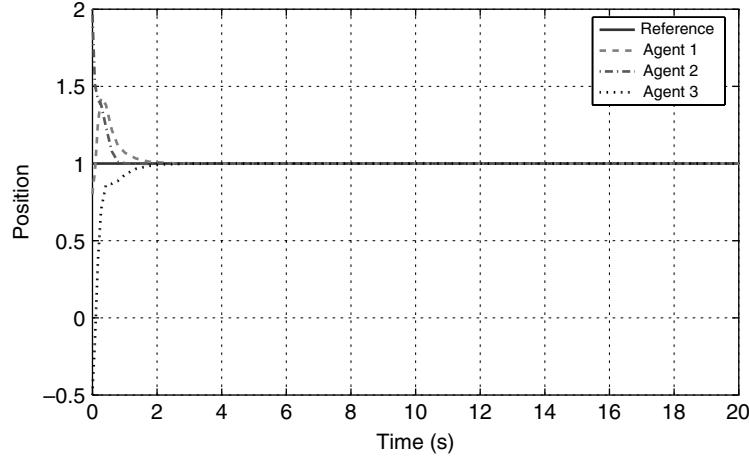
The  $\mathbf{F}$  matrix is given by:

$$\mathbf{F} = \begin{bmatrix} 0.00 & 0.00 & 0.00 \\ 0.00 & 0.36 & 0.48 \\ 0.00 & 0.48 & 0.64 \end{bmatrix}. \quad [4.59]$$

Thus, the optimal control law is given by:

$$\mathbf{K} = [1 \ 1.3908 \ 0.4766]. \quad [4.60]$$

After a simple inverse transformation of the optimal control law [4.60], the resulting optimal control law is applied to the multiagent system [4.56]. The performance of the robust optimal control is shown in Figure 4.5.

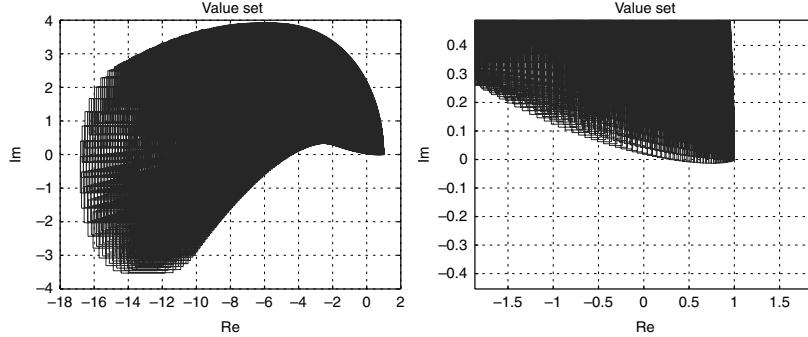


**Figure 4.5.** Multiagent system with chain topology

Then, the uncertain time-delay system of the form [4.34] has the following characteristic equation:

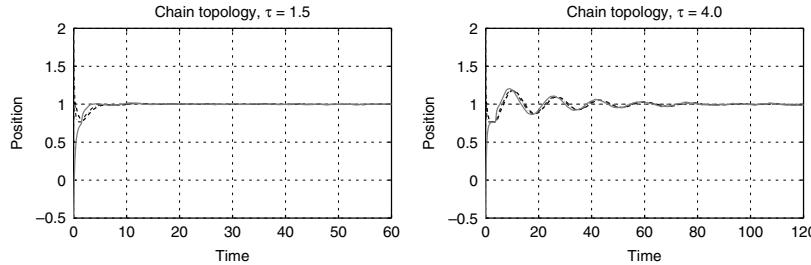
$$\begin{aligned} p(s, \mathbf{q}, \mathbf{r}, e^{-\tau s}) = & s^3 + [3.6, 4.4]s^2 + [2.7, 3.3]s \\ & + (0.4766s^2 + 1.3908s + 1)e^{-[0, 4.2]} \end{aligned} \quad [4.61]$$

It is clear that the value set of  $\mathcal{P}_\tau$  is a set of complex numbers plotted on the complex plane for values of  $\mathbf{q}$ ,  $\mathbf{r}$ ,  $\omega$ , and  $\tau$  inside the defined boundaries. The results presented in [ROM 95] and [ROM 97] permit building the value set  $V_\tau(\omega)$  for the characteristic equation [4.61] and is presented in Figures 4.6(a) and (b).



**Figure 4.6.** (a) Value set for the uncertain time-delay system [4.40]; (b) Value set zoom

It can be noted that the zero is not included in the value set  $V_\tau(\omega)$ . Then, the multiagent system with chain information exchange topology [4.56] and  $\tau < 4.2$  is robustly stable. The behavior of the multiagent system with chain topology is shown in Figures 4.7 and 4.8.

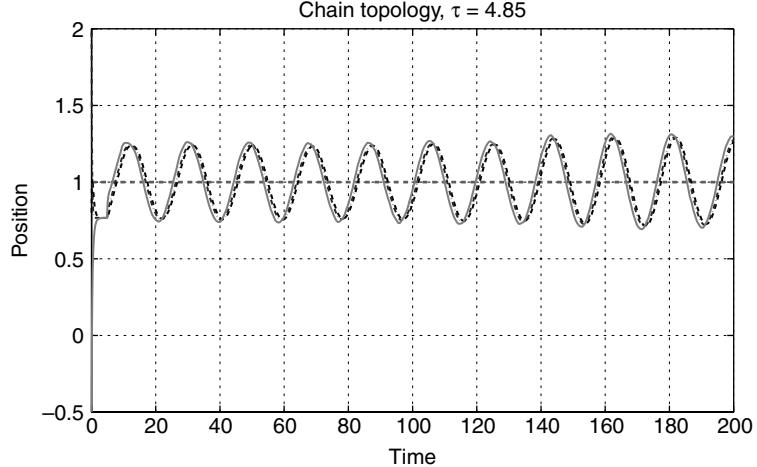


**Figure 4.7.** (a) Chain topology of information exchange with  $\tau = 1.5$ ; (b) Chain topology of information exchange with  $\tau = 4.0$

Now, let us consider that the input is saturated and the saturation function satisfies the sector condition. We will consider the case of robust absolute stability of the multiagent system as follows. The transfer function of the above system is as follows:

$$G(s, \mathbf{q}) = \frac{s^2 + s + 1}{s^3 + \bar{q}_1 s^2 + \bar{q}_2 s + 1}, \quad [4.62]$$

where  $\bar{q}_1 = [4.0766, 4.8766]$  and  $\bar{q}_2 = [4.0908, 4.6908]$ .



**Figure 4.8.** Chain topology of information exchange with  $\tau = 4.85$

The fictitious transfer function has the following form:

$$z_f(s, \mathbf{q}) = \frac{s^3 + (\bar{q}_1 + 1)s^2 + (\bar{q}_2 + 1)s + 2}{s^3 + \bar{q}_1 s^2 + \bar{q}_2 s + 1}, \quad [4.63]$$

where the positive and negative parts of  $\mathfrak{h}(\omega, \mathbf{q})$  and  $\mathfrak{g}(\omega, \mathbf{q})$  functions are the following:

$$\mathfrak{h}_p(\omega, \mathbf{q}) = \omega^3 + \bar{q}_1^2 \omega^2 + \bar{q}_2^2 \omega + 1, \quad [4.64]$$

$$\mathfrak{h}_n(\omega, \mathbf{q}) = 2\bar{q}_2^2 \omega^2 + 2\bar{q}_1^2 \omega, \quad [4.65]$$

$$\mathfrak{g}_p(\omega, \mathbf{q}) = 2\omega^3 + (2\bar{q}_1^2 + 2\bar{q}_1)\omega^2 + (2\bar{q}_2^2 + 2\bar{q}_2)\omega + 4, \quad [4.66]$$

$$\mathfrak{g}_n(\omega, \mathbf{q}) = (4\bar{q}_2 + 2)\omega^2 + (6\bar{q}_1 + 2)\omega. \quad [4.67]$$

Using the equations defined in [4.25] and [4.26], we get

$$\mathfrak{h}_{\min}(\omega) = \omega^3 - 30.8278\omega^2 - 27.3885\omega + 1, \quad [4.68]$$

$$\mathfrak{g}_{\min}(\omega) = 2\omega^3 + 20.1445\omega^2 + 11.2457\omega + 1. \quad [4.69]$$

On the one hand, it is clear that the  $\mathfrak{g}_{\min}(\omega)$  function is positive, because all of its coefficients are positive and the frequency  $\omega$  belongs to the interval  $(0, \infty)$ . On the other hand, from  $\mathfrak{h}_{\min}(\omega)$  function it is possible to get the interval  $[0.0351, 31.6911]$ . Then, the set  $\mathcal{V}$  is defined as:

$$\mathcal{V} = [0.0351, 31.6911] \times [4.0766, 4.8766] \times [4.0908, 4.6908]. \quad [4.70]$$

Now, applying Theorem 4.5, we can get the following result:

$$\mathfrak{h}(\omega, \mathbf{q}) > 59.2048. \quad [4.71]$$

With this, we can conclude that the three-agent system with chain topology of information exchange is robustly absolutely stable.

#### 4.5.3. Balanced graph topology

A four-agent system with balanced topology of information exchange is given by:

$$\dot{\mathbf{x}}(t) = \begin{bmatrix} -2 & 1 & 0 & 1 \\ 1 & -3 & 1 & 1 \\ 0 & 1 & -1 & 0 \\ 1 & 1 & 0 & -2 \end{bmatrix} \mathbf{x}(t) + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \mathbf{u}(t). \quad [4.72]$$

The multiagent system in terms of the canonical form is given by:

$$\dot{\boldsymbol{\eta}}(t) = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & -12 & -19 & -8 \end{bmatrix} \boldsymbol{\eta}(t) + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \mathbf{u}(t). \quad [4.73]$$

We consider a [10%, 10%, 10%, 10%] uncertainty in the last row coefficients of the state matrix. As a result of this assumption, the following matrix is obtained:

$$\mathbf{A}(\mathbf{q}) = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -q_0 & -q_1 & -q_2 & -q_3 \end{bmatrix}, \quad [4.74]$$

where  $q_0 = [0, 0]$ ,  $q_1 = [10.8, 13.2]$ ,  $q_2 = [17.1, 20.9]$ , and  $q_3 = [7.2, 8.8]$ .

Then,

$$\mathbf{A}(\mathbf{q}^-) = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & -13.2 & -20.9 & -8.8 \end{bmatrix},$$

$$\mathbf{F}^T(r) = [r_1 \ r_2 \ r_3 \ r_4],$$

where  $r_0 = 0$ ,  $r_1 = [0, 2.4]$ ,  $r_2 = [0, 3.8]$ , and  $r_3 = [0, 1.6]$ .

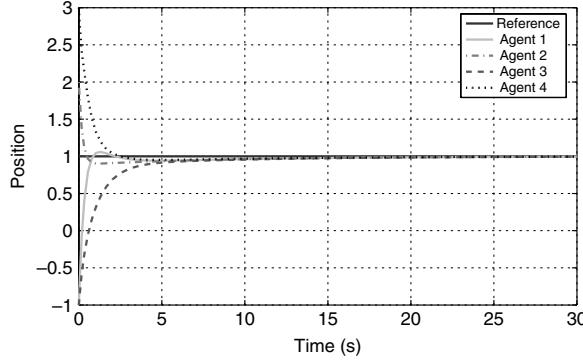
The  $\mathbf{F}$  matrix is given by:

$$\mathbf{F} = \begin{bmatrix} 0.00 & 0.00 & 0.00 & 0.00 \\ 0.00 & 5.76 & 9.12 & 3.84 \\ 0.00 & 9.12 & 14.44 & 6.08 \\ 0.00 & 3.84 & 6.08 & 2.56 \end{bmatrix}. \quad [4.75]$$

Thus, the optimal control law is given by:

$$\mathbf{K} = [0.3162 \quad 0.5221 \quad 0.2538 \quad 0.0489]. \quad [4.76]$$

After a simple inverse transformation of the optimal control law [4.76], the resulting optimal control law is applied to the multiagent system [4.72]. The performance of the robust optimal control is shown in Figure 4.9.



**Figure 4.9.** Balanced graph with 10% uncertainty

Then, the uncertain time-delay system of the form [4.34] has the following characteristic equation:

$$\begin{aligned} p(s, \mathbf{q}, \mathbf{r}, e^{-\tau s}) = & s^4 + [7.2, 8.8]s^3 + [17.1, 20.9]s^2 + [10.8, 13.2]s \\ & + (0.0489s^3 + 0.2538s^2 + 0.5221s + 0.3162)e^{-[0, 4.7]}. \end{aligned} \quad [4.77]$$

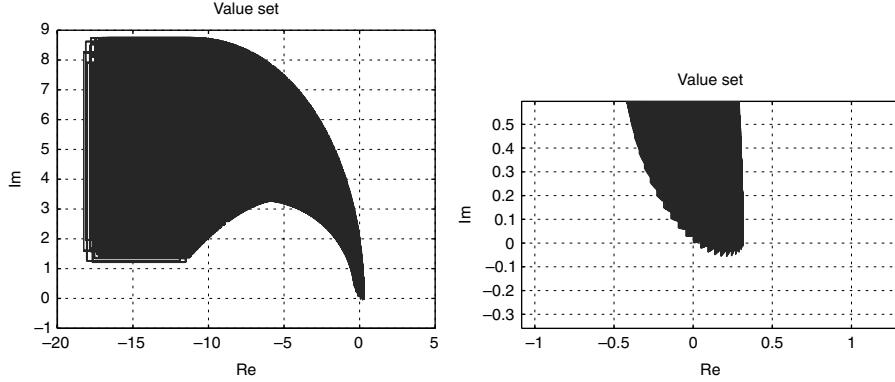
It is clear that the value set of  $\mathcal{P}_\tau$  is a set of complex numbers plotted on the complex plane for values of  $\mathbf{q}$ ,  $\mathbf{r}$ ,  $\omega$ , and  $\tau$  inside the defined boundaries. The results presented in [ROM 95] and [ROM 97] permit building the value set  $V_\tau(\omega)$  for the characteristic equation [4.77] and is presented in Figures 4.10(a) and (b).

It can be noted that the zero is not included in the value set  $V_\tau(\omega)$ . Then, the multiagent system with balanced information exchange topology [4.72] and  $\tau < 4.7$  is robustly stable.

Now, let us consider that the input is saturated and the saturation function satisfies the sector condition. We will consider the case of robust absolute stability of the multiagent system as follows. The transfer function of the above system is as follows:

$$G(s, \mathbf{q}) = \frac{s^3 + s^2 + s + 1}{s^4 + \bar{q}_1 s^3 + \bar{q}_2 s^2 + \bar{q}_3 s + 0.3162}, \quad [4.78]$$

where  $\bar{q}_1 = [7.2489, 8.8489]$ ,  $\bar{q}_2 = [17.3538, 21.1538]$ , and  $\bar{q}_3 = [11.3221, 13.7221]$ .



**Figure 4.10.** (a) value set of the uncertain time-delay system;  
(b) value set zoom

The fictitious transfer function has the following form:

$$z_f(s, \mathbf{q}) = \frac{s^4 + (\bar{q}_1 + 1)s^3 + (\bar{q}_2 + 1)s^2 + (\bar{q}_3 + 1)s + 0.3162}{s^4 + \bar{q}_1 s^3 + \bar{q}_2 s^2 + \bar{q}_3 s + 0.3162}, \quad [4.79]$$

where the positive and negative parts of  $\mathfrak{h}(\omega, \mathbf{q})$  and  $\mathfrak{g}(\omega, \mathbf{q})$  functions are the following:

$$\mathfrak{h}_p(\omega, \mathbf{q}) = \omega^4 + (2\bar{q}_1\bar{q}_3 + \bar{q}_2^2 + 0.62)\omega^2 + 0.0961, \quad [4.80]$$

$$\mathfrak{h}_n(\omega, \mathbf{q}) = (2\bar{q}_2 + \bar{q}_1^2)\omega^3 + (\bar{q}_3^2 + 0.62\bar{q}_2)\omega, \quad [4.81]$$

$$\mathfrak{g}_p(\omega, \mathbf{q}) = 2\omega^4 + 2(\bar{q}_1^2 + \bar{q}_1)\omega^3 + 2(\bar{q}_2^2 + \bar{q}_2)\omega^2 + 2(\bar{q}_3^2 + \bar{q}_3)\omega + 0.8122, \quad [4.82]$$

$$\mathfrak{g}_n(\omega, \mathbf{q}) = 2(\bar{q}_2 + 1)\omega^3 + 2(\bar{q}_1 + \bar{q}_3 + 2\bar{q}_1\bar{q}_3)\omega^2 + 2(1.62\bar{q}_2 + 0.31)\omega. \quad [4.83]$$

Using the equations defined in [4.25]–[4.26], we get

$$\mathfrak{h}_{\min}(\omega) = \omega^4 + 465.9199\omega^2 - 120.6106\omega^2 - 201.4114\omega + 0.091, \quad [4.84]$$

$$\mathfrak{g}_{\min}(\omega) = 2\omega^4 + 32.9757\omega^3 + 109.4124\omega^2 + 209.8658\omega + 0.8122. \quad [4.85]$$

On the one hand, it is clear that the  $\mathfrak{g}_{\min}(\omega)$  function is positive, because all of its coefficients are positive and the frequency  $\omega$  belongs to the interval  $(0, \infty)$ . On the other hand, from  $\mathfrak{h}_{\min}(\omega)$  function it is possible to get the interval  $[0.0005, 0.7985]$ . Then, the set  $\mathcal{V}$  is defined as:

$$\mathcal{V} = [0.0005, 0.7985] \times [7.2489, 8.8489] \times [17.3538, 21.1538] \times [11.3221, 13.7221]. \quad [4.86]$$

Now, applying Theorem 4.5, we can get the following result:

$$\mathfrak{h}(\omega, \mathbf{q}) > 53019536.0. \quad [4.87]$$

With this, we can conclude that the four-agent system with balanced topology of information exchange is robustly absolutely stable.

#### 4.6. Conclusions

We have presented a simple methodology for robust trajectory-tracking control design for multiagent systems with uncertainty and time delay. Parametric uncertainty was considered in the transformed system to simulate noise in the measured signals. We have presented sufficient conditions on maximum time delay for robust stability of the multiagent consensus and regulation system. Several cases such as cyclic topology, chain topology, as well as other balanced configurations have been considered. Some results to verify the robust absolute stability property for polynomic Lur'e systems were presented. These results were based on the sign decomposition concept which enables us to obtain a simpler test to verify the robust stability property of a nonlinear system than other previous results. It has been shown that the robust optimal control is enough to achieve multiagent consensus and trajectory tracking to a smooth time-varying reference.

#### 4.7. Bibliography

- [AND 90] ANDERSON B., MOORE J., *Optimal Control, Linear Quadratic Methods*, Dover Publications, 1990.
- [BAR 94] BARMISH B., *New Tools for Robustness of Linear Systems*, Macmillan, 1994.
- [ELI 99] ELIZONDO C., Estabilidad y controlabilidad robusta de sistemas lineales con incertidumbre multilineal, PhD Thesis, FIME-UANL, February 1999.
- [ELI 01] ELIZONDO C., “Robust positivity of the determinant via sign decomposition”, *World Multiconference on Systems, Cybernetics and Informatics*, pp. 355–360, 2001.
- [FU 89a] FU M., OLBROT A., POLIS M., “Introduction to the parametric approach to robust stability”, *IEEE Control Systems Magazine*, pp. 7–11, 1989.
- [FU 89b] FU M., OLBROT A., POLIS M., “Robust stability for time-delay systems: the edge theorem and graphical tests”, *IEEE Control Systems Magazine*, pp. 813–820, 1989.
- [GUE 05] GUERRERO J., Estabilidad absoluta robusta de sistemas lur'e con incertidumbre paramétrica polinómica, Master's Thesis, Universidad Autónoma de Tamaulipas, September 2005.
- [GUE 06] GUERRERO J., ROMERO G., MENDEZ A., DOMINGUEZ R., PANDURO M., PEREZ I., “Robust absolute stability using polynomial positivity and sign decomposition”, in *Lectures Notes in Control and Information Sciences*, vol. 341, pp. 423–430, Springer-Verlag, 2006.

- [HEL 98] HELTON J. W., MERINO O., *Classical Control Using  $H^\infty$  Methods*, SIAM, 1998.
- [IOA 96] IOANNOU P., SUN J., *Robust Adaptive Control*, Prentice Hall, 1996.
- [KHA 79] Kharitonov V., *Asymptotic Stability of an Equilibrium Point Position of a Family of Systems of Linear Differential Equations*, Plenum Publishing Corporation, pp. 1483–1485, 1979.
- [KRA 88] KRAVARIS C., PALANKI S., “A Lyapunov approach for robust nonlinear state feedback synthesis”, *IEEE Transactions in Automatic Control*, vol. 33, no. 12, pp. 1188–1191, 1988.
- [LEW 95] LEWIS F., SYRMOΣ V., *Optimal Control*, John Wiley and Sons, 1995.
- [LIN 07] LIN F., *Robust Control Design, An Optimal Control Approach*, John Wiley and Sons, 2007.
- [ROM 95] ROMERO G., COLLADO J., “Robust stability of interval plants with perturbed time delay”, *Proceedings of the IEEE American Control Conference*, pp. 326–327, 1995.
- [ROM 97] ROMERO G., Analysis of robust stability of dynamics systems with time delay, PhD Thesis, FIME-Universidad Autonoma de Nuevo Leon, June 1997.
- [STI 00] STIPANOVIĆ D., ŠILJAK D., “Robust strict positive realness via polynomial positivity”, *Proceedings of the IEEE American Control Conference*, pp. 4318–4325, 2000.
- [ZHO 96] ZHOU K., DOYLE J., GLOVER K., *Robust and Optimal Control*, Prentice Hall, 1996.

## Chapter 5

# On Adaptive and Robust Controlled Synchronization of Networked Robotic Systems on Strongly Connected Graphs

### 5.1. Summary

In this chapter, we study controlled synchronization of networked robotic systems communicating on strongly connected graphs. Adaptive and robust tracking control algorithms are used to synchronize heterogeneous robotic systems (with dynamic uncertainty) while following a desired trajectory. We demonstrate that synchronization is achievable when the robotic systems communicate over strongly connected graphs that are not necessarily balanced. The robustness of the control algorithms to constant delays in communication is also demonstrated. Numerical simulations using two-link robotic manipulators are presented to demonstrate the performance of the control algorithms.

### 5.2. Introduction

The idea of controlled synchronization [NIJ 03] was proposed to study collective behavior in coupled nonlinear dynamical systems. Different from self-synchronization in the nature [STR 03], controlled synchronization uses a control scheme with artificially induced interconnections to achieve synchronous action in interconnected dynamical systems. In networked robotic systems, the interconnections are achieved

---

Chapter written by Y.-C. LIU and N. CHOPRA.

by using a communication network that provides the agents with the ability to share information with their neighbors. By using the shared information, the agents can be controlled to achieve mutual synchronization. This phenomenon can potentially result in numerous advantages, for instance, synchronization of multiple robotic systems can improve the technology for mass and quick production in industrial manufacturing [KAN 92]. Furthermore, in a bilateral teleoperation system the idea of controlled synchronization was utilized for demonstrating delay-independent convergence of tracking errors between the master and the slave robots [CHO 08b].

In this chapter, the synchronization problem in networked robotic systems is studied where the individual agents are required to follow a desired trajectory. Synchronization can be used in the tracking problem for not only improving the tracking performance, but also for improving the transient behavior [CHO 08a, CHU 09b, ROD 04]. In the control of ship replenishment [KYR 07], synchronization was used for the leader following the main ship, which tracked a desired trajectory. Controlled synchronization has also been applied to altitude control for satellites and spacecraft [BON 05, KRI 09], and for tracking and formation control of multiple grounded and aerial vehicles [CHU 09a, FAX 04, LIU 07, SUN 09, YAD 06].

Controlled synchronization was proposed in [CHO 06] and [POG 98] to study collective behavior of nonlinear dynamical systems. Coordination of robotic manipulators was studied in [ROD 01] where two robots synchronize while tracking a desired trajectory. In [ROD 04], mutual synchronization of nonlinear robotic system while tracking a desired trajectory was proposed. With only position measurements, [ROD 04] demonstrated that the robotic systems can achieve synchronization by ensuring semi-global exponential convergence of the synchronization errors. However, the proposed control schemes did not scale well with the number of agents due to the requirement of all-to-all couplings. Furthermore, the system dynamics of the agents were required to be known, and no communication delay between the robots was considered [ROD 04].

A similar synchronization problem was studied in [CHO 08a] based on the passivity property. The synchronization results were developed under the assumption of balanced communication graphs and in the presence of constant time delays. These results were extended to the case of task space control in [LIU 09], where it was demonstrated that a group of nonidentical robots can achieve task synchronization in the presence of unknown constant delays. Task space synchronization in the presence of time-varying delays was studied in [LIU 12]. In addition, synchronization of networked robotic systems on strongly connected graphs was studied in [LIU 10b].

Contraction theory was recently exploited to guarantee synchronization and trajectory tracking for multiple robotic manipulators in [CHU 09b], where different time-scales for tracking and synchronization were presented. A neural network was used in [CHE 11] to synchronize networked Lagrangian systems with tracking control,

and in the absence of time delays. Two robotic manipulators were used to validate the proposed protocol, which consisted of a proportional and derivative (PD) controller and a nonlinear term with adaptive tuning laws at each robotic agent. A distributed controller was proposed in [MEH 10] to guarantee state synchronization of Euler–Lagrange systems with trajectory tracking under switching topologies. However, the communication links were assumed to be undirected, and time delays between agents were not considered. The synchronization problem with dynamic uncertainty was recently studied in [NUÑ 11] where nonlinear couplings were used together with a new adaptive control algorithm. The coordinated tracking problem with a dynamic leader was studied in [MEI 11].

In this chapter, we study the advantages and relative drawbacks of using adaptive tracking [CHO 08a] and robust tracking [LIU 10a] algorithms for synchronizing the agents. In the research of controlled synchronization, the assumption that the communication topology is balanced and strongly connected was required in [CHO 08a] and [LIU 12]. Even though the balanced graph assumption is weaker than all-to-all communication [ROD 01, ROD 04] and the undirected graph assumption [MEH 10], it can nevertheless result in high communication costs. Therefore, in this chapter, the controlled synchronization problem is studied with the assumption that the communication topology is only strongly connected. In addition to reducing the number of communication links, unreliability in the communication channel is another issue that should be carefully considered. Hence, we also study the robustness of the proposed controlled synchronization algorithms to communication delays.

The chapter is organized as follows. The problem formulation is described in section 5.3. The main results on controlled synchronization using adaptive tracking and robust tracking algorithms are presented in sections 5.4 and 5.5, respectively. Subsequently, numerical examples are discussed in section 5.6 to illustrate the performance of the controllers. Finally, the results are summarized in section 5.7.

### 5.3. Problem formulation

The agents (robotic systems) are modeled as Lagrangian systems, and no friction is considered in the agent dynamics. Following [SPO 06], the heterogeneous (and possibly unknown) dynamics for the individual agents are given as:

$$M_i(q_i)\ddot{q}_i + C_i(q_i, \dot{q}_i)\dot{q}_i + g_i(q_i) = u_i \quad [5.1]$$

where the index  $i = 1, \dots, N$  denotes the signals of the  $i$ th agent which is the vertex  $v_i$  (see section 5.8 for details on the graph theoretic terminology) in the communication topology. Additional details and properties of the considered robotic system are described in section 5.8.

It is assumed that there are  $N$  agents in the multiagent system, and  $\mathcal{N}_i$  denotes the set of neighbors of the  $i$ th agent. Furthermore, the interconnection graph is assumed to be strongly connected (see section 5.8 for details). The individual systems are required to synchronize with their neighbors and track a common time-varying trajectory  $q^d(t)$  that is assumed to be twice differentiable. Thus, the signals  $\dot{q}^d(t)$  and  $\ddot{q}^d(t)$  are well defined and are assumed to be bounded. The  $i$ th agent asymptotically tracks a desired trajectory if

$$\lim_{t \rightarrow \infty} \|q_i(t) - q^d(t)\| = \lim_{t \rightarrow \infty} \|\dot{q}_i(t) - \dot{q}^d(t)\| = 0 \quad [5.2]$$

where  $\|\cdot\|$  denotes the Euclidean norm of the enclosed signal.

We next define the signals  $a_i(t)$ ,  $v_i(t)$ , and  $s_i(t)$ , which are used to define the notion of synchronization studied in this chapter. Let

$$\begin{aligned} \tilde{q}_i(t) &= q_i(t) - q^d(t), & v_i(t) &= \dot{q}^d(t) - \Lambda \tilde{q}_i(t) \\ a_i(t) &= \ddot{q}^d(t) - \Lambda \dot{\tilde{q}}_i(t), & s_i(t) &= \dot{\tilde{q}}_i(t) + \Lambda \tilde{q}_i(t) \end{aligned} \quad [5.3]$$

where the matrix  $\Lambda \in R^{n \times n}$  is a positive definite diagonal matrix that is assumed to be the same for all agents. Consequently, the definition of synchronization for the networked robotic system follows.

**DEFINITION 5.1.**— Consider the robotic system [5.1], where  $s_i$  is the output of the  $i$ th agent. Then the networked robotic system is said to output synchronize if

$$\lim_{t \rightarrow \infty} \|s_j(t) - s_i(t)\| = 0 \quad \forall i, j \in \mathcal{N}_i \quad [5.4]$$

Owing to the interagent communication in the networked robotic system, it is possible to have delays in the closed-loop system. In this chapter, the time delays are assumed to be constant and bounded, and unknown to the controller. Denote  $T_{ji}$  as the unique transmission delay in the communication channel from the  $i$ th agent to the  $j$ th agent.

**DEFINITION 5.2.**— In the presence of communication delays, the networked robotic system [5.1] with  $s_i$  as output is said to delay-output synchronize if

$$\lim_{t \rightarrow \infty} \|s_j(t - T_{ji}) - s_i(t)\| = 0 \quad \forall i, j \in \mathcal{N}_i \quad [5.5]$$

In the sequel, for the sake of simplicity, we omit the arguments of signals unless specifically required for clarity. The uncertainty in the agent dynamics can be addressed using both adaptive and robust control methods, and we study synchronization based on both these methods.

## 5.4. Adaptive controlled synchronization on strongly connected graphs

### 5.4.1. Delay-free synchronization

The problem of synchronization is first studied using an adaptive tracking control algorithm for addressing the dynamic uncertainty in the agent dynamics. Let the control input be given as [CHO 08a]:

$$u_i = \hat{M}_i(q_i)a_i + \hat{C}_i(q_i, \dot{q}_i)v_i + \hat{g}_i(q_i) - K_i s_i + \tau_i \quad [5.6]$$

where  $\hat{M}_i$  denotes the estimate of the enclosed signal,  $K_i$  is a positive definite matrix, and  $\tau_i$  is the synchronizing control that will be defined subsequently.

Using Property 5.3 (see section 5.8) and substituting the above control input in [5.1], the closed-loop system can be rewritten as:

$$\begin{aligned} M_i(q_i)\dot{s}_i + C_i(q_i, \dot{q}_i)s_i + K_i s_i &= Y_i(q_i, \dot{q}_i, a_i, v_i)(\hat{\Theta}_i - \Theta_i) + \tau_i \\ &= Y_i(q_i, \dot{q}_i, a_i, v_i)\tilde{\Theta}_i + \tau_i \end{aligned} \quad [5.7]$$

where  $\tilde{\Theta}_i := \hat{\Theta}_i - \Theta_i$ ,  $i = 1, \dots, N$ . The signal  $s_i$  is the new output of the system, and the  $i$ th agent exchanges its output signal  $s_i$  with the other agents based on the communication graph.

Then, we let the vector of uncertain parameters be estimated by:

$$\dot{\hat{\Theta}}_i = -\Gamma_i^{-1}Y_i^T(q_i, \dot{q}_i, a_i, v_i)s_i \quad [5.8]$$

where  $\Gamma_i$  is a positive definite diagonal matrix.

The synchronizing control between the agents is given as:

$$\tau_i = \sum_{j \in \mathcal{N}_i} w_{ji}(s_j - s_i) \quad i = 1, \dots, N \quad [5.9]$$

where  $w_{ji}$  is a positive constant in the weighted Laplacian  $L_w(\mathcal{G}_w)$  (see section 5.8), and  $\mathcal{N}_i$  is the set of neighbors of the  $i$ th agent.

Define  $z_i = [s_i \ \tilde{q}_i \ \tilde{\Theta}_i]^T$  as the state of the individual agent, and denote by  $Z = [z_1^T \ \dots \ z_N^T]^T$  the state of the interconnected multiagent system. Our first result demonstrates adaptive synchronization of the networked robotic system on strongly connected graphs.

**THEOREM 5.1.–** Consider the dynamical system described by [5.3], [5.7], [5.8] with the synchronizing control [5.9]. If the interagent communication graph is strongly

*connected, then the agents output synchronize and asymptotically follow the desired trajectory.*

PROOF.– Consider a weighted positive definite storage function for the  $N$  agent system as:

$$V(Z) = \sum_{i=1}^N \gamma_i \left( s_i^T M_i s_i + \tilde{q}_i^T P_i \tilde{q}_i + \tilde{\Theta}_i^T \Gamma_i \tilde{\Theta}_i \right). \quad [5.10]$$

It is to be noted that the scalars  $\gamma_i$  are positive due to the assumption of a strongly connected graph and Lemma 5.1. Differentiating the storage function along the trajectory of the system and using Property 5.2, [5.7], and [5.8], the derivative reduces to:

$$\begin{aligned} \dot{V}(Z) &= \sum_{i=1}^N \gamma_i (2s_i^T M_i \dot{s}_i + s_i^T \dot{M}_i s_i + 2\tilde{q}_i^T P_i \dot{\tilde{q}}_i + 2\tilde{\Theta}_i^T \Gamma_i \dot{\tilde{\Theta}}_i) \\ &= \sum_{i=1}^N \gamma_i (-2s_i^T K_i s_i + 2s_i^T \tau_i + 2\tilde{q}_i^T P_i \dot{\tilde{q}}_i). \end{aligned}$$

Using the definition of  $s_i$  in [5.3] and choosing  $P_i = 2\Lambda K_i$  (see [SPO 90] for details), the derivative becomes:

$$\dot{V}(Z) = 2 \sum_{i=1}^N \gamma_i (-\dot{\tilde{q}}_i^T K_i \dot{\tilde{q}}_i - \tilde{q}_i^T \Lambda^T K_i \Lambda \tilde{q}_i + s_i^T \tau_i).$$

Substituting the synchronizing control [5.9] in the above equation yields:

$$\begin{aligned} \dot{V}(Z) &= -2 \sum_{i=1}^N \gamma_i (\dot{\tilde{q}}_i^T K_i \dot{\tilde{q}}_i + \tilde{q}_i^T \Lambda^T K_i \Lambda \tilde{q}_i) + 2 \sum_{i=1}^N \gamma_i s_i^T \sum_{j \in \mathcal{N}_i} w_{ji} (s_j - s_i) \\ &= -2 \sum_{i=1}^N \gamma_i \dot{\tilde{q}}_i^T K_i \dot{\tilde{q}}_i - 2 \sum_{i=1}^N \gamma_i \tilde{q}_i^T \Lambda^T K_i \Lambda \tilde{q}_i - \sum_{i=1}^N \sum_{j \in \mathcal{N}_i} \gamma_i w_{ji} (s_i^T s_i - s_j^T s_j) \\ &\quad - \sum_{i=1}^N \sum_{j \in \mathcal{N}_i} \gamma_i w_{ji} (s_j - s_i)^T (s_j - s_i). \end{aligned}$$

By denoting that  $S^T S = [s_1^T s_1 \ s_2^T s_2 \ \dots \ s_N^T s_N]^T$ , the above equation can be rewritten as:

$$\begin{aligned} \dot{V}(Z) &= -2 \sum_{i=1}^N \gamma_i \tilde{q}_i^T \Lambda^T K_i \Lambda \tilde{q}_i - 2 \sum_{i=1}^N \gamma_i \dot{\tilde{q}}_i^T K_i \dot{\tilde{q}}_i - \gamma^T L_w (S^T S) \\ &\quad - \sum_{i=1}^N \sum_{j \in \mathcal{N}_i} \gamma_i w_{ji} (s_j - s_i)^T (s_j - s_i). \end{aligned}$$

By applying the fact that  $\gamma^T L_w = 0$  from Lemma 5.1, the above equation can be written as:

$$\begin{aligned} \dot{V}(Z) &= -2 \sum_{i=1}^N \gamma_i \dot{\tilde{q}}_i^T K_i \dot{\tilde{q}}_i - 2 \sum_{i=1}^N \gamma_i \tilde{q}_i^T \Lambda^T K_i \Lambda \tilde{q}_i \\ &\quad - \sum_{i=1}^N \sum_{j \in \mathcal{N}_i} \gamma_i w_{ji} (s_j - s_i)^T (s_j - s_i) \leq 0. \end{aligned} \quad [5.11]$$

Since  $V(Z)$  is positive definite and  $\dot{V}(Z)$  is negative semi-definite, the zero solution of the system is globally stable and all signals are bounded. Integrating the above equation and letting  $t \rightarrow \infty$ , it can be seen that  $\dot{\tilde{q}}_i, \tilde{q}_i, (s_j - s_i) \in \mathcal{L}_2[0, \infty)$ , where  $j \in \mathcal{N}_i, \forall i$ . As all signals are bounded,  $\ddot{\tilde{q}}_i, \dot{\tilde{q}}_i, (\dot{s}_j - \dot{s}_i) \in \mathcal{L}_\infty$ . Hence, by Barbalat's Lemma [KHA 02],  $\lim_{t \rightarrow \infty} \dot{\tilde{q}}_i(t) = \lim_{t \rightarrow \infty} \tilde{q}_i(t) = 0$  and  $\lim_{t \rightarrow \infty} (s_j(t) - s_i(t)) = 0 \ j \in \mathcal{N}_i, \forall i$ . Therefore, the agents output synchronize and asymptotically follow the desired trajectory.

In addition, using the definition of  $s_i$  in [5.3], for any two agents  $i$  and  $j$ , output synchronization [5.4] implies that

$$\begin{aligned} s_j - s_i &= (\dot{\tilde{q}}_j + \Lambda \tilde{q}_j) - (\dot{\tilde{q}}_i + \Lambda \tilde{q}_i) \\ &= (\dot{q}_j + \Lambda q_j) - (\dot{q}_i + \Lambda q_i) = \dot{e}_{ij} + \Lambda e_{ij} \end{aligned} \quad [5.12]$$

where  $e_{ij} := q_j - q_i$  denotes the synchronization error between any two agents. Equation [5.12] represents an exponentially stable linear system with the input  $s_j - s_i$ . Hence, it follows [SON 03] that if  $s_j - s_i$  is a signal that converges asymptotically to zero and  $e_{ij}$  is bounded, then

$$\lim_{t \rightarrow \infty} \|e_{ij}(t)\| = 0 \quad \forall i, j \in \mathcal{N}_i. \quad [5.13]$$

Consequently, the agents' configuration and velocities in the joint space asymptotically approach each other.

#### 5.4.2. Synchronization with time delay

In this section, we extend the synchronization result with adaptive tracking scheme to the case where there are time delays in the communication network. As the  $i$ th agent receives the delayed output of its neighbors, the synchronizing control [5.9] is modified as:

$$\tau_i = \sum_{j \in \mathcal{N}_i} w_{ji} (s_j(t - T_{ji}) - s_i) \quad i = 1, \dots, N \quad [5.14]$$

where  $T_{ji}$  is the transmission delay from the  $j$ th agent to the  $i$ th agent. The signal  $s_j(t - T_{ji})$  in the synchronizing control [5.14] is the output signal of the  $j$ th agent that is transmitted  $T_{ji}$  unit of time ago. It is worth pointing out that knowledge of the communication delays is not required by the controller. Define  $Z_t = Z(\varphi)$ ,  $\varphi \in [t - T_M, t]$ , where  $T_M = \max(T_{ji})$ ,  $\forall i, j$  as the state for the system. Then the following result holds.

**THEOREM 5.2.–** Consider the dynamical system described by [5.3], [5.7], [5.8] with the synchronization control law [5.14]. If the communication graph is strongly connected, then all signals in the closed-loop system are bounded independent of the constant delays, the agents delay-output synchronize in the sense of [5.5] and asymptotically follow the desired trajectory.

**PROOF.–** Consider a weighted positive semi-definite storage functional for  $N$  agent system as:

$$V(Z_t) = \sum_{i=1}^N \gamma_i \left( s_i^T M_i s_i + \tilde{q}_i^T P_i \tilde{q}_i + \tilde{\Theta}_i^T \Gamma_i \tilde{\Theta}_i + \sum_{j \in \mathcal{N}_i} w_{ji} \int_{t-T_{ji}}^t s_j^T(\sigma) s_j(\sigma) d\sigma \right).$$

Following the proof of Theorem 5.1 and the control law [5.14], the derivative of this storage function is given as:

$$\begin{aligned} \dot{V}(Z_t) &= -2 \sum_{i=1}^N \gamma_i \tilde{q}_i^T K_i \dot{\tilde{q}}_i - 2 \sum_{i=1}^N \gamma_i \tilde{q}_i^T \Lambda^T K_i \Lambda \tilde{q}_i - \sum_{i=1}^N \sum_{j \in \mathcal{N}_i} \gamma_i w_{ji} (s_i^T s_i - s_j^T s_j) \\ &\quad - \sum_{i=1}^N \sum_{j \in \mathcal{N}_i} \gamma_i w_{ji} (s_i^T s_i - 2s_i^T s_j(t - T_{ji}) + s_j^T(t - T_{ji}) s_j(t - T_{ji})) \\ &= -2 \sum_{i=1}^N \gamma_i \tilde{q}_i^T K_i \dot{\tilde{q}}_i - 2 \sum_{i=1}^N \gamma_i \tilde{q}_i^T \Lambda^T K_i \Lambda \tilde{q}_i - \gamma^T L_w (S^T S) \\ &\quad - \sum_{i=1}^N \sum_{j \in \mathcal{N}_i} \gamma_i w_{ji} (s_j(t - T_{ji}) - s_i)^T (s_j(t - T_{ji}) - s_i) \\ &= -2 \sum_{i=1}^N \gamma_i \tilde{q}_i^T K_i \dot{\tilde{q}}_i - 2 \sum_{i=1}^N \gamma_i \tilde{q}_i^T \Lambda^T K_i \Lambda \tilde{q}_i \\ &\quad - \sum_{i=1}^N \sum_{j \in \mathcal{N}_i} \gamma_i w_{ji} (s_j(t - T_{ji}) - s_i)^T (s_j(t - T_{ji}) - s_i) \leq 0. \end{aligned}$$

From the above analysis and using the definition of  $s_i$  in [5.3], all signals in the dynamical system are bounded. Following the arguments as in Theorem 5.1,

it can be shown that  $\lim_{t \rightarrow \infty} (s_j(t - T_{ji}) - s_i(t)) = 0 \forall i, j \in \mathcal{N}_i$ . Therefore, as the communication graph is strongly connected, the agents delay-output synchronize [5.5]. Moreover, it is possible to demonstrate that [CHO 08a] delay-output synchronization further implies:

$$\lim_{t \rightarrow \infty} \|e_{ij}^d(t)\| = 0 \quad \forall i, j \in \mathcal{N}_i \quad [5.15]$$

where  $e_{ij}^d = (q_j(t - T_{ji}) - q_i) + (q^d - q^d(t - T_{ji}))$ .

## 5.5. Robust controlled synchronization on strongly connected graph

It is well known that unmodeled dynamics can lead to instability in certain adaptive control algorithms [ROH 85]. As discussed in [ORT 98] and [SPO 92], due to unmodeled dynamics, the adaptive tracking control algorithm may not necessarily lead to desirable tracking performance. Therefore, an alternative robust control approach [SPO 92] was proposed to ameliorate these effects. In this section, we study robust synchronization [LIU 10a] on strongly connected communication graphs.

### 5.5.1. Delay-free synchronization

It is assumed that for the unknown parameter vector  $\Theta_i$ , there exists a fixed nominal parameter vector  $\Theta_{0i} \in R^p$  and a constant  $\rho_i \in R^+$ , such that the initial estimation error is:

$$\|\tilde{\Theta}_i\| := \|\Theta_{0i} - \Theta_i\| \leq \rho_i. \quad [5.16]$$

Following [SPO 92], define a nominal control vector  $u_{0i}$  as:

$$u_{0i} = Y_i(q_i, \dot{q}_i, v_i, a_i)\Theta_{0i} - K_i s_i \quad [5.17]$$

where  $Y_i(q_i, \dot{q}_i, v_i, a_i)\Theta_{0i}$  is defined according to Property 5.3. Let the control input for agent  $i$  be given as:

$$\begin{aligned} u_i &= u_{0i} + Y_i(q_i, \dot{q}_i, v_i, a_i)\delta\Theta_i + \tau_i \\ &= Y_i(q_i, \dot{q}_i, v_i, a_i)(\Theta_{0i} + \delta\Theta_i) - K_i s_i + \tau_i \end{aligned} \quad [5.18]$$

where  $\delta\Theta_i$  is an additional control input defined as:

$$\delta\Theta_i = \begin{cases} -\rho_i \frac{Y_i^T s_i}{\|Y_i^T s_i\|}, & \text{if } \|Y_i^T s_i\| > \epsilon_i \\ -\frac{\rho_i}{\epsilon_i} Y_i^T s_i, & \text{if } \|Y_i^T s_i\| \leq \epsilon_i \end{cases} \quad [5.19]$$

with positive control parameter  $\epsilon_i$ . Let the estimate of parameter vector  $\hat{\Theta}_i := \Theta_{0i} + \delta\Theta_i$ , thus the estimation error is:

$$\hat{\Theta}_i - \Theta_i = \Theta_{0i} + \delta\Theta_i - \Theta_i = \tilde{\Theta}_i + \delta\Theta_i.$$

By substituting the control [5.18] in the dynamics [5.1], the closed-loop system can be written as:

$$M_i(q_i)\dot{s}_i + C_i(q_i, \dot{q}_i)s_i + K_i s_i = Y_i(q_i, \dot{q}_i, v_i, a_i)(\tilde{\Theta}_i + \delta\Theta_i) + \tau_i \quad [5.20]$$

where  $s_i$  is the new output of the closed-loop system. For the delay-free networked robotic system, the agents are coupled using the synchronizing control [5.9]. We define  $E_i = \{e_{ij} \mid j \in \mathcal{N}_i\}$  as the synchronization state of the  $i$ th agent, then let  $z_i = [s_i \ \tilde{q}_i \ E_i]^T$  denote the state of the  $i$ th agent and  $\bar{z}_i = [\dot{q}_i \ \tilde{q}_i \ E_i]^T$ . The agent state  $z_i$  is related to  $\bar{z}_i$  by a linear diffeomorphism,  $z_i = J_i \bar{z}_i$ . The matrix  $J_i$  is given by:

$$J_i = \begin{bmatrix} I_n & \Lambda & \emptyset_{n \times n_i} \\ \emptyset_{n \times n} & I_n & \emptyset_{n \times n_i} \\ \emptyset_{n_i \times n} & \emptyset_{n_i \times n} & I_{n_i} \end{bmatrix} \quad [5.21]$$

where  $\emptyset_{n \times n_i}$  denotes an  $n \times n_i$  zero matrix,  $I_n$  denotes an  $n \times n$  identical matrix, and  $n_i$  denotes the number of neighbors to the  $i$ th agent. Since  $J_i$  is a non-singular positive definite matrix with all non-distinct eigenvalues of one, we have

$$\|z_i\| = \|J_i \bar{z}_i\| \leq \|\bar{z}_i\|$$

where  $\|z_i\| := \sqrt{s_i^T s_i + \tilde{q}_i^T \tilde{q}_i + \sum_{j \in \mathcal{N}_i} e_{ij}^T e_{ij}}$ . By defining  $Z = [z_1^T \dots z_N^T]^T$  the state of the networked system, and  $\|Z\| := \sqrt{\sum_{i=1}^N \|z_i\|^2}$ , the next result follows for the robust tracking scheme.

**THEOREM 5.3.–** *Given nominal parameter vector  $\Theta_{0i}$ , positive control parameters  $\epsilon_i$ , and  $\rho_i \in R^+$  which satisfy [5.16], consider the dynamical system described by [5.19], [5.20], and synchronization control law [5.9]. If the interagent communication graph is strongly connected, then the synchronization errors and all solutions of the coupled dynamical system are uniformly ultimately bounded.*

**PROOF.–** Consider a positive definite Lyapunov function for the  $N$  agent system as:

$$V(Z) = \sum_{i=1}^N \gamma_i (s_i^T M_i s_i + \tilde{q}_i^T P_i \tilde{q}_i + \sum_{j \in \mathcal{N}_i} w_{ji} e_{ij}^T \Lambda e_{ij}). \quad [5.22]$$

Substituting the synchronizing control law [5.9] and choosing  $P_i = 2\Lambda K_i$ , the derivative of the storage function along trajectories becomes:

$$\begin{aligned} \dot{V}(Z) = 2 \sum_{i=1}^N \gamma_i & \left( -\dot{\tilde{q}}_i^T K_i \dot{\tilde{q}}_i - \tilde{q}_i^T \Lambda^T K_i \Lambda \tilde{q}_i + s_i^T Y_i(\tilde{\Theta}_i + \delta\Theta_i) \right. \\ & \left. + s_i^T \tau_i + \sum_{j \in \mathcal{N}_i} w_{ji} e_{ij}^T \Lambda \dot{e}_{ij} \right). \end{aligned} \quad [5.23]$$

Following the proof in Theorem 5.1 with Lemma 5.1, and substituting the definition of  $s_i$  in the above equation, the derivative of the storage function can be written as:

$$\begin{aligned} \dot{V}(Z) &= 2 \sum_{i=1}^N \gamma_i \left( -\tilde{q}_i^T K_i \dot{\tilde{q}}_i - \tilde{q}_i^T \Lambda^T K_i \Lambda \tilde{q}_i + s_i^T Y_i (\tilde{\Theta}_i + \delta \Theta_i) \right) \\ &\quad - \sum_{i=1}^N \sum_{j \in N_i} \gamma_i w_{ji} \dot{e}_{ij}^T \dot{e}_{ij} - \sum_{i=1}^N \sum_{j \in \mathcal{N}_i} \gamma_i w_{ji} e_{ij}^T \Lambda^T \Lambda e_{ij} \\ &\leq 2 \sum_{i=1}^N \gamma_i \left( -\tilde{q}_i^T K_i \dot{\tilde{q}}_i - \tilde{q}_i^T \Lambda^T K_i \Lambda \tilde{q}_i + s_i^T Y_i (\tilde{\Theta}_i + \delta \Theta_i) \right) \\ &\quad - \sum_{i=1}^N \sum_{j \in \mathcal{N}_i} \gamma_i w_{ji} e_{ij}^T \Lambda^T \Lambda e_{ij}. \end{aligned} \quad [5.24]$$

From the robust tracking control scheme [SPO 92], the term  $s_i^T Y_i (\tilde{\Theta}_i + \delta \Theta_i)$  is upper bounded by  $\epsilon_i \rho_i / 4$ . Hence, the last equation leads to the following inequality:

$$\dot{V}(Z) \leq - \sum_{i=1}^N \gamma_i \bar{z}_i^T Q_i \bar{z}_i + \sum_{i=1}^N \gamma_i \epsilon_i \rho_i / 2 \quad [5.25]$$

where  $Q_i \in R^{(2+n_i)n \times (2+n_i)n}$  is defined as:

$$Q_i = \begin{bmatrix} 2K_i & \emptyset_{n \times n} & \emptyset_{n \times nn_i} \\ \emptyset_{n \times n} & 2\Lambda^T K_i \Lambda & \emptyset_{n \times nn_i} \\ \emptyset_{nn_i \times n} & \emptyset_{nn_i \times n} & K_w \otimes \Lambda^T \Lambda \end{bmatrix} \quad [5.26]$$

where  $K_w$  is diagonal matrix whose diagonal term is  $w_{ji}$  corresponding to  $E_i$ ,  $\otimes$  is Kronecker product, and  $n_i$  denotes the number of neighbors connecting to the  $i$ th agent. Denoting  $\Xi := \sum_{i=1}^N \gamma_i \epsilon_i \rho_i / 2$  and  $\alpha_i := \lambda_{\min}(Q_i)$ , the minimum eigenvalue of  $Q_i$ , the inequality [5.25] becomes:

$$\dot{V}(Z) \leq - \sum_{i=1}^N \gamma_i \alpha_i \|\bar{z}_i\|^2 + \Xi. \quad [5.27]$$

By using the fact that  $\|z_i\| \leq \|\bar{z}_i\|$ , and defining  $\gamma_m := \min(\gamma_i)$  and  $\alpha_m := \min(\alpha_i)$  for  $i = 1, \dots, N$ , the derivative function can be rewritten as:

$$\begin{aligned} \dot{V}(Z) &\leq - \sum_{i=1}^N \gamma_i \alpha_i \|z_i\|^2 + \Xi \leq -\gamma_m \alpha_m \|Z\|^2 + \Xi \\ &\leq -(1-\eta) \gamma_m \alpha_m \|Z\|^2 - \eta \gamma_m \alpha_m \|Z\|^2 + \Xi \\ &\leq -(1-\eta) \gamma_m \alpha_m \|Z\|^2 := -W(Z), \quad \forall \|Z\| \geq \beta_s \end{aligned}$$

where the constant  $\eta \in (0, 1)$ ,  $\beta_s := \sqrt{\Xi/\eta\gamma_m\alpha_m}$ , and  $W(Z)$  is a continuous positive definite function.

Noting Property 5.1 and [5.22], there exist  $\mathcal{K}_\infty$  functions  $\alpha_a$  and  $\alpha_b$  such that

$$\alpha_a(\|Z\|) \leq V(Z) \leq \alpha_b(\|Z\|).$$

Using Theorem 4.18 in [KHA 02], the synchronization system is uniformly ultimately bounded, and there exist  $T > 0$ , such that  $\forall t \geq T$ ,  $\|Z\| \leq \alpha_a^{-1}(\alpha_b(\beta_s))$ .

### 5.5.2. Synchronization with time delay

In the presence of delays, define  $E_i^d = \{e_{ij}^d | j \in \mathcal{N}_i\}$  where  $e_{ij}^d = (q_j(t - T_{ji}) - q_i) + (q^d - q^d(t - T_{ji}))$  as the delay-synchronization state of the  $i$ th agent. By denoting  $z_{ti} = [s_i \ \tilde{q}_i \ E_i^d]^T$ ,  $\bar{z}_{ti} = [\tilde{q}_i \ \tilde{q}_i \ E_i^d]^T$ , and following the definition in section 5.5.1, we have  $\|z_{ti}\| \leq \|\bar{z}_{ti}\|$ . Moreover, based on the definition of  $e_{ij}^d$ , we obtain that  $\dot{e}_{ij}^d = (\dot{q}_j(t - T_{ji}) - \dot{q}_i(t)) + (\dot{q}^d(t) - \dot{q}^d(t - T_{ji}))$ . It is possible to demonstrate that the trajectories are ultimately bounded in this case as well, and this is outlined in the next result.

**THEOREM 5.4.**— Given nominal parameter vector  $\Theta_{0i}$ , positive control parameters  $\epsilon_i$ , and  $\rho_i \in R^+$  which satisfy [5.16], consider the dynamical system described by [5.19], [5.20], and synchronization control law [5.14]. If the interagent communication graph is strongly connected, then the synchronization and tracking errors for the interconnected system are bounded.

**PROOF.**— Consider a positive semi-definite storage functional for this system as:

$$\begin{aligned} V(Z_t) = & \sum_{i=1}^N \gamma_i \left( s_i^T M_i s_i + \tilde{q}_i^T P_i \tilde{q}_i + \sum_{j \in \mathcal{N}_i} w_{ji} e_{ij}^{dT} \Lambda e_{ij}^d \right. \\ & \left. + \sum_{j \in \mathcal{N}_i} w_{ji} \int_{t-T_{ji}}^t s_j^T(\sigma) s_j(\sigma) d\sigma \right). \end{aligned}$$

Differentiating along trajectories of the system and choosing  $P_i = 2\Lambda K_i$  yields:

$$\begin{aligned} \dot{V}(Z_t) = & 2 \sum_{i=1}^N \gamma_i \left( -\dot{\tilde{q}}_i^T K_i \dot{\tilde{q}}_i - \tilde{q}_i^T \Lambda^T K_i \Lambda \tilde{q}_i + s_i^T Y_i (\tilde{\Theta}_i + \delta \Theta_i) + s_i^T \tau_i \right. \\ & \left. + \sum_{j \in \mathcal{N}_i} w_{ji} e_{ij}^{dT} \Lambda \dot{e}_{ij}^d \right) + \sum_{i=1}^N \gamma_i \sum_{j \in \mathcal{N}_i} w_{ji} (s_j^T s_j - s_j^T (t - T_{ji}) s_j (t - T_{ji})). \end{aligned}$$

Substituting the synchronizing control [5.14] into the equation above, we have:

$$\begin{aligned}\dot{V}(Z_t) = & 2 \sum_{i=1}^N \gamma_i \left( -\ddot{\tilde{q}}_i^T K_i \dot{\tilde{q}}_i - \tilde{q}_i^T \Lambda^T K_i \Lambda \tilde{q}_i + s_i^T Y_i (\tilde{\Theta}_i + \delta \Theta_i) \right) \\ & + 2 \sum_{i=1}^N \gamma_i \sum_{j \in \mathcal{N}_i} w_{ji} e_{ij}^{dT} \Lambda e_{ij}^d + 2 \sum_{i=1}^N \gamma_i \sum_{j \in \mathcal{N}_i} w_{ji} (s_j^T s_j - s_i^T s_i) \\ & - \sum_{i=1}^N \gamma_i \sum_{j \in \mathcal{N}_i} w_{ji} (s_j(t - T_{ji}) - s_i)^T (s_j(t - T_{ji}) - s_i).\end{aligned}\quad [5.28]$$

Using Lemma 5.1, the derivative of the storage function can be written as:

$$\begin{aligned}\dot{V}(Z_t) = & 2 \sum_{i=1}^N \gamma_i \left( -\ddot{\tilde{q}}_i^T K_i \dot{\tilde{q}}_i - \tilde{q}_i^T \Lambda^T K_i \Lambda \tilde{q}_i + s_i^T Y_i (\tilde{\Theta}_i + \delta \Theta_i) \right) \\ & - \sum_{i=1}^N \gamma_i \sum_{j \in \mathcal{N}_i} w_{ji} \dot{e}_{ij}^{dT} \dot{e}_{ij}^d - \sum_{i=1}^N \gamma_i \sum_{j \in \mathcal{N}_i} w_{ji} e_{ij}^{dT} \Lambda^T \Lambda e_{ij}^d.\end{aligned}\quad [5.29]$$

Following the proof in Theorem 5.3, the derivative of the storage function becomes:

$$\begin{aligned}\dot{V}(Z_t) \leq & - \sum_{i=1}^N \gamma_i \alpha_i \|\bar{z}_i\|^2 + \Xi < -\gamma_m \alpha_m \|Z_t\|^2 + \Xi \\ \leq & -(1-\eta) \gamma_m \alpha_m \|Z_t\|^2 - \eta \gamma_m \alpha_m \|Z_t\|^2 + \Xi \\ \leq & -(1-\eta) \gamma_m \alpha_m \|Z_t\|^2 := -W(Z_t), \quad \forall \|Z_t\| \geq \beta_s\end{aligned}$$

where the constant  $\eta \in (0, 1)$ ,  $\beta_s := \sqrt{\Xi / \eta \gamma_m \alpha_m}$ , and  $W(Z_t)$  is a continuous positive definite function. Hence, it is evident from the above equation that  $\exists \beta_s$  such that for  $\|Z_t\| \geq \beta_s$ ,  $\dot{V}(Z_t) \leq 0$ . As  $V(Z_t)$  is positive definite, this implies that the state vector is ultimately bounded.

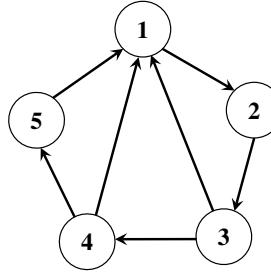
## 5.6. Numerical examples

Numerical simulations are presented in this section to demonstrate the efficiency of the proposed algorithms. In the simulations, five agents, modeled as nonlinear

2DOF planar robots [SPO 06], are interconnected according to the strongly connected topology shown in Figure 5.1. The weighted Laplacian is given as:

$$L_w = \begin{bmatrix} 60 & 0 & -25 & -20 & -15 \\ -45 & 45 & 0 & 0 & 0 \\ 0 & -55 & 55 & 0 & 0 \\ 0 & 0 & -50 & 50 & 0 \\ 0 & 0 & 0 & -40 & 40 \end{bmatrix}.$$

Based on  $L_w$ , we have the vector  $\gamma^T = [1.000, 1.333, 1.090, 0.700, 0.375]$ , which satisfies the property that  $\gamma^T L_w = 0$ .



**Figure 5.1.** Robots communicate over a strongly connected communication graph

Using the linear parametrization property of Lagrangian system, the constant vector of inertia parameters  $\Theta$  and the matrix of known functions  $Y$  can be given from [SPO 06]. In the following simulations, the actual constant vector of five planar robots are selected as:

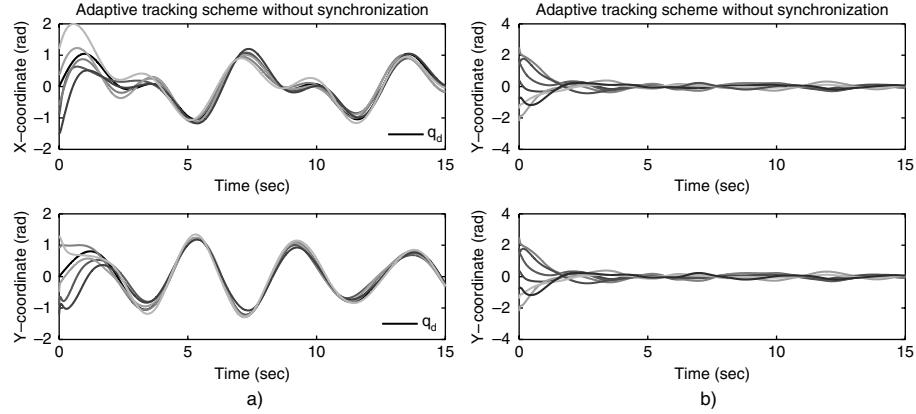
$$\begin{aligned} \Theta_1 &= [2.406, 0.792, 0.416, 7.980, 0.660] \\ \Theta_2 &= [0.688, 0.096, 0.172, 0.840, 0.160] \\ \Theta_3 &= [2.108, 0.520, 0.408, 1.900, 0.520] \\ \Theta_4 &= [1.520, 0.500, 0.260, 1.500, 0.500] \\ \Theta_5 &= [2.612, 0.786, 0.353, 2.015, 0.605] \end{aligned} \quad [5.30]$$

In addition, the tracking gains are given as  $K_i = 3I_3$ ,  $i = 1, \dots, 5$ , and  $\Lambda = 2I_3$ .

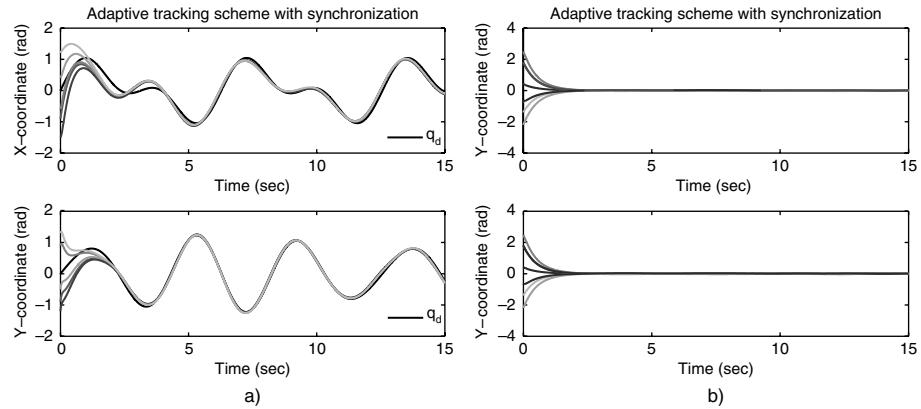
### 5.6.1. Adaptive tracking algorithm

We first demonstrate controlled synchronization with the use of the adaptive tracking scheme. Given the adaptive control parameters  $\Gamma_i = 20I_5$ , and the initial unknown parameters  $\Theta_1(0) = [2.647, 0.871, 0.458, 2.178, 0.726]$ ,  $\Theta_2(0) = [0.585, 0.082, 0.146, 0.714, 0.136]$ ,  $\Theta_3(0) = [2.003, 0.494, 0.388, 1.805, 0.494]$ ,  $\Theta_4(0) = [1.368, 0.450, 0.234, 1.350, 0.450]$ , and  $\Theta_5(0) = [1.596, 0.525, 0.273, 1.575, 0.525]$ ,

the simulation results in the absence of communication delays are shown in Figure 5.2. Owing to the dynamic uncertainty in the robotic agents, the tracking errors are relatively high, which leads to larger synchronization errors as shown in Figure 5.2(b).



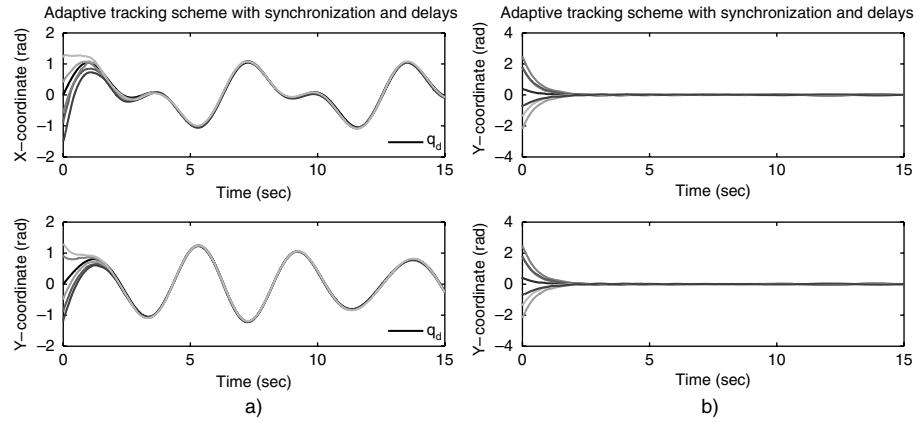
**Figure 5.2.** Performance of the adaptive tracking algorithm in the absence of controlled synchronization: a) Configurations of the robotic agents in the networked system and b) Synchronization errors, which are the position errors between an agent with its neighbors



**Figure 5.3.** Both tracking and synchronization performance improves with the use of controlled synchronization: a) Configurations of the robotic agents in the networked system and b) Synchronization errors between agents in the network

If controlled synchronization is used, and the agents are interconnected using strongly connected graphs (Figure 5.1) with synchronization gains  $L_w$ , the tracking performance improves and the synchronization errors between the agents converge to the origin asymptotically as shown in Figure 5.3. In the presence of communication

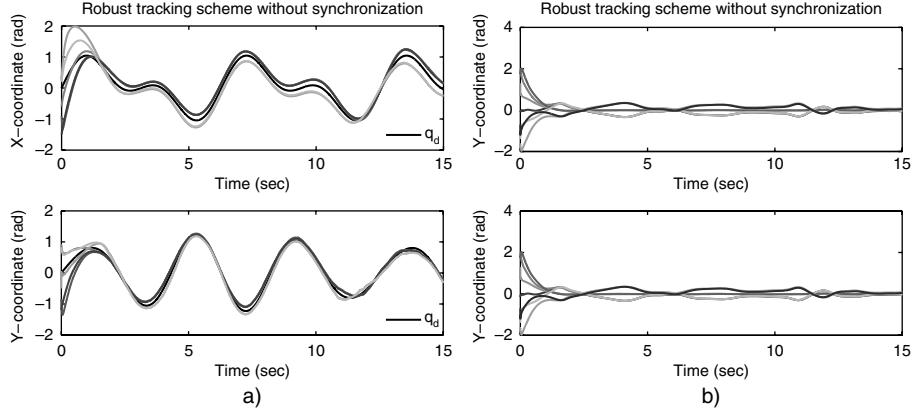
delays  $T_{31} = 0.1$  sec,  $T_{41} = 0.8$  sec,  $T_{51} = 0.7$  sec,  $T_{12} = 0.2$  sec,  $T_{23} = 0.6$  sec,  $T_{34} = 0.3$  sec, and  $T_{45} = 0.25$  sec, the simulation result is shown in Figure 5.4. We can observe that even with time delays in the communication channels, the interconnected robotic system is stable, and the synchronization errors approach the origin.



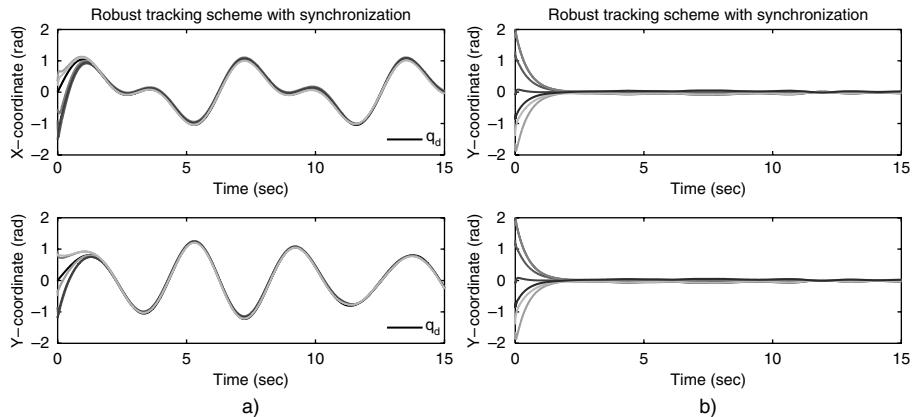
**Figure 5.4.** The interconnected system is stable and achieves synchronization even if the communication channels are subjected to delays: a) Configurations of the robotic agents in the networked system and b) Synchronization errors between agents in the network

### 5.6.2. Robust tracking algorithm

We next illustrate the performance of the networked robotic system using the robust tracking scheme. The control parameters are selected as  $\epsilon_i = 10$ , and  $\rho_1 = 1.6423$ ,  $\rho_2 = 1.0846$ ,  $\rho_3 = 1.1006$ ,  $\rho_4 = 1.1049$ ,  $\rho_5 = 2.2698$ . The fixed nominal parameter vectors are given as  $\Theta_{01} = [2.387, 1.750, 0.399, 1.576, 0.092]$ ,  $\Theta_{02} = [0.844, -0.752, 0.186, 1.220, 0.78]$ ,  $\Theta_{03} = [2.186, -0.384, 0.426, 2.32, 1.116]$ ,  $\Theta_{04} = [1.324, 1.400, 0.212, 1.000, -0.100]$ , and  $\Theta_{05} = [2.590, -0.171, 0.382, 2.412, 1.184]$ . In the case without synchronization, the simulation result is shown in Figure 5.5, where we can see that the tracking errors are ultimately bounded. The synchronization errors are shown in Figure 5.5 (b). Using controlled synchronization with the gain matrix  $L_w$ , the simulation results are shown in Figure 5.6. As expected, the interconnected robotic system is stable and the tracking performance improves as compared to the case where no controlled synchronization is used between the agents. Figure 5.7 shows the simulation results under communication delays, where the delays are selected to be the same as in the adaptive tracking scheme. We can see that even under delays, the entire system is stable, and the synchronization errors are ultimately bounded.



**Figure 5.5.** Robust tracking scheme in the absence of controlled synchronization: a) Configurations of the robotic agents in the networked system and b) Synchronization errors between agents in the network



**Figure 5.6.** Controlled synchronization improves the tracking and synchronization performance: a) configurations of the robotic agents in the networked system and b) synchronization errors between agents in the network

### 5.6.3. Disturbances

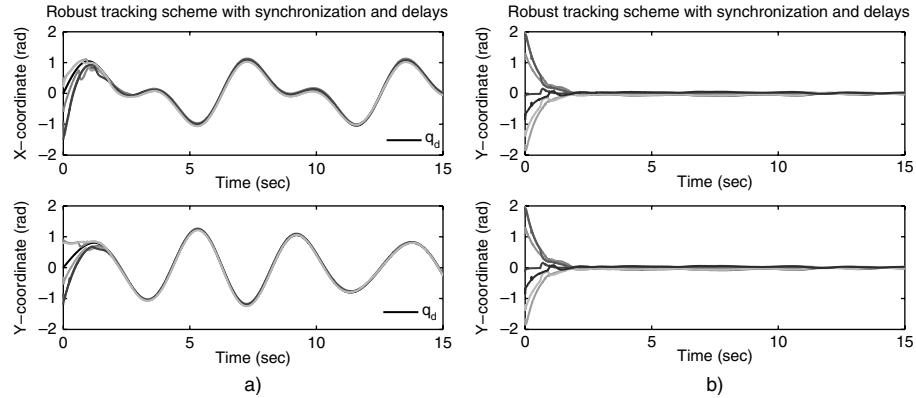
In this subsection, simulation results were conducted to illustrate the influence of disturbances on the synchronization performance. Control and simulation parameters were selected as in sections 5.6.1 and 5.6.2. We assume that there exists noise in the adaptive control algorithm,

$$\dot{\hat{\Theta}}_i = -\Gamma_i^{-1}Y_i^T s_i + d_a, \quad [5.31]$$

and in the synchronization control

$$\tau_i = \sum_{j \in \mathcal{N}_i} w_{ji}(s_j(t - T_{ji}) - s_i) + d_s \quad [5.32]$$

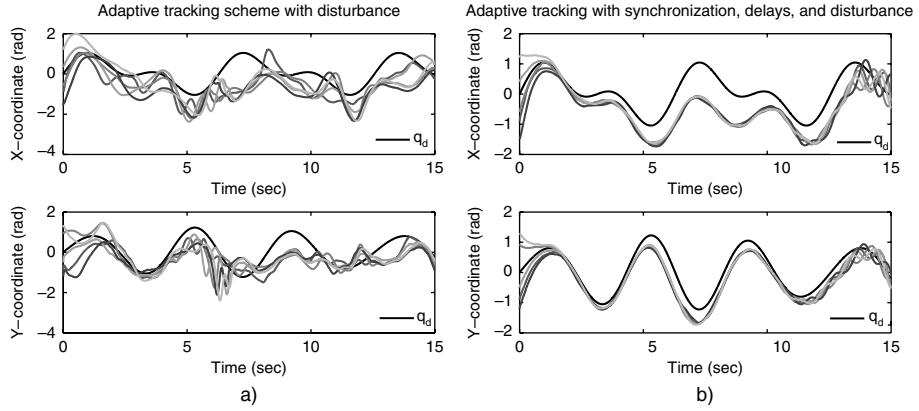
where  $d_a$  and  $d_s$  are assumed to be Gaussian white noise with high variance. The noise  $d_a$  could potentially emanate from the sensors, and  $d_s$  may be caused due to noise in the communication channels.



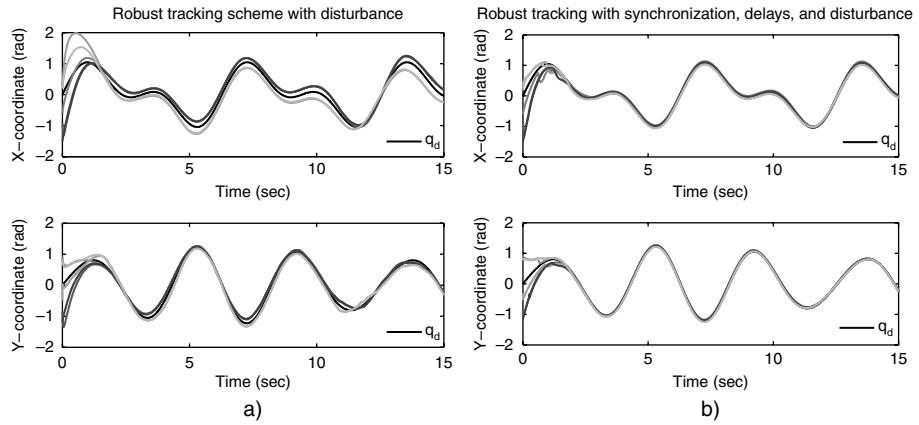
**Figure 5.7.** In the presence of delays, the networked robotic system is stable, and the synchronization errors are bounded: a) configurations of the robotic agents in the networked system and b) synchronization errors between agents in the network

For the adaptive control algorithm, the simulation results with and without synchronization are shown in Figure 5.8. The noise in the parameter update law [5.31] increases the tracking error as seen in Figure 5.8 (a). With the use of synchronization control [5.32], and with time delays and disturbances, the agents in the communication network achieve synchronous action initially, however eventually the presence of noise renders the networked system unstable as illustrated in Figure 5.8 (b).

For the robust tracking scheme, the dynamics considered are described by [5.19], [5.20] with noisy synchronization control [5.32]. In the simulation results (Figure 5.9), we can observe that even with disturbances and delays in the communication channels, the robust controlled synchronization ensures that the tracking and synchronization errors are bounded. The simulation results in this subsection illustrate the fact that, in the synchronization problem, the adaptive tracking control algorithm may not necessarily lead to desirable tracking and synchronizing performance in the presence of disturbances.



**Figure 5.8.** Controlled synchronization with adaptive tracking scheme in the presence of disturbances: a) configurations of the robotic agents with disturbance and b) configurations of the robotic agents with synchronization, delays, and disturbance



**Figure 5.9.** Controlled synchronization with robust tracking scheme in the presence of disturbances: a) configurations of the robotic agents with disturbance and b) configurations of the robotic agents with synchronization, delays, and disturbance

## 5.7. Conclusions

In this chapter, controlled synchronization of networked robotic systems was studied using both adaptive and robust tracking schemes. Using a weighted storage function, it was demonstrated that synchronization of networked robotic systems is achievable on communication graphs that are strongly connected and not necessarily

balanced. In addition to the delay-free case, the effect of communication delays on the synchronization behavior was also studied. While the adaptive tracking algorithm was able to provide asymptotic synchronization, the robust tracking scheme resulted in bounded tracking and synchronization errors. However, simulations indicated that the adaptive controller was sensitive to external disturbances and noise in the parameter update dynamics, resulting in poor and potentially unstable closed-loop behavior. The study of controlled synchronization in this chapter can be used for flight and vehicle formation control, and will be explored in the future.

## 5.8. Appendix

### 5.8.1. Robotic system

The robotic systems are modeled as Lagrangian systems, and the dynamics (ignoring friction) of the  $i$ th agent evolve as:

$$M_i(q_i)\ddot{q}_i + C_i(q_i, \dot{q}_i)\dot{q}_i + g_i(q_i) = u_i \quad [5.33]$$

where  $q_i \in R^n$  is the vector of generalized configuration coordinates,  $u_i \in R^n$  is the vector of generalized forces acting on the system,  $M_i(q_i) \in R^{n \times n}$  is a symmetric, positive definite matrix,  $C_i(q_i, \dot{q})\dot{q}_i \in R^n$  is the vector of Coriolis/centrifugal forces, and  $g_i(q_i) = \partial H_i / \partial q_i \in R^n$  is the gradient of the potential function  $H_i(q_i)$ . The above equations of motion exhibit certain fundamental properties.

**PROPERTY 5.1.–** The matrix  $M(q)$  is symmetric positive definite and there exist positive constants  $\lambda_m$  and  $\lambda_M$ , such that

$$\lambda_m I_n \leq M(q) \leq \lambda_M I_n \quad [5.34]$$

where  $I_n$  denotes an  $n \times n$  identical matrix.

**PROPERTY 5.2.–** Under an appropriate definition of the matrix  $C$ , the matrix  $\dot{M} - 2C$  is skew symmetric.

**PROPERTY 5.3.–** For any differentiable vector  $\xi \in R^n$ , the Lagrangian dynamics are linearly parameterizable, which gives

$$M(q)\dot{\xi} + C(q, \dot{q})\xi + g(q) = Y(q, \dot{q}, \xi, \dot{\xi})\Theta \quad [5.35]$$

where  $\Theta$  is a constant  $p$ -dimensional vector of inertia parameters, and  $Y(q, \dot{q}, \xi, \dot{\xi}) \in R^{n \times p}$  is the matrix of known functions of the generalized coordinates and their higher derivatives.

### 5.8.2. Graph theory

Communication topology and information exchange between agents can be represented as a graph [GOD 01].

**DEFINITION 5.3.–** By a graph  $\mathcal{G}$  we mean a finite set  $\mathcal{V}(\mathcal{G}) = \{v_i, \dots, v_N\}$ , whose elements are called nodes or vertices, together with set  $\mathcal{E}(\mathcal{G}) \subset \mathcal{V} \times \mathcal{V}$ , whose elements are called edges, which is an ordered pair of distinct vertices. An edge  $(v_i, v_j)$  is said to be incoming with respect to  $v_j$  and outgoing with respect to  $v_i$  and can be represented as an arrow with vertex  $v_i$  as its tail and vertex  $v_j$  as its head.

The in-degree of a vertex  $v \in \mathcal{G}$  is the number of edges that have this vertex as a head. Similarly, the out-degree of a vertex  $v \in \mathcal{G}$  is the number of edges that have this vertex as the tail. If the in-degree equals the out-degree for all vertices  $v \in \mathcal{V}(\mathcal{G})$ , then the graph is said to be balanced.

If, for all  $(v_i, v_j) \in \mathcal{E}(\mathcal{G})$ , the edge  $(v_j, v_i) \in \mathcal{E}(\mathcal{G})$  then the graph is said to be undirected. Otherwise, it is called a directed graph. A path of length  $r$  in a directed graph is a sequence  $v_0, \dots, v_r$  of  $r + 1$  distinct vertices such that for every  $i \in \{0, \dots, r - 1\}$ ,  $(v_i, v_{i+1})$  is an edge. A directed graph is **strongly connected** if any two vertices can be joined by a path.

From the above terminology, the information exchange between the networked robots can be represented by a weighted directed graph  $\mathcal{G}_w = (\mathcal{V}, \mathcal{E}, \mathcal{W})$ , where  $\mathcal{W}(\mathcal{G}_w) = \{w_{ji}\}$ ,  $j \in \mathcal{N}_i$  and  $w_{ji}$  is the weight of the edge from  $v_j$  to  $v_i$ , and  $\mathcal{N}_i$  denotes the subset of vertices, which are joined by an edge with vertex  $v_i$  as an arrow. The weighted Laplacian  $L_w(\mathcal{G}_w)$  for the interconnection graph is defined as:

$$L_w := [L_{wij}] = \begin{cases} = \sum_{j \in \mathcal{N}_i} w_{ji} & \text{if } i = j \\ = -w_{ji} & \text{if } j \in \mathcal{N}_i \\ = 0 & \text{Otherwise} \end{cases} \quad [5.36]$$

**LEMMA 5.1.–** [IGA 09, LIU 10b] If the communication graph is strongly connected and weights are positive, then there exists a vector  $\gamma$  (with positive elements) satisfying  $\gamma^T L_w = 0$ , where the vector  $\gamma$  is defined as:

$$\gamma = [\gamma_1, \dots, \gamma_N]^T, \quad \gamma_i > 0 \quad \forall i \in \{1, \dots, N\} \quad [5.37]$$

for the case with  $N$  vertices.

### 5.9. Bibliography

- [BON 05] BONDHUS A.K., PETERSEN K.Y., GRAVDAHL J.T., “Leader/follower synchronization of satellite attitude without angular velocity measurements”, *IEEE Conference on Decision and Control*, pp. 7270–7277, December 2005.

- [CHE 11] CHEN G., LEWIS F.L., “Distributed adaptive tracking control for synchronization of unknown networked Lagrangian systems”, *IEEE Transactions on Systems, Man, and Cybernetics-Part B: Cybernetics*, vol. 41, no. 3, pp. 805–816, June 2011.
- [CHO 06] CHOPRA N., SPONG M.W., “Passivity-based control of multi-agent systems”, in KAWAMURA S., SVININ M. (eds), *Advances in Robot Control: From Everyday Physics to Human-Like Movements*, Springer Verlag, New York, pp. 107–134, 2006.
- [CHO 08a] CHOPRA N., LIU Y.-C., “Controlled synchronization of mechanical systems”, *ASME Dynamic Systems and Control Conference*, pp. 1221–1228, October 2008.
- [CHO 08b] CHOPRA N., SPONG M.W., LOZANO R., “Synchronization of bilateral teleoperators with time delay”, *Automatica*, vol. 44, no. 8, pp. 2142–2148, August 2008.
- [CHU 09a] CHUNG S.-J., AHSUN U., SLOTINE J.-J.E., “Application of synchronization to formation flying spacecraft: Lagrangian approach”, *AIAA Journal of Guidance, Control and Dynamics*, vol. 32, no. 2, pp. 512–526, March–April 2009.
- [CHU 09b] CHUNG S.-J., SLOTINE J.-J.E., “Cooperative robot control and concurrent synchronization of Lagrangian systems”, *IEEE Transactions on Robotics*, vol. 25, no. 3, pp. 686–700, June 2009.
- [FAX 04] FAX J.A., MURRAY R.M., “Information flow and cooperative control of vehicle formations”, *IEEE Transactions on Automatic Control*, vol. 49, no. 4, pp. 1465–1476, September 2004.
- [GOD 01] GODSIL C., ROYLE G., *Algebraic Graph Theory*, Springer, New York, 2001.
- [IGA 09] IGARASHI Y., HATANAKA T., FUJITA M., SPONG M.W., “Passivity-based attitude synchronization in  $SE(3)$ ”, *IEEE Transactions on Control Systems Technology*, vol. 17, no. 5, pp. 1119–1134, September 2009.
- [KAN 92] KANITANI K., Industrial Robot Synchronous Control Method and Apparatus, US Patent, October 1992.
- [KHA 02] KHALIL H.K., *Nonlinear Systems*, Prentice Hall, New Jersey, 2002.
- [KRI 09] KRISTIANSEN R., LORÍA A., CHAILLET A., NICKLASSON P.J., “Spacecraft relative rotation tracking without angular velocity measurements”, *Automatica*, vol. 45, pp. 750–756, 2009.
- [KYR 07] KYRKJEBO E., PETTERSEN K.Y., WONDERGEM M., NIJMEIJER H., “Output synchronization control of ship replenishment operations: theory and experiments”, *Control Engineering Practice*, vol. 15, pp. 741–755, 2007.
- [LIU 07] LIU H.-T., SHAN J., SUN D., “Adaptive synchronization control of multiple spacecraft formation flying”, *Journal of Dynamic Systems, Measurement, and Control*, vol. 129, no. 3, pp. 337–342, May 2007.
- [LIU 09] LIU Y.-C., CHOPRA N., “Controlled synchronization of robotic manipulators in the task space”, *ASME Dynamic Systems and Control Conference*, pp. 443–450, October 2009.
- [LIU 10a] LIU Y.-C., CHOPRA N., “Robust controlled synchronization of interconnected robotic systems”, *American Control Conference*, pp. 1434–1439, June 2010.

- [LIU 10b] LIU Y.-C., CHOPRA N., “Synchronization of networked robotic systems on strongly connected graphs”, *IEEE Conference on Decision and Control*, pp. 3194–3199, 2010.
- [LIU 12] LIU Y.-C., CHOPRA N., “Controlled synchronization of heterogeneous robotic manipulators in the task space”, *IEEE Transactions on Robotics*, vol. 28, no. 1, 2012.
- [MEH 10] MEHRABIAN A.R., TAFAZOLI S., KHORASANI K., “Cooperative tracking control of Euler-Lagrange systems with switching communication network topologies”, *IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, pp. 756–761, July 2010.
- [MEI 11] MEI J., REN W., MA G., “Distributed coordinated tracking with a dynamic leader for multiple Euler-Lagrange systems”, *IEEE Transactions on Automatic Control*, vol. 56, no. 6, pp. 1415–1421, June 2011.
- [NIJ 03] NIJMEIJER H., RODRIGUEZ-ANGELES A., *Synchronization of Mechanical Systems*, World Scientific, River Edge, NJ, 2003.
- [NUÑ 11] NUÑO E., ORTEGA R., BASAÑEZ L., HILL D., “Synchronization of networks of nonidentical Euler-Lagrange systems with uncertain parameters and communication delays”, *IEEE Transactions on Automatic Control*, vol. 56, no. 4, pp. 935–941, April 2011.
- [ORT 98] ORTEGA R., LORIA A., NICKLASSON P.J., SIRA-RAMIREZ H., *Passivity-Based Control of Euler-Lagrange Systems: Mechanical, Electrical and Electromechanical Applications*, Communications and Control Engineering Series, Springer Verlag, London, 1998.
- [POG 98] POGROMSKY A.Y., “Passivity based design of synchronizing systems”, *International Journal of Bifurcation and Chaos*, vol. 8, pp. 295–319, 1998.
- [ROD 01] RODRIGUEZ-ANGELES A., NIJMEIJER H., “Coordination of two robot manipulators based on position measurements only”, *International Journal of Control*, vol. 74, no. 13, pp. 1311–1323, 2001.
- [ROD 04] RODRIGUEZ-ANGELES A., NIJMEIJER H., “Mutual synchronization of robots via estimated state feedback: a cooperative approach”, *IEEE Transactions on Control Systems Technology*, vol. 12, no. 4, pp. 542–554, July 2004.
- [ROH 85] ROHRS C., VALAVANI L., ATHANS M., STEIN G., “Robustness of continuous-time adaptive control algorithms in the presence of unmodeled dynamics”, *IEEE Transactions on Automatic Control*, vol. 30, no. 9, pp. 881–889, September 1985.
- [SON 03] SONTAG E.D., “A remark on the converging-input converging-state property”, *IEEE Transactions on Automatic Control*, vol. 48, no. 2, pp. 313–314, February 2003.
- [SPO 92] SPONG M.W., “On the robust control of robot manipulators”, *IEEE Transactions on Automatic Control*, vol. 37, no. 11, pp. 1782–1786, November 1992.
- [SPO 06] SPONG M.W., HUTCHINSON S., VIDYASAGAR M., *Robot Modeling and Control*. John Wiley & Sons, Inc., New York, 2006.
- [SPO 90] SPONG M.W., ORTEGA R., KELLY R., “Comments on ‘adaptive manipulator control: a case study’ by J. Slotine and W. Li”, *IEEE Transactions on Automatic Control*, vol. 35, no. 6, pp. 761–762, June 1990.

- [STR 03] STROGATZ S.H., *Sync: The Emerging Science of Spontaneous Order*, Hyperion, New York, 2003.
- [SUN 09] SUN D., WANG C., SHANG W., FENG G., “A synchronization approach to trajectory tracking of multiple mobile robots while maintaining time-varying formations”, *IEEE Transactions on Robotics*, vol. 25, no. 5, pp. 1074–1086, October 2009.
- [YAD 06] YADLAPALLI S.K., DARBHA S., RAJAGOPAL K.R., “Information flow and its relation to stability of the motion of vehicles in a rigid formation”, *IEEE Transactions on Automatic Control*, vol. 51, no. 8, pp. 1315–1319, August 2006.

# Chapter 6

## Modeling and Control of Mini UAV

### 6.1. Introduction

The applications of mini unmanned aerial vehicles (UAVs) comprise both military and civilian, though the latter has had a lower development rate. The use of aerial robots, especially miniature (mini and micro) UAVs (MAVs), has enhanced activities such as surveillance of sensitive areas (borders, harbors, prisons), wildlife study, or natural disaster assessment. Flight formation between several MAVs improves the tasks of traffic surveillance, pollution monitoring, or path-following on the highway, through their collaborative work.

The complete dynamics of these kind of vehicles, taking into account aeroelastic effects, flexibility of the wings, internal dynamics of the engine, and the whole set of changing variables are quite complex and somewhat unmanageable for the purposes of control. Therefore, it is interesting to consider a simplified model of an aircraft formed by a minimum number of states and inputs, but retaining the main features that must be considered when designing control laws for a real aircraft.

There are missions whose scope is beyond the capabilities of conventional small UAVs designs, since they require not only longer flight endurance but also hovering/VTOL (vertical takeoff and landing) capabilities. Besides the commonly used aerial vehicles, the hybrid or convertible MAVs have been gaining popularity recently. By matching the takeoff and landing capabilities of the helicopter with the forward flight efficiencies of fixed-wing aircraft, the convertible UAV promises a unique blend of capabilities at a lower cost than other UAV configurations. Two kinds

---

Chapter written by G. FLORES COLUNGA, J.A. GUERRERO, J. ESCAREÑO and R. LOZANO.

of Convertible UAV vehicles are discussed: the tailsitter and the quad-plane, in sections 6.3 and 6.4, respectively.

Tailsitter vehicles represent a configuration of an aircraft that remains relatively unexplored. Tailsitters have more operational flexibility than conventional UAVs because a vertical airframe attitude is adopted during takeoff and landing, while maintaining a horizontal airframe attitude during cruise just like conventional airplanes. Tailsitters have not been as widely adopted as an aircraft configuration due to complex flight dynamics in the hover mode, making them typically very difficult to control. The Convair XF-Y1 and Lockheed XF-V1 were examples of experimental Tailsitter aircraft in the 1950s, but they were unsuccessful due mostly to the problem caused by the awkward positioning of the pilot required during the vertical flight phases, which would not be relevant for UAVs. In the 1990s Boeing presented its tailsitter Heliwing UAV with a flight controller using cyclic-pitch rotor control for its vertical flight phases [CAS 05], while more recently in [STO 02] the University of Sydney's T-Wing UAV has an autopilot that uses control surfaces in the slipstream of fixed-pitch propellers for control in its vertical flight phases.

While the tilt-rotor concept is very promising, it also comes with significant challenges. Indeed, it is necessary to design controllers that will work over the complete flight envelope of the vehicle: from low-speed vertical flight through high-speed forward flight. The main change in this respect (besides understanding the detailed aerodynamics) is the large variation in the vehicle dynamics between these two different flight regimes. Several experimental platforms have been realized with a body structure in which the transition flight is executed by turning the complete body of the aircraft [GRE 06, ESC 06, STO 04, STO 02, ESC 07]. In [STO 04] and [STO 02] the authors described the development (modeling, control architecture, and experimental prototype) of a two-rotor tailsitter. The control architecture features a complex switching logic of classical linear controllers to deal with the vertical, transition, and forward flight. Green and Oh [GRE 06] present a classical airplane configuration MAV to perform both operational modes. The hover flight is autonomously controlled by an onboard control flight system while the transition and cruise flight is manually controlled. A standard proportional and derivative (PD) controller is employed during hover flight to command the rudder and elevator. In [ESC 06] some preliminary results are presented for the vertical flight of a two-rotor MAV as well as a low-cost embedded flight control system. There are some examples of other tilt-rotor vehicles with quadrotor configurations, such as Boeing's V44 [SNY 00] and the QTW UAV [NON 07]. In [ONE 08] the authors present the progress of their ongoing project, an aircraft with four tilting wings.

This chapter deals with the modeling and control of different configurations of the UAVs, and is organized as follows. Section 6.2 presents the general model of the aircraft represented as a rigid body. Sections 6.3 and 6.4 deal with the Hybrid or Convertible MAVs, combining the advantages of horizontal and vertical flight.

Different approaches for nonlinear control are presented using the Lyapunov theory. Finally, some concluding remarks are presented in section 6.5.

## 6.2. General model

To obtain the UAV dynamical model, it is assumed that it flies over a local area on the Earth. Then, the flat-Earth model equations will be used [STE 92]. The equations representing the position, the kinematics, the forces, and the moments are explained in this section.

### 6.2.1. Translational motion

The Newton's equation applied to translational motion of the aircraft takes the form:

$$\mathbf{F}_{A,T} + \mathbf{C}_{b/n}m\mathbf{g}^n = \frac{d}{dt}[m\mathbf{v}_{CM/i}]. \quad [6.1]$$

The derivative of the right hand side of equation [6.1] must be taken with respect to an inertial reference frame. The term  $\mathbf{C}_{b/n}m\mathbf{g}^n$  refers to the body-frame gravitational attraction, where  $m$  is the vehicle mass and  $\mathbf{C}_{b/n}$  is the rotation matrix relating the inertial frame to the body frame. Also, the  $\mathbf{v}_{CM/i}$  is the velocity vector of the aircraft center of gravity measured with respect to inertial frame. Performing the differentiation in [6.1], results in:

$$\mathbf{F}_{A,T} + \mathbf{C}_{b/n}m\mathbf{g}^n = m\mathbf{v}_{CM/e}^b + \boldsymbol{\omega}_{b/i}^b \times m\mathbf{v}_{CM/e}^b. \quad [6.2]$$

Using the fact that  $\boldsymbol{\omega}_{b/i}^b \times \mathbf{v}_{CM/e}^b = \boldsymbol{\Omega}_{b/e}^b \mathbf{v}_{CM/e}^b$ , where  $\boldsymbol{\Omega}_{b/e}^b$  is a skew symmetric matrix given by:

$$\boldsymbol{\omega}_{b/i}^b \times \mathbf{v}_{CM/e}^b = \begin{bmatrix} 0 & -R & Q \\ R & 0 & -P \\ -Q & P & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \boldsymbol{\Omega}_{b/e}^b \mathbf{v}_{CM/e}^b. \quad [6.3]$$

Performing the equation [6.1] for  $\mathbf{v}_B$ , it results in:

$${}^b\dot{\mathbf{v}}_{CM/e}^b = \boldsymbol{\Omega}_{b/e}^b \mathbf{v}_{CM/e}^b + \mathbf{C}_{b/n}m\mathbf{g}^n + \frac{\mathbf{F}_{A,T}^b}{m}. \quad [6.4]$$

Using the rotation matrix

$$\mathbf{C}_{b/n} = \begin{bmatrix} c_\theta c_\psi & c_\theta s_\psi & -s_\theta \\ c_\psi s_\theta s_\phi - c_\phi s_\psi & c_\phi c_\psi + s_\theta s_\phi s_\psi & c_\theta s_\phi \\ s_\phi s_\psi + c_\phi c_\psi s_\theta & c_\phi s_\theta s_\psi - c_\psi s_\phi & c_\theta c_\phi \end{bmatrix}, \quad [6.5]$$

where  $s$  and  $c$  denote  $\sin(\cdot)$  and  $\cos(\cdot)$ , respectively, the velocity expressed in the NED (north, east, and down) frame is calculated as:

$${}^e\dot{\mathbf{p}}^n = \mathbf{C}_{n/b}\mathbf{v}_{CM/e}^b. \quad [6.6]$$

Newton's second law is used to obtain the equations of translational motion in the inertial frame of reference as:

$$\ddot{\mathbf{p}}^n = \mathbf{C}_{n/b}\frac{\mathbf{F}_{A,T}^b}{m} + \mathbf{g}^n. \quad [6.7]$$

### 6.2.2. Angular motion

By applying the Newton's second law to the rate of change of angular momentum of the vehicle, it can obtain the angular accelerations, then

$$\mathbf{M}_{A/T}^b = \frac{d}{dt}[\mathbf{h}_b], \quad [6.8]$$

$$= \dot{\mathbf{h}}_b + \boldsymbol{\omega}_{b/i}^b \times \mathbf{h}_b, \quad [6.9]$$

where  $\mathbf{M}_{A/T}^b = [l, m, n]^T$  is the torque vector,  $\mathbf{h}_b$  is the total angular velocity, and  $\boldsymbol{\omega}_{b/i}^b = [P, Q, R]^T$  the angular velocity, all of them written in body coordinates. Here  $\mathbf{h}_b$  is given by:

$$\mathbf{h}_b = \begin{bmatrix} J_{xx} & -J_{xy} & -J_{xz} \\ -J_{xy} & J_{yy} & -J_{yz} \\ -J_{xz} & -J_{yz} & J_{zz} \end{bmatrix} \begin{bmatrix} P \\ Q \\ R \end{bmatrix} = \mathbf{J}^b \boldsymbol{\omega}_{b/i}^b, \quad [6.10]$$

with  $\mathbf{J}^b$  as the inertia matrix of the rigid body. Therefore, substituting the time derivative of [6.10] in [6.8], the angular acceleration of the aircraft can be represented as:

$${}^b\dot{\boldsymbol{\omega}}_{b/e}^b = (\mathbf{J}^b)^{-1}[\mathbf{M}_{A/T}^b - \boldsymbol{\Omega}_{b/e}^b \mathbf{J}^b \boldsymbol{\omega}_{b/e}^b]. \quad [6.11]$$

If the configuration of the vehicle presents symmetry with respect to the  $x - z$  and  $x - y$  planes, the cross-products of inertia vanish. Then  $\mathbf{J}^b$  can be written as:

$$\mathbf{J}^b = \begin{bmatrix} J_{xx} & 0 & 0 \\ 0 & J_{yy} & 0 \\ 0 & 0 & J_{zz} \end{bmatrix}. \quad [6.12]$$

Then using the last inertia matrix and the torque vector  $\mathbf{M}_{A/T}^b$ , the equation [6.11] yields:

$$\dot{P} = \frac{(J_{yy} - J_{zz})QR}{J_{xx}} + \frac{l}{J_{xx}}, \quad [6.13]$$

$$\dot{Q} = \frac{(J_{zz} - J_{xx})RP}{J_{yy}} + \frac{m}{J_{yy}}, \quad [6.14]$$

$$\dot{R} = \frac{(J_{xx} - J_{yy})PQ}{J_{zz}} + \frac{n}{J_{zz}}. \quad [6.15]$$

### 6.2.3. Angular rate

Like the Euler-angle vector, the components of the Euler-rate vector are not measured along the orthogonal axes. The components of this vector are measured at every rotation (Euler angles). The inertial angular rate vector relative to body-axis frame is related to the Euler angular rate vector by:

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 1 & \tan \theta \sin \phi & \tan \theta \cos \phi \\ 0 & \cos \phi & -\sin \phi \\ 0 & \frac{\sin \phi}{\cos \theta} & \frac{\cos \phi}{\cos \theta} \end{bmatrix} \begin{bmatrix} P \\ Q \\ R \end{bmatrix}. \quad [6.16]$$

Thus, the angular rates  $P$ ,  $Q$ , and  $R$  become the elements of the absolute angular velocity vector  $\omega_{b/i}^b$  of the body frame. The equation [6.16] will be written symbolically as:

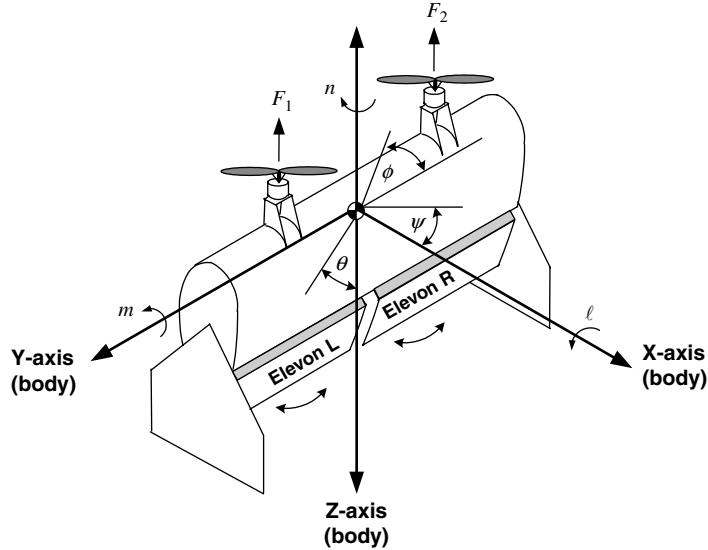
$$\dot{\Phi} = H(\Phi)\omega_{b/i}^b. \quad [6.17]$$

If we differentiate equation [6.17], we get

$$\ddot{\Phi} = \dot{H}(\Phi)\omega_{b/e}^b + H(\Phi)\dot{\omega}_{b/e}^b. \quad [6.18]$$

## 6.3. Control of a mini tailsitter

The vertical flight schematic of this VTOL vehicle is shown in Figure 6.1, which is basically a tailless fixed-wing aircraft with two propellers. In hover mode, the altitude is controlled with the collective thrust, this means the lift force is generated through increasing the lift force produced by the propellers. The pitch attitude angular displacement is achieved by moving the elevons in the same direction. The vertical yaw-attitude angular displacement is achieved through moving the elevons in opposing direction. The vertical roll-attitude angular displacement is controlled by changing the pitch angle of the variable pitch propeller (VPP). Typically, MAVs like the planar vertical takeoff and landing (PVTOL) platforms [CAS 05] modify the speed of the DC electric motors to effect altitude and attitude control. But when brushless electric AC motors are used, control responses have been too slow due to the time delay produced by the available speed controllers, leading to problems using motor speed for roll control. VPP is thus being investigated as a potential solution, increasing the control response. This allows us to implement a simple flight controller without considering the time delay in actuators.



**Figure 6.1.** Vehicle schematic for vertical flight mode of Bidule [WON 07]

### 6.3.1. Linear control strategy

In this section, following [WON 07], attitude stabilization of the VTOL in hover flight is presented. Kinematic and moment equations will be used to obtain three decoupled attitude systems for the pitch, roll, and yaw angular position. These subsystems will be obtained using only the kinematics and moment equations from the general model. The vehicle main wing has a NACA 0008 profile. Several aerodynamic factors will be taken into account to obtain the transfer function that represents the dynamic of each system.

#### 6.3.1.1. Roll subsystem

To obtain the roll control system, it is assumed that the pitch and yaw rates are zero. Then, the vehicle can be analyzed in a similar manner to a PVTOL flight platform, as in [CAS 05]. This configuration is shown in Figure 6.2. Therefore, using the equation of  $\dot{\phi}$  in [6.16] and equation [6.13], the rotational dynamics for the roll angle can be represented by:

$$\ddot{\phi} = l/J_x, \quad [6.19]$$

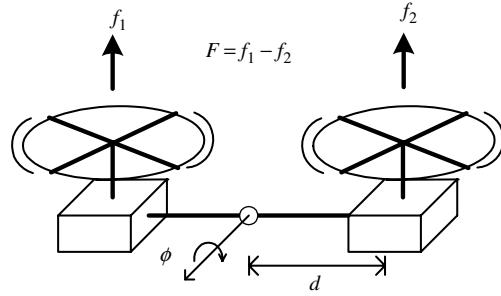
where the moment  $l$  can be calculated as:

$$l = F \cdot d - C_{\ell_\phi} \dot{\phi}. \quad [6.20]$$

where  $F = f_1 - f_2$  is the force produced by the thrust difference between the right and the left rotors,  $d$  is the distance from the center of mass to each rotor, and  $C_{\ell_\phi} = 0.36$  is the roll damping derivative. Then, equation [6.19] can be rewritten as:

$$\ddot{\phi} = (C_{t_\varphi} \varphi \rho n^2 D_p^4 \cdot d - C_{\ell_\phi} \dot{\phi}) / J_{xx}, \quad [6.21]$$

where  $C_{t_\varphi} = 0.0025$  is a derivative which represents the thrust slope with respect to the VPP angle  $\varphi$ ,  $\rho$  is the density of the air,  $n$  is the number of revolutions per second of the motor, and  $D_p$  is the diameter of the propellers. The derivatives have been estimated using a shareware program called JavaProp [HEP 06].



**Figure 6.2.** Approach of PVTOL to control the roll position

Then, using the values given in Table 6.1, the transfer function for the roll angle with respect to the VPP angle is given by:

$$\frac{\phi(s)}{\varphi(s)} = \frac{5}{s^2 + 25s}. \quad [6.22]$$

It is clear that in order to control the roll subsystem, a VPP dynamic model is needed. As shown in Figure 6.3, the aerodynamic pitch moment of the blades must be equal to the moment generated by the servo mechanism. The blade profile has been identified as NACA 0014, then the blade pitch moment can be approximated as:

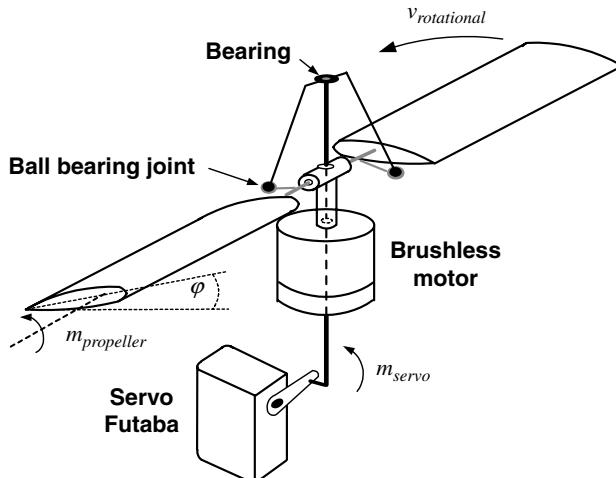
$$m_b = \frac{\rho V_{t_b}^2 S_b \bar{c}_b}{2 J_{y_b}} [C_{m_\varphi} \varphi + C_{m_\dot{\varphi}} \dot{\varphi}] = k_s f_s \delta_s, \quad [6.23]$$

where the subscript  $b$  denotes the blade. The term  $V_{t_b}$  denotes the total velocity of the propeller at the tip which is given by:

$$V_{t_b} = \sqrt{v_{axial}^2 + v_{radial}^2}, \quad [6.24]$$

where  $v_{radial} = \pi n D$ .

Parameter	Value	Definition
$S$	0.18 m <sup>2</sup>	Wing reference area
$\bar{c}$	0.3 m	Wing chord
$b$	0.6 m	Wing span
$\rho$	1.225 kg/m <sup>3</sup>	Air density
$V$	10 m/s	Wind velocity (airflow)
$J_x$	0.0144 kg.m	$x$ -axis moment of inertia
$J_y$	0.0254 kg.m	$y$ -axis moment of inertia
$J_z$	0.0312 kg.m	$z$ -axis moment of inertia
$d$	0.2 m	Rotor distance from the center of mass
$D$	0.27	Propeller diameter
$n$	9,000 RPM	Rotor speed
$S_b$	0.006 m <sup>2</sup>	Blade reference area
$\bar{c}_b$	0.3 m	Blade chord
$J_{y_b}$	$2 \times 10^{-6}$	$y$ -axis blade inertia
$f_s$	17 N	Force produced by the servo

**Table 6.1.** Bidule-CSyRex aircraft parameters**Figure 6.3.** Schematic of variable pitch propeller (VPP) system [WON 07]

The term  $C_{m_\varphi} = -0.0019$  is the estimated blade pitch moment coefficient slope with respect to  $\varphi$ .  $C_{m_\varphi} = 1.6 \times 10^{-5}$  is a stability derivative generated by the variation of the VPP rate. The expression  $k_s f_s \delta_s$  represents the moment produced by the servo, where  $f_s$  is the force produced by the servo,  $\delta_s$  is the servo displacement, and  $k_s$  is a mechanical reduction factor. Using the parameter values in Table 6.1, the VPP

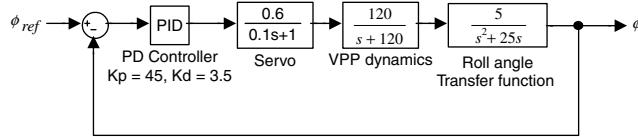
dynamic's transfer function is given by:

$$\frac{\varphi(s)}{\delta_s(s)} = \frac{120}{s + 120}. \quad [6.25]$$

The actuator dynamics given in [KAN 01] are as follows:

$$\frac{\delta_s(s)}{\delta_c(s)} = \frac{0.6}{0.1s + 1}. \quad [6.26]$$

Then using the transfer functions given previously, the control closed-loop system shown in Figure 6.4 is proposed to stabilize the roll angle. Since the characteristic equation  $0.1s^4 + 15.5s^3 + 445s^2 + 4,260s + 16,200$  has all its roots in  $-6.7 \pm 4.5i$ ,  $-121$ , and  $-20.5$ , then the system is stable.



**Figure 6.4. Roll control loop**

### 6.3.1.2. Pitch subsystem

To obtain the pitch control system the vehicle is considered to be a tailless aircraft flying in forward flight. Assuming that the roll angle is small enough and the roll rate is instantaneously zero, then using the equation of  $\dot{\theta}$  in [6.16] and equation [6.14], a second-order differential equation describing the rotational dynamics for the platform pitch angle can be written as:

$$\ddot{\theta} = \frac{\rho V^2 S \bar{c}}{2J_{yy}} \left[ C_{m_{ac}} + C_{m_\alpha} \alpha + C_{m_{\delta_e}} \delta_e + C_{m_q} \frac{\bar{c}}{V} \dot{\theta} \right], \quad [6.27]$$

where  $\bar{c}$  is the wing chord,  $S$  is the wing reference area,  $V$  is the airflow speed,  $C_m$  is the pitching moment coefficient (see [ETK 96]). It is assumed that in steady hover flight  $\theta = \alpha$  and  $C_{m_{ac}} = 0$ . Therefore, [6.27] can be reduced to:

$$\ddot{\theta} = \frac{\rho V^2 S \bar{c}}{2J_{yy}} \left[ C_{m_\alpha} \theta + C_{m_{\delta_e}} \delta_e + C_{m_q} \frac{\bar{c}}{V} \dot{\theta} \right]. \quad [6.28]$$

The derivative  $C_{m_\alpha} = -0.145$  represents the variation of the pitching moment with respect to the angle-of-attack  $\alpha$ . The derivative  $C_{m_{\delta_e}} = \partial C_m / \partial q = 0.65$  represents the variation of the pitching moment with respect to the elevator control. The derivative  $C_{m_q} = -10$  represents the aerodynamic effects due to rotations of the vehicle while the angle of attack remains zero. Using the parameters given in

Table 6.1, a second-order transfer function representing the pitch angle dynamics is given as:

$$\frac{\theta(s)}{\delta_e(s)} = \frac{85}{s^2 + 40s + 18}. \quad [6.29]$$

Then, using the actuator transfer function [6.26], a proportional derivative compensator with  $K_p = 80$  and  $K_d = 17$  is proposed to stabilize the pitch angle. The roots of the characteristic equation  $0.1s^3 + 5s^2 + 908.8s + 4,098$  are located at  $-22.7 \pm 91.45i$  and  $-4.62$  which are in the left hand side of the complex plane. Hence, the proposed PD controller stabilizes the platform pitch angle system.

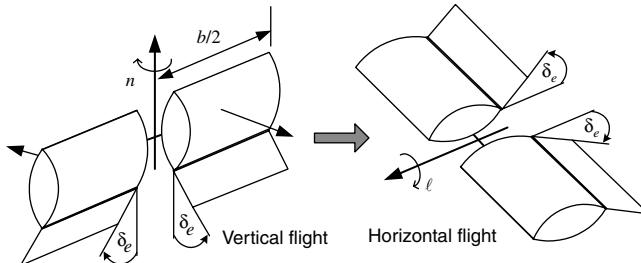
#### 6.3.1.3. Yaw subsystem

To control the vehicle yaw position, it is assumed that the pitch and roll angles are stabilized, and so the roll rate and the pitch rate vanish; therefore, equation of  $\dot{\psi}$  in [6.16] can be written as:

$$\ddot{\psi} = (\rho V^2 S b)(C_{n_\psi} \dot{\psi} + C_{n_{\delta_e}} \delta_e)/4J_{zz}, \quad [6.30]$$

where the vehicle yaw moment  $n$  is given by  $\rho V^2 S b C_n / 4$ ,  $b$  is the wing span and  $C_n = C_{n_\psi} \dot{\psi} + C_{n_{\delta_e}} \delta_e$  is the yawing moment coefficient,  $C_{n_{\delta_e}} = 0.19$  is the variation of the yaw moment with respect to the ailerons positions.  $C_{n_\psi} = 0.19$  is the yaw damping derivative. Note that  $n$  is used to control yaw during hover flight and to control roll during forward flight as shown in Figure 6.5. Using the parameters given in Table 6.1, the yaw transfer function is given by:

$$\frac{\psi(s)}{\delta_e(s)} = \frac{20}{s^2 + 20s}. \quad [6.31]$$



**Figure 6.5.** Yaw control

Then, using the actuator transfer function [6.26], a proportional derivative compensator with  $K_p = 68$  and  $K_d = 17$  is proposed to stabilize the yaw angle. The characteristic equation is  $0.1s^3 + 3s^2 + 224s + 20$  and its roots are located at  $-13 \pm 44.4i$  and  $-3.8$ , therefore, the system is stable.

### 6.3.2. Robust control considering parametric uncertainty

It is usual to handle inaccurate mathematical models. In the previous section, aerodynamic coefficients for a tailsitter had been estimated using Shareware software. Then, mathematical model obtained for the tailsitter vehicle may not be accurate due to the estimation errors and imperfections in the vehicle building process. Thus, in this section, a robust control design and analysis is introduced.

Recall that an attitude dynamic model has been obtained using the parameters values in Table 6.1. It is important to note that some of the aircraft parameters were obtained by experimental tests and the remaining aircraft parameters were estimated using Shareware software. In order to take into account possible measurement errors, we will consider uncertainty in the last row coefficients of the state matrix. In spite of the uncertainty structure in the coefficients of the last row of the matrix, it is always possible to lump the uncertainty such that the resulting polynomial family is a lumped version of the original interval polynomial family.

#### 6.3.2.1. Pitch subsystem

The pitch angle dynamics [6.28] can also be written as in [6.34]. Using the vehicle parameters given in Table 6.1 and the same procedure for the roll subsystem, we obtain

$$\mathbf{A}(\mathbf{q}) = \begin{bmatrix} 0 & 1 \\ -q_0 & -q_1 \end{bmatrix}, \quad [6.32]$$

with  $q_0 = [-21.6, -14.4]$ ,  $q_1 = [-48, -32]$ .

#### 6.3.2.2. Yaw subsystem

The yaw angle dynamics [6.30] can also be written as in [6.34]. Using the vehicle parameters given in Table 6.1 and the same procedure used in roll subsystem, we obtain

$$\mathbf{A}(\mathbf{q}) = \begin{bmatrix} 0 & 1 \\ -q_0 & -q_1 \end{bmatrix}, \quad [6.33]$$

with  $q_0 = 0$ ,  $q_1 = [-24, -16]$ .

#### 6.3.2.3. Roll subsystem

From [6.21] and [6.23], the roll subsystem can be expressed as follows:

$$\dot{\boldsymbol{\eta}} = \mathbf{A}\boldsymbol{\eta} + \mathbf{B}u, \quad [6.34]$$

$$y = \mathbf{C}\boldsymbol{\eta}, \quad [6.35]$$

where

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & -30,000 & -4,450 & -155 \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 3,600 \end{bmatrix},$$

and

$$\boldsymbol{\eta} = \begin{bmatrix} \dot{\phi} \\ \ddot{\phi} \\ -25\dot{\phi} + 5\varphi \\ 625\dot{\phi} - 725\varphi + 600\delta_s \end{bmatrix}, \quad \mathbf{C}^T = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}.$$

Taking into account parameter uncertainty, the following matrix is obtained:

$$\mathbf{A}(\mathbf{q}) = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ q_0 & q_1 & q_2 & q_3 \end{bmatrix}, \quad [6.36]$$

where  $q_0 \in [-10, 10]$ ,  $q_1 \in [-36,000, -24,000]$ ,  $q_2 \in [-5,340, -3,560]$ , and  $q_3 \in [-186, -124]$ .

Now, a robust state feedback control design to stabilize a system with uncertain parameter values can be used. Subsequently, the value set characterization is used to verify the robust stability property when a time delay in the process is considered. To do this, the Bidule–CSyRex roll subsystem [6.34] and [6.36] will be represented in the form:

$$\Sigma_{un} \triangleq \begin{cases} \dot{\boldsymbol{\eta}} = \mathbf{A}(\mathbf{q}^-)\boldsymbol{\eta} + \mathbf{B}u + \mathbf{B}\Gamma(r)\boldsymbol{\eta}, \\ y = \mathbf{C}\boldsymbol{\eta}, \end{cases} \quad [6.37]$$

where

$$\mathbf{A}(\mathbf{q}^-) = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -10 & -36,000 & -5,340 & -186 \end{bmatrix},$$

$$\Gamma(r) = [r_3 \ r_2 \ r_1 \ r_0],$$

where  $r_0 \in [0, 0.0172]$ ,  $r_1 \in [0, 0.4944]$ ,  $r_2 \in [0, 3.3333]$ , and  $r_3 \in [0, 0.0056]$ .

Now, the  $\mathbf{F}$  matrix is defined in such a way that the following condition is satisfied:

$$\Gamma(r)^T \Gamma(r) \leq \mathbf{F} \quad \forall r \in \mathbb{R}, \quad [6.38]$$

and the nominal system will be considered as follows:

$$\Sigma_{nom} \triangleq \begin{cases} \dot{\boldsymbol{\eta}} = \mathbf{A}(\mathbf{q}^-)\boldsymbol{\eta} + \mathbf{B}u, \\ y = \mathbf{C}\boldsymbol{\eta}. \end{cases} \quad [6.39]$$

Then, using the robust control design method described in Chapter 4, the following control law is obtained:

$$u = -35,999\eta_1 - 5,553\eta_2 - 442\eta_3 - 12\eta_4. \quad [6.40]$$

#### 6.3.2.4. Time delay case

All processes have time delays due to sensor information process, actuator time delay, etc. Considering a time delay  $\tau$ , system [6.37] can be rewritten as:

$$\dot{\boldsymbol{\eta}}(t) = \begin{bmatrix} 0 & 1 & \dots & 0 \\ 0 & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & 1 \\ -q_0 & -q_1 & \dots & -q_{n-1} \end{bmatrix} \boldsymbol{\eta}(t) + \begin{bmatrix} 0 \\ 0 \\ 0 \\ r_0 \end{bmatrix} u(t - \tau). \quad [6.41]$$

**PROPOSITION 6.1.-** Consider the Bidule–CSyRex roll subsystem with time delay [6.41], then this system is robustly stable if the same control law [6.40] is used and the maximum time delay is  $\tau_{max} = 1$  s.

**PROOF.–** The uncertain time delay system [6.41] has the following characteristic equation:

$$\begin{aligned} p(s, \mathbf{q}, \mathbf{r}, e^{-\tau s}) = & s^4 + [124,186]s^3 + [3,560,5,340]s^2 + [24,000,36,000]s + \\ & [-10, 10] + (12s^3 + 422s^2 + 5,553s + 35,990)e^{-[0,1]s}. \end{aligned} \quad [6.42]$$

These kind of functions are known as quasi-polynomials. It is clear that the above characteristic equation [6.42] represents an infinite number of quasi-polynomials that have to be considered to verify the robust stability property. This family is defined as follows:

$$\mathcal{P}_\tau \triangleq \left\{ p(s, \mathbf{q}, \mathbf{r}, e^{-\tau s}) : \mathbf{q} \in \mathcal{Q}; \mathbf{r} \in \mathcal{R}; \tau \in [0, \tau_{max}] \right\}, \quad [6.43]$$

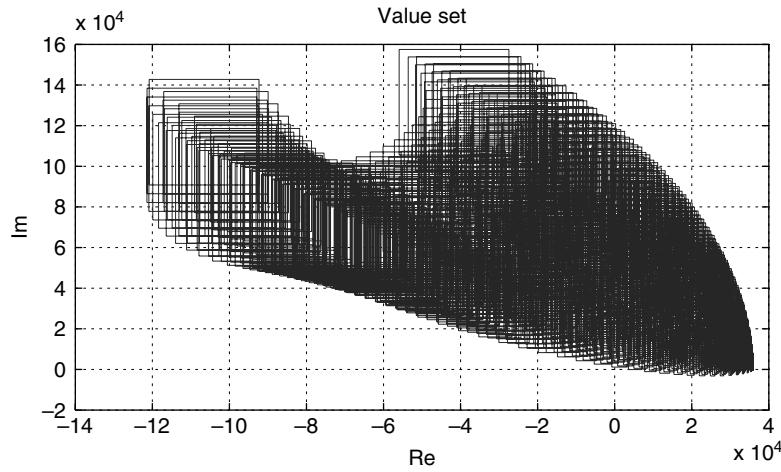
where  $\mathcal{Q}$  and  $\mathcal{R}$  represent the set of uncertainty (see [BAR 94]).

It is clear that the value set of  $P_\tau$  is a set of complex numbers plotted on the complex plane for values of  $\mathbf{q}$ ,  $\mathbf{r}$ ,  $\omega$ , and  $\tau$  inside the defined boundaries. Next, the zero exclusion principle is presented in order to verify the robust condition [BAR 94].

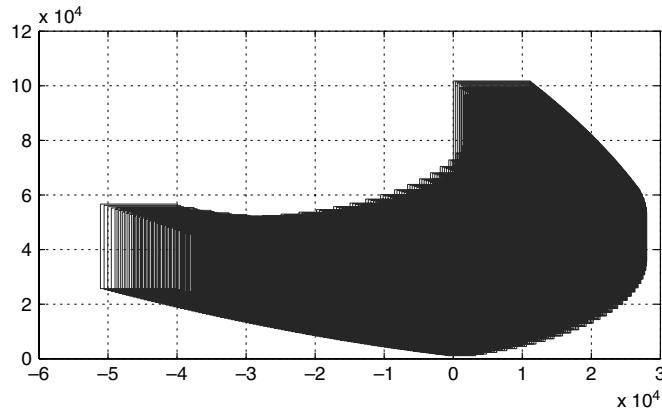
**LEMMA 6.1.**— Consider the characteristic equation [6.42], also called quasi-polynomials. Suppose that [6.42] has at least one stable member. Then the robust stability property of the control system is guaranteed if and only if:

$$0 \notin V_\tau(\omega) \quad \forall \omega \geq 0. \quad [6.44]$$

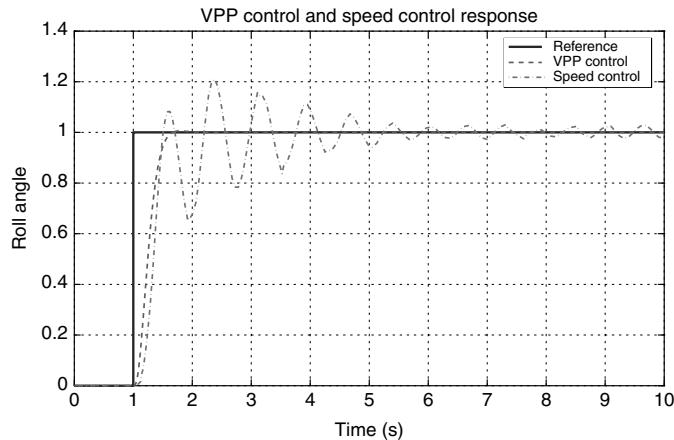
The results presented in [ROM 95] and [ROM 97] permit building the value set  $V_\tau(\omega)$  for the characteristic equation [6.42] and is presented in Figures 6.6 and 6.7.



**Figure 6.6.** Value set of roll subsystem control



**Figure 6.7.** Value set zoom



**Figure 6.8.** Response of the roll control loop

It can be noted that the zero is not included in the value set  $V_\tau(\omega)$ . Then the system [6.41] is then robustly stable.

REMARK 6.1.– Note that pitch and yaw control laws can be obtained by using the robust control design described in Chapter 4.

### 6.3.3. Simulation results

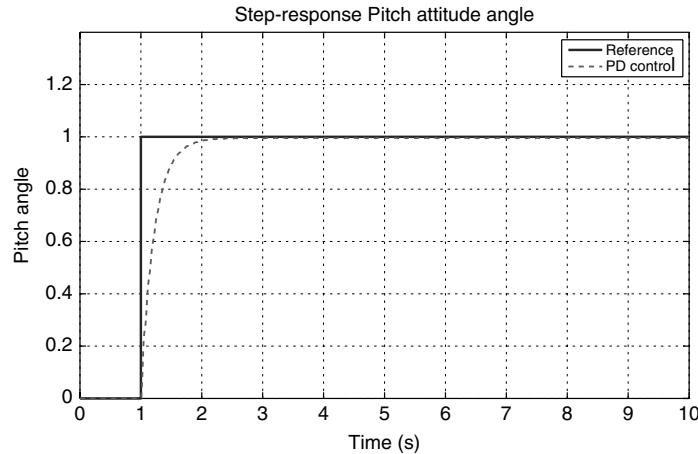
To investigate the behavior of the control stabilization system, several simulations of the model have been run using Matlab Simulink™. This helps determine the flight handling qualities of the vehicle. The following graphs show stable dynamics using both PD and robust controllers.

#### 6.3.3.1. Linear controller

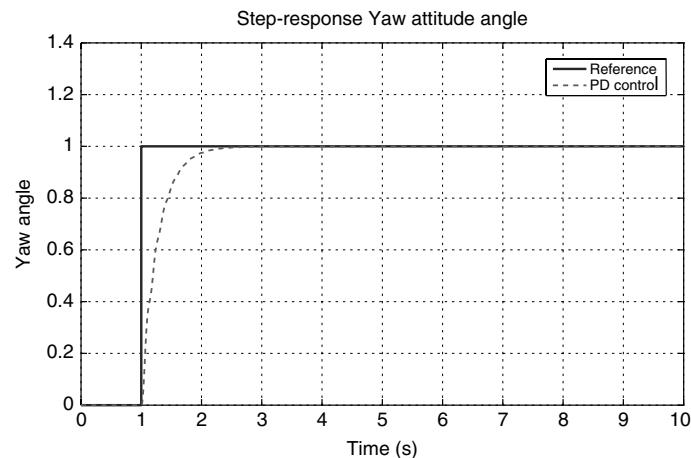
The response of the roll subsystem to a unit step is shown in Figure 6.8. It is clear that VPP control can stabilize the system very fast while speed control cannot stabilize this system at all. For each control loop the step response is evaluated. First, the roll control system is validated in simulation, this system based in the VPP mechanism has been compared with a roll control system based on the speed variation of the rotors. Normally, a control based on the speed variation introduces a time delay, which is caused by the electronics of the speed controller. This time delay provokes instability in the system making the tuning of the controller parameters a very difficult task. Figure 6.8 shows the comparison of the two systems.

The two systems reach the desired value almost at the same time, but in the system using speed variation there are oscillations in the steady state, while the VPP control

quickly stabilizes the system. In the same way, the step responses for the pitch and the yaw closed-loop control systems have been simulated. Their respective responses are shown in Figures 6.9 and 6.10. It can be observed from Figures 6.8–6.10 that the proposed aerial vehicle platform can be used as a tailsitter vehicle. Spoerry *et al.* [SPO 01] have shown the forward flight performance of the miniature UAV named Bidule.



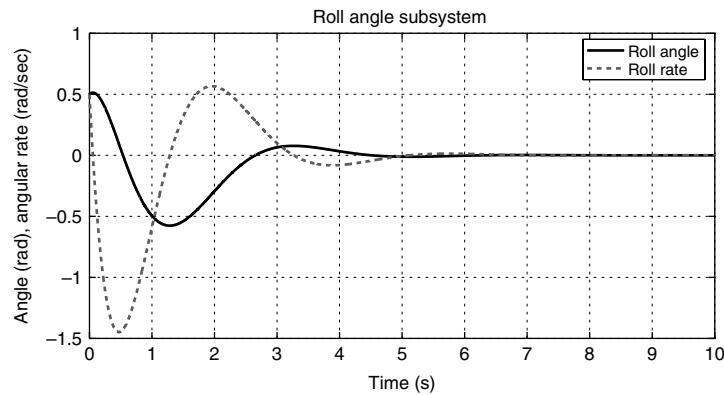
**Figure 6.9.** Response of the pitch control loop



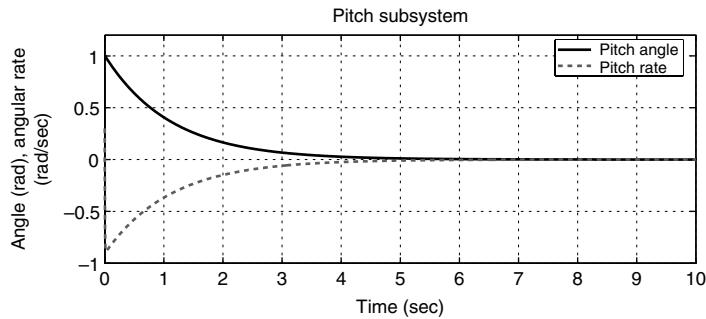
**Figure 6.10.** Response of the roll control loop

### 6.3.3.2. Robust controller

In order to improve handling qualities for the Bidule tailsitter vehicle, a robust control design has been developed. The response of the roll robust control law [6.40] is shown in Figure 6.11. In the same way, the pitch and the yaw closed-loop control systems have been simulated. Their respective responses are shown in Figures 6.12 and 6.13.



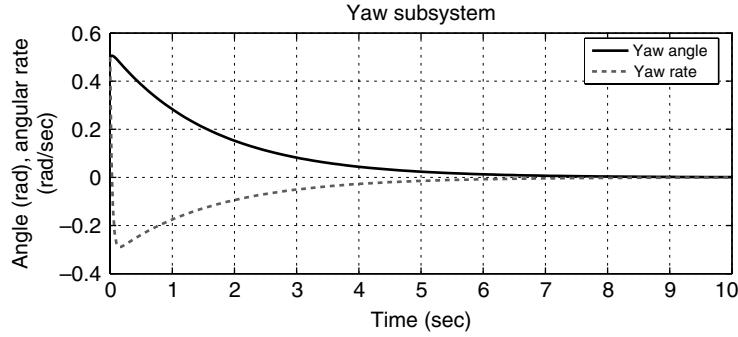
**Figure 6.11.** Response of the roll control loop using robust control approach



**Figure 6.12.** Response of the pitch control loop using robust control approach

### 6.3.4. Experimental results

In this section, qualitative results in flight tests of the tailsitter are discussed. The natural unstable behavior in vertical flight of this tailsitter vehicle makes the manual guidance and control a very difficult task even for experienced human operators. Figure 6.14 shows the vehicle's wild behavior due to the high instability in a test without any automatic control algorithm.



**Figure 6.13.** Response of the yaw control loop using robust control approach

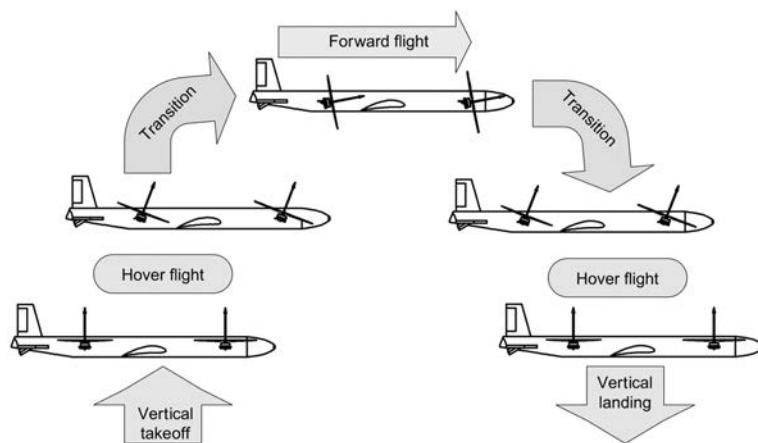


**Figure 6.14.** Bidule-CSyRex with no control

As was seen in previous sections, the first control approach for this vehicle is a simple PD control, which has been chosen because the angular position variables and its derivatives are obtained directly from the IMU. To adjust the control parameters several flight tests were carried out until a good performance of the vehicle was obtained. First, the  $K_d$  gains were adjusted to get a good stiffness in all the angular displacements, then the  $K_p$  gain was adjusted to obtain a good time response to changes of angular position. Figure 6.15 shows the vehicle flying stable when the linear control PD is used. Note that tethers were used for safety purposes only, with satisfactory flight test results used only when all the tethers are slack, thus not supporting the flight platform in any way.



**Figure 6.15.** *Bidule–CSyRex with PD control*

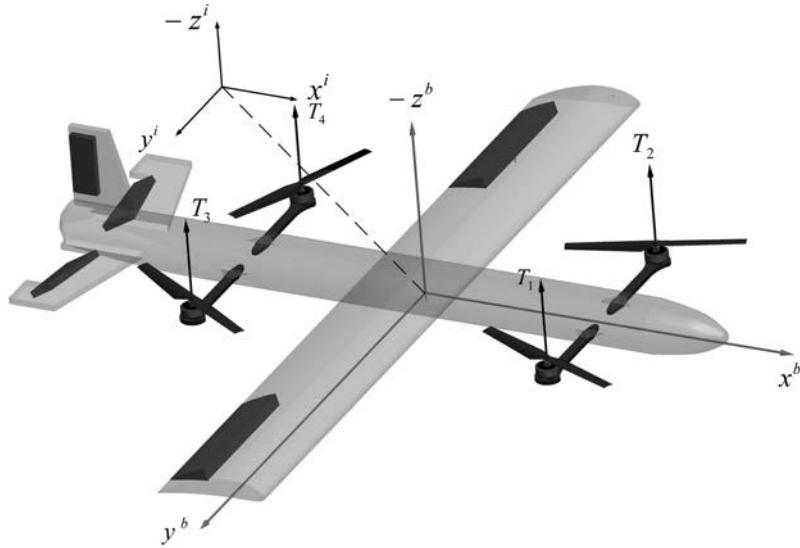


**Figure 6.16.** *Operational transition*

#### 6.4. Quad-tilting rotor convertible MAV

A new tilt-rotor aircraft (quad-plane UAV) that is capable of flying in horizontal and vertical modes (Figure 6.16) is presented in this section. The vehicle is driven by four rotors and has a conventional airplane-like structure, which constitutes a highly nonlinear plant and thus the control design should take into account this aspect. A nonlinear control strategy, consisting of a feedback-linearizable input for altitude control and a hierarchical control (inner–outer loop) scheme for the underactuated dynamic subsystem ( $x$ -position, pitch), is proposed to stabilize the aerial robot within the hovering mode. Backstepping [KHA 02], a Lyapunov-based method is presented to stabilize the vehicle within the airplane mode. Through the use recursive method,

backstepping divides the control problem into a sequence of designs for simpler systems. This mini aerial vehicle is one of the first of its kind among tilt-wing vehicles on that scale range.



**Figure 6.17.** Coordinate systems: inertial frame ( $\mathbf{F}_i$ ) and body-fixed frame ( $\mathbf{F}_b$ )

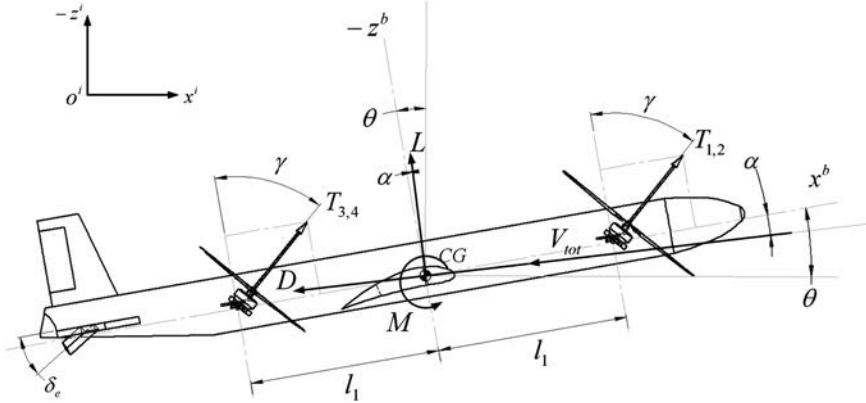
#### 6.4.1. Modeling

This section presents the longitudinal equations of motion as well as the aerodynamics of the vehicle. Owing to the flight profile of the vehicle we distinguish three operation modes depending on the position of the tilt angle  $\gamma$  (Figure 6.18): (1) *Hover Flight (HF)*, where the aircraft behaves as a rotary-wing platform ( $|\gamma| \leq \pi/6$ ), (2) *Slow-Forward Flight (SFF)* ( $\pi/6 < |\gamma| \leq \pi/3$ ), and finally (3) *Fast-Forward Flight (FFF)*, where the aerial robot behaves as a pure airborne vehicle ( $\pi/3 < |\gamma| < \pi/2$ ).

1) During the *HF* the 3D vehicle's motion relies only on the rotors. Within this phase the vehicle features VTOL flight profile. The controller for this regime disregard the aerodynamic terms due to the negligible translational speed.

2) It is possible to distinguish an intermediate operation mode, the *SFF*, which links the two flight conditions, the *HF* and the *FFF*. This is probably the most complex dynamics.

3) *FFF* regime mode (Aft position), in this flight mode the aircraft has gained enough speed to generate aerodynamic forces to lift and control the vehicle motion.



**Figure 6.18.** Free-body scheme showing the forces acting on the quad-tilting MAV

#### 6.4.1.1. Aerodynamics

It is important to consider these forces properly because they are fundamentally affected by the vehicle's motion and thus they alter the basic dynamics involved. The analysis used in this chapter is based on a combination of a low-order panel method aerodynamic model coupled with a simple actuator disk model of the flow induced by the propellers. In order to proceed with the aerodynamic analysis, it is worth mentioning the following assumptions:

- **A1.** The vehicle has a rigid body, i.e. the flexibility of the aircraft wings or fuselage will be neglected.
- **A2.** Non-varying mass is considered ( $\dot{m}(t) = 0$ ).
- **A3.** The aerodynamic center (AC) and the center of gravity (CG) are coincident.

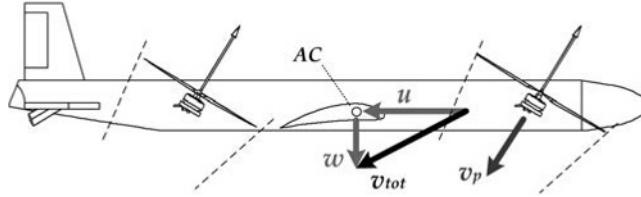
In order to determine the aerodynamic forces exerted on the vehicle, we need to know both the direction and velocity of the total airflow vector. We can identify three wind vectors acting on the vehicle: the airflow speed  $V_p$  produced by the rotors, the  $V_b$  airflow generated by the translational motion of the body ( $U, W$ ), and a third component due to the external wind (disturbance)  $V_e$ , generally unknown. Hence, the total wind vector in the body frame can be written as:

$$\mathbf{V}_{tot} = \mathbf{V}_p(\gamma) + \mathbf{V}_b + \mathbf{V}_e, \quad [6.45]$$

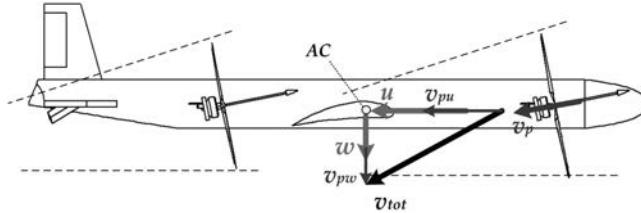
where  $\mathbf{V}_{tot} = (v_u, v_w)^T$ . The total wind vector  $\mathbf{V}_{tot}$  experienced by the wing varies depending on the flight mode. Within the *HF* and *SFF* regimes the wing is not washed by the propeller airflow  $\mathbf{V}_p$  (Figure 6.19), while in the *FFF* mode, it is assumed that the wing is significatively submerged (Figure 6.20) by  $\mathbf{V}_p$ . Therefore, the propeller

slipstream  $\mathbf{V}_p$  is disregarded in *HF* and *SFF*. To include the behavior of  $\mathbf{V}_p$  in the equations, let us introduce the following function:

$$\xi(\gamma) = \begin{cases} 0 & \text{if } \gamma \leq \frac{\pi}{3} \\ 1 & \text{if } \gamma > \frac{\pi}{3} \end{cases} \quad [6.46]$$



**Figure 6.19.** Airflow profile generated by the rotors during the flight envelope.  
Relative wind velocity in HF and SFF modes



**Figure 6.20.** Relative wind velocity in FFF mode

The parallel wind velocity  $v_u$  and the normal wind velocity  $v_w$  encompass the velocity that the vehicle experiments through the air and the corresponding components of  $\mathbf{V}_p$  due to the tilting of the rotors and the aleatory external wind  $\mathbf{V}_e$ , i.e.

$$\mathbf{V}_u = (u + \xi(\gamma)v_p \sin(\gamma) + v_{e_u})\mathbf{i}_b, \quad [6.47]$$

$$\mathbf{V}_w = (w + \xi(\gamma)v_p \cos(\gamma) + v_{e_w})\mathbf{k}_b. \quad [6.48]$$

Assuming purely axial flow into the propellers, simple actuator disk theory [STE 04] gives the induced propeller velocity for the  $i$ th rotor as:

$$v_{p_i} = \sqrt{\frac{2T_i}{\rho A_p}}, \quad [6.49]$$

where  $A_p$  is the total disk area of the propeller and  $\rho$  the air density. Figure 6.18 shows the aerodynamic forces on a small UAV with a tilt angle  $\gamma$ . The forces consist of a lift

force,  $L$ , perpendicular to the total flow vector,  $\mathbf{V}_{tot}$ , a drag force  $D$  parallel to  $\mathbf{V}_{tot}$ , and the airfoil's pitching moment,  $M$ , about the positive cartesian  $y$ -axis. The above discussion can be summarized by:

$$\begin{aligned} C_l &= C_{l_\alpha} \alpha, \\ C_d &= C_{d_p} + C_{d_i}, \\ C_m &= C_{m_\alpha} \alpha, \end{aligned}$$

where these equations are standard aerodynamic non-dimensional lift, drag, and moment coefficients.<sup>1</sup> To obtain the lift and drag forces and the pitching moment on the aircraft it is only necessary to obtain the total wind velocity vector  $\mathbf{V}_{tot}$  (see [6.45]), the angle of attack, and the aerodynamic parameters  $C_{l_\alpha}, C_{l_\delta}, C_d, C_{m_\delta}, C_{m_\alpha}$  which depend on the geometry of the vehicle.

$$\begin{aligned} L &= \frac{1}{2} C_l \rho V_{tot}^2 S, \\ D &= \frac{1}{2} C_d \rho V_{tot}^2 S, \\ M &= \frac{1}{2} C_m \rho V_{tot}^2 S \bar{c}. \end{aligned}$$

In these equations  $S$  and  $\bar{c}$  are the area and the wing chord, respectively. The angle of attack  $\alpha$  and the magnitude of  $\mathbf{V}_{tot}$  are obtained through the following equations:

$$\alpha = \arctan(v_w/v_u), \quad [6.50]$$

$$|\mathbf{V}_{tot}| = \sqrt{v_w^2 + v_u^2}. \quad [6.51]$$

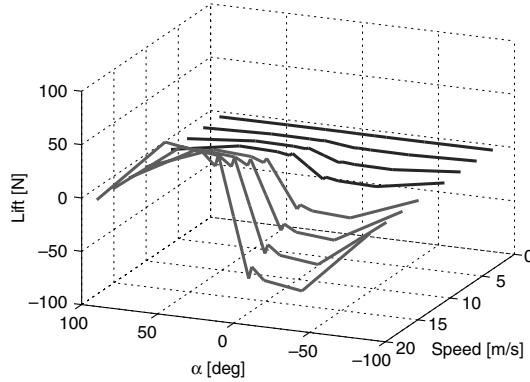
The lift force will depend on the velocity  $\mathbf{V}_{tot}$  and the angle of attack. Figure 6.21 represents the different values of lift for several speed conditions.

The vector that contains the set of forces applied to the quad-plane (Figure 6.18) is given using [6.7], and the corresponding vectorial equation grouping the moments exerted about  $CG$  is written using equation [6.18]. The corresponding scalar equations modeling the forces and moments applied to the vehicle are written as [FLO 11]:

$$\begin{aligned} m\ddot{x} &= -T_{3,4} \sin(\theta + \gamma) - T_{1,2} \sin(\theta + \gamma) - L \sin(\theta - \alpha) - D \cos(\theta - \alpha), \\ m\ddot{z} &= D \sin(\theta - \alpha) - T_{1,2} \cos(\theta + \gamma) - L \cos(\theta - \alpha) - T_{3,4} \cos(\theta + \gamma) + g, \\ J_{yy}\ddot{\theta} &= M + l_1(-T_{3,4} \cos \gamma + T_{1,2} \cos \gamma) + I_p p(\omega_1 - \omega_2 - \omega_3 + \omega_4). \end{aligned} \quad [6.52]$$

---

<sup>1</sup>  $C_*$  slopes are obtained from the software XFOIL.



**Figure 6.21.** Lift values for different velocities

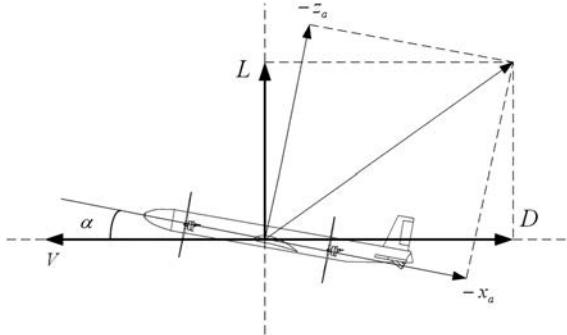
#### 6.4.1.2. FFF mathematical model

In this regime, the vehicle essentially behaves as an airplane, thus we can consider the common longitudinal aircraft model [STE 92]. In addition to the body-axis equations, it is important to express the equations of motion in the wind axis because the aerodynamic forces act in this axis and the magnitude of  $V_{tot}$  (written as  $V$  from here onwards);  $\alpha$  can be expressed in terms of  $u$  and  $w$ . This reference system is used for translational equations because the angle of attack and the velocity are either directly measurable or closely related to directly measurable quantities, while the body-axis velocities ( $u, w$ ) are not. The equations of motion take the form:

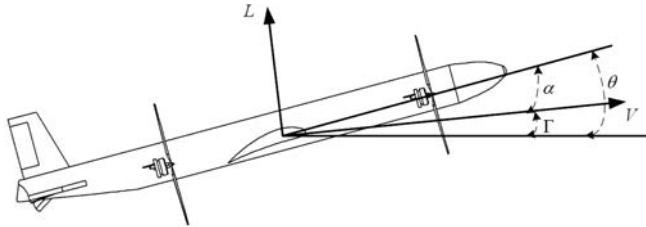
$$\begin{aligned}\dot{V} &= \frac{1}{m}[-D + T_t \cos \alpha - mg(\cos \alpha \sin \theta - \sin \alpha \cos \theta)], \\ \dot{\alpha} &= \frac{1}{Vm}[-L - T_t \sin \alpha + mg(\cos \alpha \cos \theta + \sin \alpha \sin \theta)] + q, \\ \dot{\theta} &= q.\end{aligned}\quad [6.53]$$

The angles  $\theta$  and  $\alpha$  lie in the same vertical plane above the north-east plane (Figure 6.22), and their difference is the flight-path angle  $\Gamma = \theta - \alpha$  (Figure 6.23). Under this definition and from the last equation of [6.52] we obtain the following mathematical model:

$$\begin{aligned}\dot{V} &= \frac{1}{m}[-D + T_t \cos \alpha - mg \sin \Gamma], \\ \dot{\alpha} &= \frac{1}{Vm}[-L - T_t \sin \alpha + mg \cos \Gamma] + q, \\ \dot{\theta} &= q, \\ \dot{q} &= \frac{1}{J_{yy}} M.\end{aligned}\quad [6.54]$$



**Figure 6.22.** Wind-axis reference frame



**Figure 6.23.** Flight-path angle definition

#### 6.4.2. Transition

The flight envelope of the vehicle encompasses different flight conditions, achieved by means of the collective angular displacement of the rotors. Indeed, it is this tilting that provides a *continuous mechanism* to perform the operational transition. To illustrate this, let us consider the following scenario:

- $T_t \geq mg$ , i.e. the vehicle flies at a stabilized altitude.
- $\theta \approx 0$ , i.e. stabilized vertical flight.

It is clear that as  $\gamma$  is tilted the horizontal velocity increases, while the vertical velocity is reduced (Figures 6.24 and 6.25). These facts affect proportionally to the forces coming from the rotors and the wing.

Thus, both vertical and horizontal controllers can still be used at the same time whose actions are controlled by  $\gamma$ . The vertical collective thrust is gradually reduced inhibiting the action of vertical controller and allowing the action of the horizontal controller and vice versa. So, for example, for larger values of  $\gamma$ , i.e.  $\gamma > 45$ , the rotorcraft behaves more like a classical airplane. As the vehicle is gaining speed due to rotors tilt ( $\gamma$ ), aerodynamic forces arise. For this reason we consider that the control of vertical and forward flight are active during the whole flight envelope.

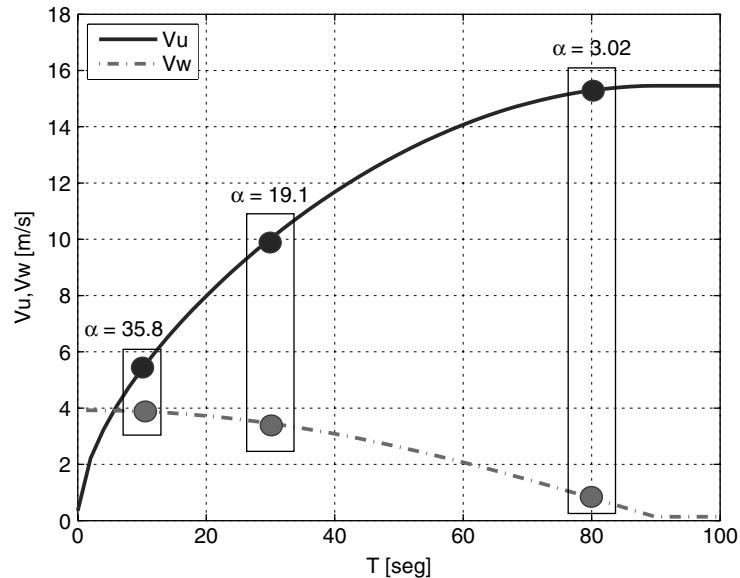


Figure 6.24. Behavior velocities during the tilting of the rotors

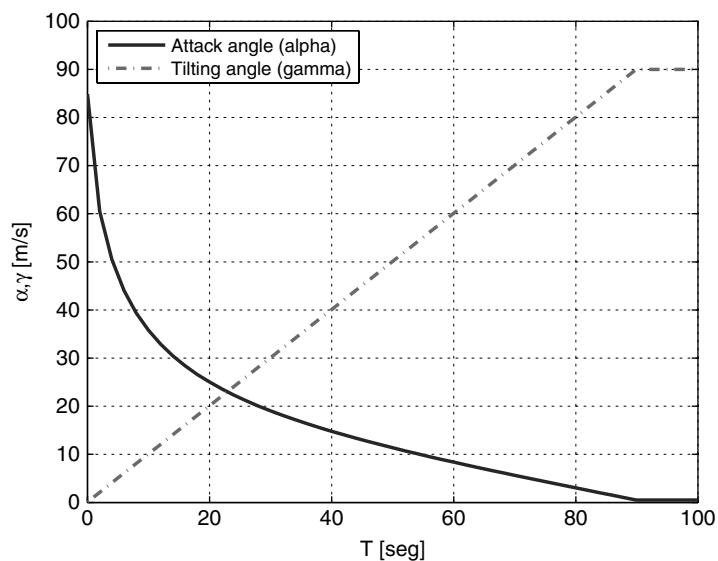


Figure 6.25. Behavior of angles  $\gamma$  and  $\alpha$  during the tilting of the rotors

#### 6.4.3. Control strategy for hover flight mode

The vertical flight of the quad-plane represents a challenging stage because the aircraft's vertical dynamics are naturally unstable. In this regime, the quad-plane aerial robot aims to emulate the flight behavior of a quadrotor which features a non-conventional quadrotor design, i.e. an asymmetrical H-form structure.

Vertical flight regime encompasses two dynamic subsystems: the altitude dynamics, actuated by the thrust  $T$ , and the horizontal translational motion, generated by the pitch attitude. Taking into account the item (1) presented in section 6.4.1, which implies  $\alpha \approx 0$ , since  $\gamma = 0$ , we can consider a simplified model from which is derived the controller in hover flight regime.

For simplicity, we consider that the gyroscopic moment is very small. These considerations allow us to rewrite [6.52] as:

$$\begin{aligned}\ddot{X} &= -\left(\frac{T_{3,4} + T_{1,2}}{m}\right)(\sin \theta), \\ \ddot{Z} &= -\left(\frac{T_{3,4} + T_{1,2}}{m}\right)(\cos \theta) + g, \\ \ddot{\theta} &= -\left(\frac{l_1}{J_{yy}}\right)(-T_{3,4} + T_{1,2}).\end{aligned}\quad [6.55]$$

If we rename the total thrust as  $T_t = T_{3,4} + T_{1,2}$ , and the difference of these thrusts as  $T_d = T_{1,2} - T_{3,4}$ . Then,

$$\ddot{X} = -\frac{T_t}{m} \sin \theta, \quad [6.56]$$

$$\ddot{Z} = -\frac{T_t}{m} \cos \theta + g, \quad [6.57]$$

$$\ddot{\theta} = -\frac{l_1}{J_{yy}} T_d. \quad [6.58]$$

Therefore, we have derived a simple model, suitable for controller design. The altitude [6.57] can be stabilized through a feedback-linearizable input through the total thrust  $T_t$

$$T_t = -\frac{mu_z - mg}{\cos(\theta)}, \quad [6.59]$$

where  $u_z = -k_{p_z}(z - z^d) - k_{d_z}\dot{z}$  with  $k_{p_z}, k_{d_z} > 0$  and  $z_d$  is the desired altitude. Since the vehicle works in an area close to  $\theta \approx 0$ , the singularity is avoided. For the subsystem [6.56] and [6.58], a two-level hierarchical control scheme is used to stabilize its dynamics. The outer-loop control stabilizes the translational motion (slow

dynamics [KOK 86]) along the  $x$ -axis, while the inner-loop control stabilizes the attitude (fast dynamics). Introducing [6.59] into [6.56] and assuming that  $z \approx z^d$ , namely  $u_z \rightarrow 0$ , lead to

$$\ddot{x} \approx -g \tan \theta = -g \tan u_x. \quad [6.60]$$

For the horizontal motion [6.60],  $\theta$  can be considered as virtual control input  $u_x$ . However, it is a state not an actual control. Given that  $\dot{\theta}^d$  is slowly time-varying, we will assume that the  $x$ -dynamics converges slower than the  $\theta$ -dynamics. The reference for the inner-loop systems is:

$$u_x = \theta^d = \arctan\left(\frac{-v_x}{g}\right), \quad [6.61]$$

$$\dot{u}_x = \dot{\theta}^d \approx 0, \quad [6.62]$$

where  $v_x = k_{v_x} \dot{x} + k_{p_x} x$  with  $k_{v_x}, k_{p_x} > 0$ . Using the linearizing control input [6.61] in [6.60] yields:

$$\ddot{x} = v_x, \text{ provided that } \tilde{\theta} = 0 \text{ (i.e. } \theta = \theta^d).$$

As the previous equation shows, the success of the outer-loop controller relies directly on the inner-loop attitude control performance, thus the inner-loop controller must guarantee the stabilization of the attitude around the reference. For this reason, the stability analysis of the inner-loop controller is presented next. Consider the following positive function, which is an unbounded function,

$$V(\tilde{\theta}, \dot{\theta}) = \frac{1}{2} J_{yy} \dot{\theta}^2 + \ln(\cosh \tilde{\theta}). \quad [6.63]$$

Using [6.56] its corresponding time derivative yields:

$$\dot{V}(\tilde{\theta}, \dot{\theta}) = J_{yy} \dot{\theta} \left( -\frac{l_1}{J_{yy}} T_d \right) + \dot{\tilde{\theta}} \tanh \tilde{\theta}. \quad [6.64]$$

Considering  $\dot{\tilde{\theta}} = \dot{\theta}$ , thus [6.64] may be rewritten as:

$$\dot{V}(\tilde{\theta}, \dot{\theta}) = \dot{\theta} \left( -l_1 T_d + \tanh \tilde{\theta} \right). \quad [6.65]$$

Using the control input:

$$T_d = \frac{\tanh \tilde{\theta} + \tanh \dot{\theta}}{l_1}, \quad [6.66]$$

in [6.65] yields:

$$\dot{V}(\tilde{\theta}, \dot{\theta}) = -\dot{\theta} \tanh \dot{\theta}, \quad [6.67]$$

where  $\dot{V}(\tilde{\theta}, \dot{\theta}) \leq 0$ . Therefore, the origin  $(\tilde{\theta}, \dot{\theta})$  is stable and the state vector remains bounded. The asymptotic stability analysis can be obtained from LaSalle's Theorem. Therefore,  $\tilde{\theta} \rightarrow 0$  and  $\dot{\theta} \rightarrow 0$  as  $t \rightarrow \infty$ .

#### 6.4.4. Control strategy for forward flight mode

In this section, the flight path angle  $\Gamma$  will be controlled using the *backstepping* algorithm taking the following approximations into consideration:

- The air speed is assumed constant,  $\dot{V} = 0$  [MAT 07].
- From the definition of flight-path angle, the dynamics  $\dot{\Gamma} = \dot{\theta} - \dot{\alpha}$  yields  $\dot{\Gamma} = 1/mV[T_t \sin \alpha + L - mg \cos \Gamma]$ .
- The thrust term  $T_t \sin \alpha$  in [6.54] will be neglected as it is generally much smaller than lift.
- $C_m = C_{m\delta}(\alpha)\delta$ , since the main contribution to  $M$  is provided by the elevator.

With these considerations in mind and using the change of coordinates  $\zeta = \Gamma - 1/2\pi$ , the system [6.54] may be expressed as:

$$\begin{aligned}\dot{\zeta} &= -\frac{g \cos(z + 1/2\pi)}{V} + \frac{C_{l\alpha}\alpha}{mV}, \\ \dot{\alpha} &= \frac{g \cos(z + 1/2\pi)}{V} - \frac{C_{l\alpha}\alpha}{mV} + q, \\ \dot{q} &= \frac{1}{J_{yy}}C_{m\delta}\delta.\end{aligned}\quad [6.68]$$

Equation [6.69] is now in *feed forward* form for *backstepping* procedure. For notational simplification, let

$$\begin{aligned}\dot{x} &= f(x) + \xi_1, \\ \dot{\xi}_1 &= f_1(x, \xi_1) + \xi_2, \\ \dot{\xi}_2 &= f_2 + g_2(\xi_1)u,\end{aligned}\quad [6.69]$$

with

$$\begin{aligned}4x &= \frac{mV\Gamma}{C_{l\alpha}}; & f(x) &= -\frac{g}{V} \cos\left(\frac{C_{l\alpha}x}{mV}\right), \\ \xi_1 &= \alpha; & f_1(x, \xi_1) &= \frac{g}{V} \cos\left(\frac{C_{l\alpha}x}{mV}\right) - \frac{C_{l\alpha}\xi_1}{mV}, \\ \xi_2 &= q; & f_2 &= 0, \\ u &= \delta; & g_2(\xi_1) &= \frac{1}{J_y}C_{m\delta}.\end{aligned}\quad [6.70]$$

Defining the following error states:

$$\begin{aligned} e &\triangleq x - x_{des}, \\ e_1 &\triangleq \xi_1 - \xi_{1,des}, \\ e_2 &\triangleq \xi_2 - \xi_{2,des}. \end{aligned} \quad [6.71]$$

Now, following the backstepping procedure differentiating the first equation in [6.71] yields:

$$\dot{e} = f(x) + \xi_{1,des} + e_1 - \dot{x}_{des}, \quad [6.72]$$

where  $\xi_{1,des}$  is viewed as a virtual control for the last equation, choosing as  $\xi_{1,des} = -f(x) - k_1 e + \dot{x}_{des}$ . Then substituting this virtual control in [6.72], we have

$$\dot{e} = -ke + e_1. \quad [6.73]$$

Repeating the same procedure, differentiating  $e_1$  yields:

$$\dot{e}_1 = f_1(x, \xi_1) + e_2 + \xi_{2,des} - \dot{\xi}_{1,des}. \quad [6.74]$$

Let  $\xi_{2,des} = -f_1(x, \xi_1) - e - k_1 e_1 + \dot{\xi}_{1,des}$  so that

$$\dot{e}_1 = -e - k_1 e_1 + e_2. \quad [6.75]$$

As a last step, now the real control signal is obtained in a similar way. Differentiating  $e_2$  yields:

$$\dot{e}_2 = f_2 + g_2(\xi_1)u - \dot{\xi}_{2,des}. \quad [6.76]$$

Let

$$u = \frac{1}{g_2(\xi_1)} \left[ -f_2 - e_1 - k_2 e_2 + \dot{\xi}_{2,des} \right] = u(\ddot{z}_d, \ddot{z}_d, \dot{z}_d, z_d, e, e_1, e_2), \quad [6.77]$$

so that

$$\dot{e}_2 = -e_1 - k_2 e_2. \quad [6.78]$$

It is important to ensure that  $g_2(\xi_1) \neq 0$ , which occurs only with big enough negative values of  $\alpha$ . These values are assumed to be impossible to achieve in standard operation of the airplane, so avoiding division by zero. Equations [6.73], [6.75], and [6.78] expressed in vectorial form:

$$\dot{e} = -Ke + Se, \quad [6.79]$$

$S = -S^T$  satisfies  $e^T S e = 0, \forall e$ , so that with the Lyapunov-candidate-function  $V(e) = 1/2e^T e$ , and the time derivative evaluated in the trajectories yields:

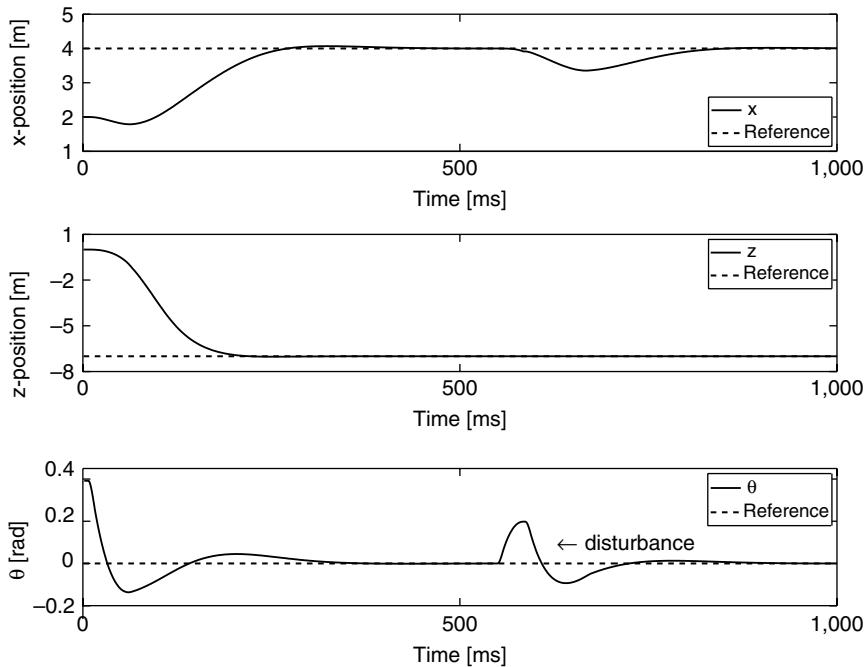
$$\dot{V}(e) = e^T (-Ke + Se) = -e^T K e < 0, \forall e. \quad [6.80]$$

This proves that the above differential equation is asymptotically stable about the origin.

#### 6.4.5. Simulation results

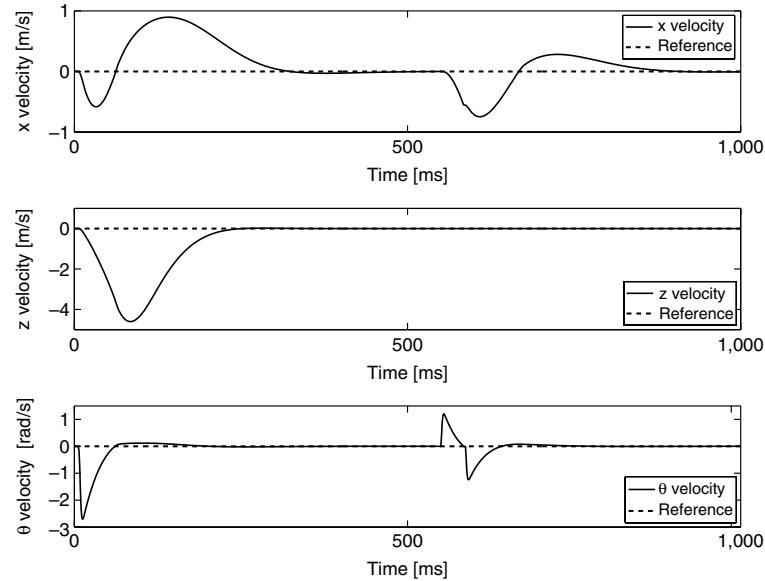
##### 6.4.5.1. HF mode

The performance of the nonlinear controller presented in the previous section is evaluated on the dynamic model [6.56] in MATLAB/Simulink. We started the quad-plane at the position  $(x_0, z_0, \theta_0) = (2, 0, \pi/8)$  and  $(\dot{x}, \dot{z}, \dot{\theta}) = (0, 0, 0)$ . The aircraft had the task of performing hover flight at  $(x^d, z^d, \theta^d) = (4, -7, 0)$ . Figures 6.26 and 6.27 show the evolution and convergence of the states  $(x, \dot{x}, z, \dot{z}, \theta, \dot{\theta})$  to the desired references with the initial conditions mentioned above. It is important to note that position and angle references are tracked with negligible steady state errors. The controller is robust in the presence of a perturbation in  $t = 600$  ms with a magnitude of  $1/8\pi$  radians, as seen in Figures 6.26 and 6.27.

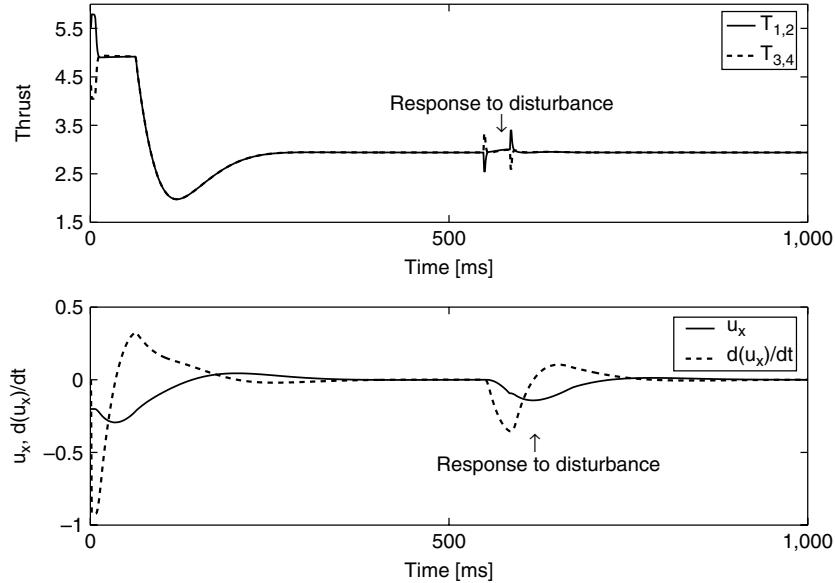


**Figure 6.26.** Position and attitude of the vehicle under disturbance condition

The control inputs are depicted in Figure 6.28, which shows the reaction to the disturbance.



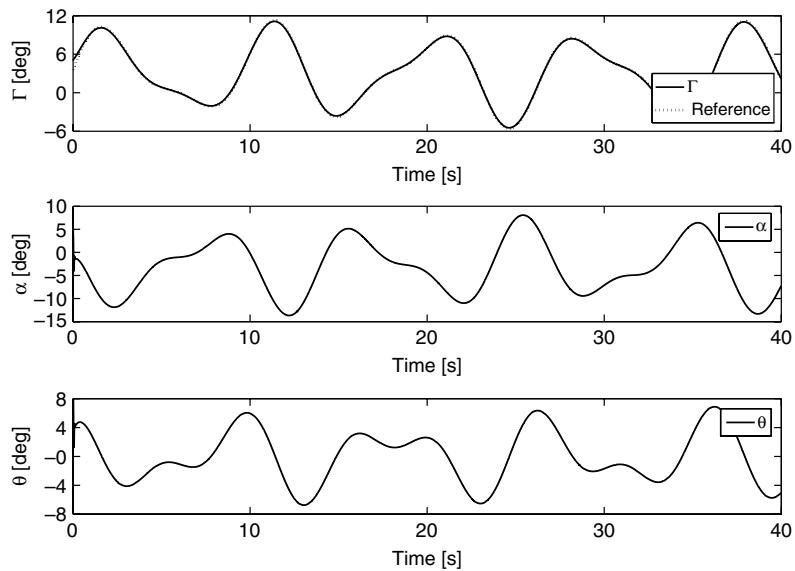
**Figure 6.27.** Derivatives of the position and attitude of the vehicle under disturbance condition



**Figure 6.28.** UAV's control inputs and response to disturbance

#### 6.4.5.2. FFF mode

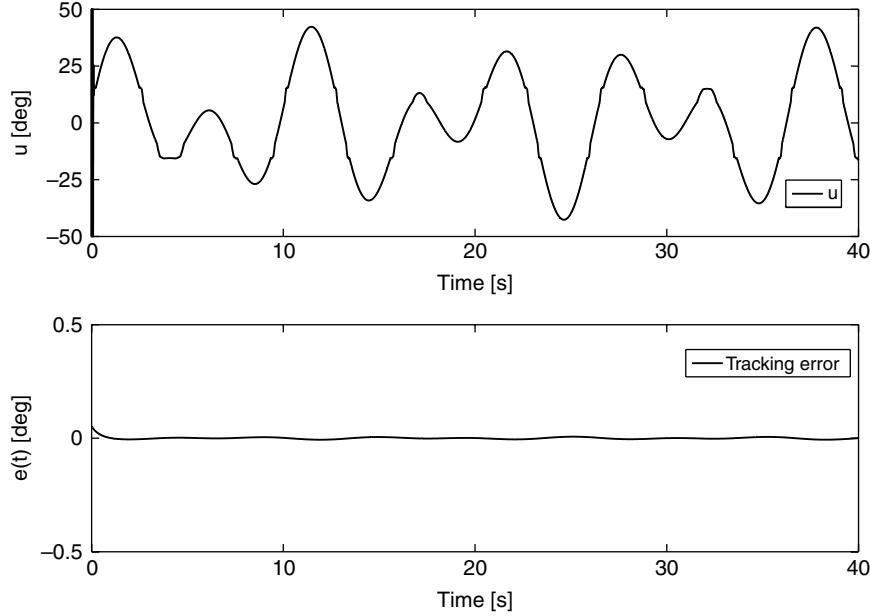
After the vehicle experiences the transition flight, its behavior is like an airplane. We have considered the next initial conditions for purposes of simulation:  $\Gamma_0 = 5$ ,  $\alpha_0 = 5$ , and  $\theta_0 = 10$ . The aircraft had the task of tracking a trajectory shown in the first part of Figure 6.29. This figure shows the evolution and tracking trajectory of the states ( $\Gamma$ ,  $\alpha$ ,  $\theta$ ) to the desired reference, with the initial conditions mentioned above. It is important to note that angle references are tracked with negligible steady state errors.



**Figure 6.29.** Trajectory tracking

## 6.5. Concluding remarks

In this chapter, the Newton–Euler approach has been applied to obtain the dynamical model for miniature UAVs such as tailsitters and convertible UAVs. Since tailsitter UAVs represent a challenging problem for automatic control, a prototype with a VPP mechanism has been developed. The VPP system has been proposed as a solution to time-delay problems induced by speed controllers for brushless motors. In this case, first, we have explored the feasibility of the proposed scheme in hover flight by implementing a linear PD control. Second, we have developed a robust controller to improve the performance of the prototype. Also, using the value set, a graphical tool, we have obtained a maximum time delay margin in the input that the system can handle. Robust control was validated with simulations with good results. This gives an insight of what may be expected in future experimental results for this prototype.



**Figure 6.30.** Control and error defined by  $e = \Gamma - \Gamma_d$

The longitudinal dynamics of the quad-plane including its aerodynamics are derived at the hover and forward flight operating mode. The proposed control strategies were evaluated, at simulation level, for the nonlinear dynamic model, obtaining satisfactory results. The proposed control algorithm is based on an inner–outer loop scheme since it is suitable for implementation purposes.

For energy-saving purposes during forward flight (airplane mode), it is proposed that the vehicles can lead their orientation toward wind velocity vector. To achieve this objective, the vehicles could easily rotate their orientation (yaw movement in helicopter mode) and once the wind vector is addressed, switch to airplane mode. In the quad-plane configuration, this process could be quite simple, since the vehicle is always maintained with the roll and pitch angles close to zero, which is not possible in the case of a tailsitter configuration since within vertical mode the wing surface is highly vulnerable to wind gusts.

## 6.6. Bibliography

- [BAR 94] BARMISH B., *New Tools for Robustness of Linear Systems*, Macmillan, 1994.
- [CAS 05] CASTILLO P., LOZANO R., DZUL A., *Modelling and Control of Mini-Flying Machines*, Springer-Verlag, 2005.

- [ESC 06] ESCAREÑO J., SALAZAR S., LOZANO R., “Modeling and control of a convertible VTOL aircraft”, *45th IEEE Conference on Decision and Control*, San Diego, California, 13–15 December 2006.
- [ESC 07] ESCAREÑO J., STONE H., SANCHEZ A., LOZANO R., “Modeling and control strategy for the transition of a convertible UAV”, *European Control Conference (ECC07)*, Kos, Greece, 2–5 July 2007.
- [ETK 96] ETKIN B., REID L., *Dynamics of Flight Stability and Control*, John Wiley and Sons, 1996.
- [FLO 11] FLORES G., ESCAREÑO J., LOZANO R., SALAZAR S., “Quad-tilting rotor convertible MAV: modeling and real-time hover flight control”, *Journal of Intelligent and Robotic Systems*, 2011.
- [GRE 06] GREEN W.E., OH P.Y., “Autonomous hovering of a fixed-wing micro air vehicle”, *International Conference on Robotics and Automation*, Orlando, Florida, USA, 15–19 May 2006.
- [HEP 06] HEPPERLE M., Aerodynamics for model aircraft, Technical Report, 2006.
- [KAN 01] KANNAN N., SEETHARAMA M., “Longitudinal infinity stability augmentation system for a thrust-vectorized unmanned aircraft”, *Journal of Guidance, Control and Dynamics*, vol. 28, no. 6, pp. 1240–1250, 2001.
- [KHA 02] KHALIL H., *Nonlinear Systems*, Prentice Hall, 2002.
- [KOK 86] KOKOTOVIC P., O'REILLY P., KHALIL H., *Singular Perturbation Methods in Control: Analysis and Design*, Academic Press, 1986.
- [MAT 07] MATES D., HAGSTROM M., “Nonlinear analysis and synthesis techniques for aircraft control”, *Lecture Notes in Control and Information Sciences*, Springer, vol. 365, pp. 230–234, 2007.
- [NON 07] NONAMI K., “Prospect and recent research and development for civil use autonomous unmanned aircraft as UAV and MAV”, *Journal of System Design and Dynamics*, vol. 1, 2007.
- [ONE 08] ONER K., CETINSOY E., UNEL M., AKSIT M., KANDEMIR I., GULEZ K., “Dynamic model and control of a new quadrotor unmanned aerial vehicle with tilt-wing mechanism”, *World Academy of Science, Engineering and Technology*, vol. 45, 2008.
- [ROM 95] ROMERO G., COLLADO J., “Robust stability of interval plants with perturbed time delay”, *Proceedings of the IEEE American Control Conference*, Seattle, WA, USA, pp. 326–327, 1995.
- [ROM 97] ROMERO G., Analysis of robust stability of dynamics systems with time delay, PhD Thesis, FIME–UANL, 1997.
- [SNY 00] SNYDER D., “The Quad tiltrotor: its beginning and evolution”, *Proceedings of the 56th Annual Forum, American Helicopter Society*, Virginia Beach, Virginia, May 2000.
- [SPO 01] SPOERRY T., WONG K.C., “Design and development of a micro air vehicle (MAV) concept: Project Bidule”, *9th Australian International Aerospace Congress (AIAC)*, Canberra, Australia, 6–8 March 2001.

- [STE 04] STENGEL R., *Flight Dynamics*, Princeton University Press, 2004.
- [STE 92] STEVENS B., LEWIS F., *Aircraft Control and Simulation*, John Wiley and Sons Inc., 1992.
- [STO 02] STONE H., “Aerodynamic modelling of a wing-in-slipstream tailsitter UAV”, *AIAA Biennial International Powered Lift Conference*, Williamsburg, Virginia, 5–7 November 2002.
- [STO 04] STONE H., “Control architecture for a tailsitter unmanned air vehicle”, *5th Asian Control Conference*, Melbourne, Australia, 2004.
- [WON 07] WONG K., GUERRERO J., LARA D., LOZANO R., “Attitude stabilization in hover flight of a mini tail-sitter UAV with variable pitch propeller”, *IEEE International Conference on Intelligent Robots and Systems*, San Diego, California, USA, 2007.

## Chapter 7

# Flight Formation Control Strategies for Mini UAVs

In this chapter, we present two methods to solve the unmanned aerial vehicle formation flying problem in which multiple aircraft vehicles try to keep a geometric formation during flight. Aerial refueling is another example of the importance of formation flying. In this case, the two aircraft keep an almost fixed formation between them (the tanker and the receiver). The formation flying control relays in two approaches: the first approach relays in the coordination control scheme, introduced in Chapter 3, and the second approach relays on a nonlinear control based on nested saturation control. We have considered the case of cyclic topology of information flow due to the fact that this communication scheme is easier to implement. Due to the complexity of handling more than one vehicle at a time, formation flying has been experimentally implemented at axial synchronization level of mini-rotorcrafts. Both simulation and experimental results show the effectiveness of the two control approaches – the nonlinear control based on coordination control and the nonlinear control based on nested saturations in the presence of information loss and time delays.

### 7.1. Introduction

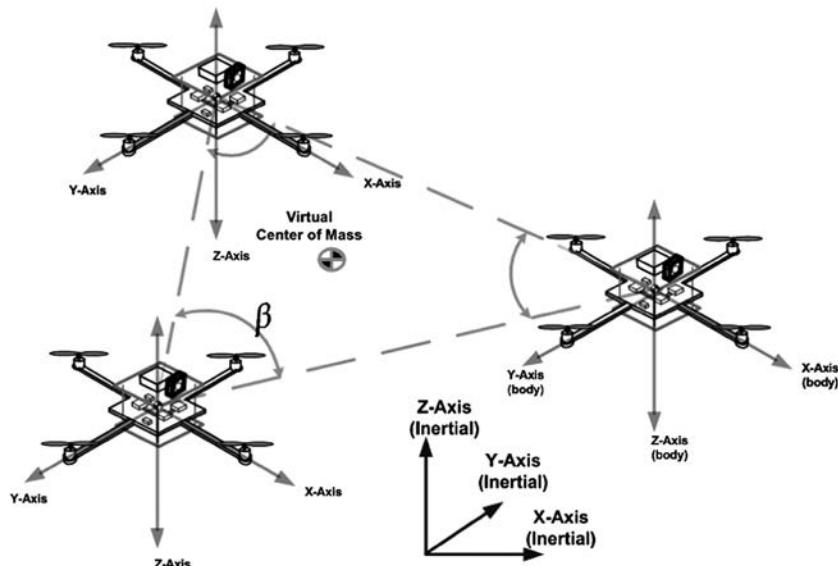
The miniature rotorcraft flight formation control involves the integration of different domains such as rotorcraft control, and coordination control among others. The work reported in the literature is by now quite vast and addresses different approaches for miniature rotorcraft stabilization including linear control

---

Chapter written by J.A. GUERRERO.

[BOU 04, LAR 06, ERG 07], robust control [LAC 03, ISI 03, LAR 06], nonlinear control [LOZ 07, CAS 05b, GUE 10] among others. Different approaches for multiple aircraft flying in formation have been proposed in the literature for the coordination of multiple robot systems. There are mainly three approaches: leader/follower [CHE 06, GUE 10, GIU 00], virtual structure [LEO 01, BEA 01], and behavioral control [ARR 06, BAL 98], which have been discussed in Chapter 1.

Recent advances in unmanned aerial vehicle (UAV) control, graph theory, communication systems, etc., have spurred interest in the UAV flight formation problem. In [GUE 10], a flight formation control based on a forced consensus algorithm is presented. Experimental results on cooperative and coordination control of UAVs have been obtained using the Vicon motion capture system, see [MIC 11] and [OUN 10] among others. Michael *et al.* [MIC 11] present a cooperative control strategy to lift a rigid body using three quadrotors. From a practical point of view, we note that the Vicon system is an indoor localization system which parameterizes the 3D space using a set of infrared cameras. The main advantage of this system is the accuracy (in millimeters) and the processing speed (100–400 Hz). However, the main disadvantages of this system are its price, which is expensive, and the fact that it cannot be used in real outdoor missions. Another important disadvantage of the Vicon system is the centralized nature of the localization system.



**Figure 7.1.** Multiple mini rotorcraft flying in formation

We are interested in the problem of multiple miniature rotorcraft flying in formation, shown in Figure 7.1, using a nonlinear control based on separated

saturations and a single integrator coordination control strategy. The coordination algorithm assumes that there are  $n$ -aerial vehicles which have some kind of information exchange between them. In this approach, every mini rotorcraft is considered as an agent in the multiagent system. We propose a decoupled dynamic coordination. Thus, the lateral, longitudinal, heading, and altitude dynamical subsystems of each mini rotorcraft are considered as agents to be coordinated and to follow a desired reference. To do this, combined with a nonlinear control, we use an algebraic graph theoretical approach to synchronize the behavior of a miniature rotorcraft platoon.

In order to achieve consensus among the members of the platoon, we propose two methods based on single and high-order integrator coordination. It will be shown that the advantage of single integrator coordination strategy is that a member of the platoon will not be affected by perturbations in the altitude of its neighbors. Experimental results have been implemented using two experimental prototypes provided based on a wireless communication system. Position experimental results are still under development due to the lack of accuracy of the commercial mini global positioning system (GPS) modules (small size, small weight) available in the market.

## 7.2. Formation geometry

UAV formation flying implies a geometric formation in either 2D or 3D space. In this section, we present a simple leader-relative formation for three-quadrrotor helicopters. Triangular and line formations are described next.

### 7.2.1. Triangular formation

A triangular formation around a circle of radius  $r$  for the team of three-quadrrotor vehicles, as shown in Figure 7.1, is proposed. Assuming a cyclic information flow topology, the relative position is given by

$$x_1 - x_2 = r \cos(\pi/6), \quad [7.1]$$

$$x_3 - x_1 = -r \cos(\pi/6), \quad [7.2]$$

$$x_2 - x_3 = r \cos(\pi/2), \quad [7.3]$$

$$y_1 - y_2 = r \sin(\pi/6), \quad [7.4]$$

$$y_3 - y_1 = -r \sin(\pi/6), \quad [7.5]$$

$$y_2 - y_3 = 2r \sin(\pi/6). \quad [7.6]$$

Therefore, we can use [7.1]–[7.6] as a relative position reference with respect to each other. Assuming a chain information flow topology, the relative position is given by

$$x_1 - x_2 = \cos(\pi/6), \quad [7.7]$$

$$x_2 - x_3 = \cos(\pi/2), \quad [7.8]$$

$$y_1 - y_2 = \sin(\pi/6), \quad [7.9]$$

$$y_2 - y_3 = 2 \sin(\pi/6). \quad [7.10]$$

Therefore, we can use [7.7]–[7.10] as a relative position reference with respect to each other.

### 7.2.2. Line formation

For the team of three-quadrotor vehicles assuming chain information flow topology, the relative position for a line formation over the  $y$ -axis is given by

$$x_i - x_j = 0, \quad [7.11]$$

$$y_i - y_j = d_{ij}, \quad [7.12]$$

where  $d_{ij}$  is a fixed distance between any two mini rotorcraft. Similarly, the relative position for a line formation over the  $x$ -axis is given by

$$x_i - x_j = d_{ij}, \quad [7.13]$$

$$y_i - y_j = 0. \quad [7.14]$$

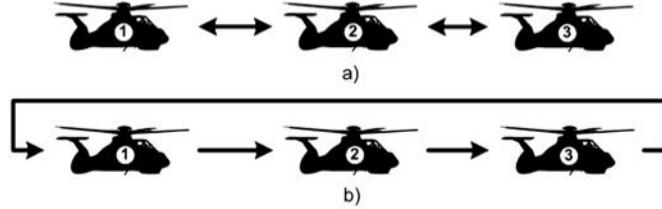
Therefore, we can use [7.11]–[7.14] as a relative position reference with respect to each other. It should be noticed that for a line formation along any axis, the relative position of an agent with respect to its neighbors is the same for any information flow topology.

## 7.3. Communication network

The following is a fundamental result to understand and to achieve multiple autonomous vehicle formation.

**AXIOM 7.1.**— Information sharing through a wireless communication network is a necessary condition for UAV flight formation.

To exemplify this, let us consider the case of three quadrotors with the following information flow topologies: cyclic and chain (see Figure 7.2).



**Figure 7.2.** Chain (a) and cyclic (b) topologies of information flow

**DEFINITION 7.1.–** A chain topology of information flow or cyclic information flow topology is one in which agent  $i$  and agent  $(i + 1)$  mutually exchange their informations, as shown in Figure 7.2(a).

**DEFINITION 7.2.–** A cyclic topology of information flow or cyclic information flow topology is one in which agent  $i$  receives information from agent  $((i + 1) \bmod n)$ , as shown in Figure 7.2(b).

Notice that most of the time, the information flow topology can be defined in such a way to meet the real-life application requirements. In this chapter, we limit ourselves to show the most common information flow topologies used in robotic applications.

#### 7.4. Dynamic model

To obtain the vehicle dynamical model, it will be assumed that it flies over a local area on the Earth. Then, the flat-Earth model equations will be used [STE 03]. The equations representing the kinematic and the moments are written as

$$\dot{\Phi} = \mathbf{H}(\Phi) \omega_{b/e}^b, \quad [7.15]$$

$$\dot{\omega}_{b/e}^b = (\mathbf{J}^b)^{-1} [\mathbf{M}^b - \Omega_{b/e}^b \mathbf{J}^b \omega_{b/e}^b]. \quad [7.16]$$

The vehicle center of mass,  $CM$ , is coincident with the body frame origin,  $F_b$ . The angular velocity in terms of the body system is given by  $\omega_{b/e}^b = [P \ Q \ R]^T$  and its cross-product matrix is denoted by  $\Omega_{b/e}^b$ . The angular velocity in the local inertial system has components  $\dot{\Phi} = [\dot{\phi} \ \dot{\theta} \ \dot{\psi}]^T$ . The matrix of rotation from the inertial frame  $F_e$  to  $F_b$  is denoted by  $\mathbf{C}_{b/n}$ . The matrix of rotation from the inertial frame to the body frame is as follows:

$$\mathbf{C}_{b/n} = \begin{bmatrix} c\theta c\psi & -c\phi s\psi + s\phi c\theta c\psi & s\phi s\psi + c\phi s\theta c\psi \\ c\theta s\phi & c\phi c\psi + s\phi s\theta s\psi & -s\phi c\theta + c\phi s\theta s\psi \\ -s\theta & s\phi c\theta & c\phi c\theta \end{bmatrix}.$$

The set of altitude equations can be obtained using equations [7.15] and [7.16]. The transformation of the components of the angular velocity generated by a sequence of Euler rotations from the body to the local reference system is written as follows:

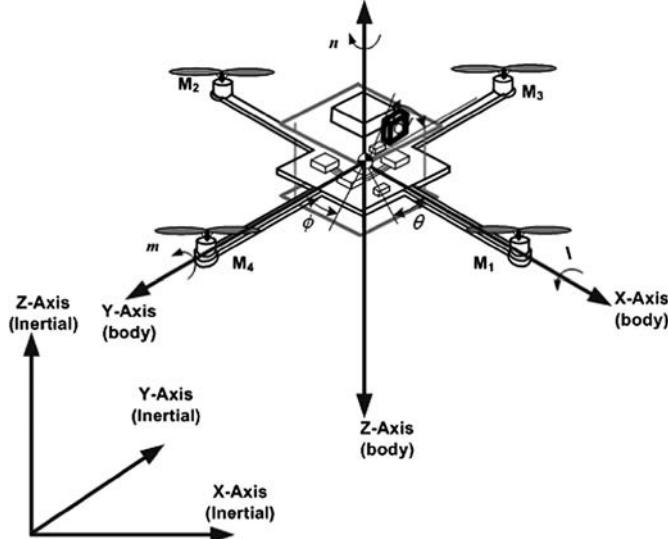
$$\mathbf{H}(\Phi) = \begin{bmatrix} 1 & t\theta s\phi & t\theta c\phi \\ 0 & c\phi & -s\phi \\ 0 & s\phi/c\theta & c\phi/c\theta \end{bmatrix},$$

where  $s$ ,  $c$ , and  $t$  are used to denote the sin, cos, and tan, respectively.

The term  $\mathbf{J}^b$  in [7.16] represents the inertia matrix. Since the quadrotor prototype is symmetrical in the  $xz$ -plane and the  $xy$ -plane, the products of inertia  $J_{xy}$ ,  $J_{yz}$ , and  $J_{xz}$  vanish. Then,  $\mathbf{J}^b$  and its inverse can be written by

$$\mathbf{J}^b = \begin{bmatrix} J_x & 0 & 0 \\ 0 & J_y & 0 \\ 0 & 0 & J_z \end{bmatrix}.$$

The aerodynamics and thrust moments can be denoted by  $\mathbf{M}^b = [\ell \ m \ n]^T$ , and are shown in Figure 7.3.



**Figure 7.3.** Vehicle schematic for vertical flight mode

Then differentiating [7.15], we get

$$\ddot{\Phi} = \dot{\mathbf{H}}(\Phi) \omega_{b/e}^b + \mathbf{H}(\Phi) \dot{\omega}_{b/e}^b. \quad [7.17]$$

Introducing [7.16] into [7.17], we get

$$\ddot{\Phi} = \dot{\mathbf{H}}(\Phi) \omega_{b/e}^b + \mathbf{H}(\Phi) (\mathbf{J}^b)^{-1} \left[ \mathbf{M}^b - \Omega_{b/e}^b \mathbf{J}^b \omega_{b/e}^b \right].$$

It is proposed that

$$\mathbf{M}^b \triangleq \Omega_{b/e}^b \mathbf{J}^b \omega_{b/e}^b + \mathbf{J}^b \mathbf{H}(\Phi)^{-1} \left[ \tilde{\tau} - \dot{\mathbf{H}}(\Phi) \omega_{b/e}^b \right],$$

where  $\tilde{\tau} = [\tilde{\tau}_\phi \quad \tilde{\tau}_\theta \quad \tilde{\tau}_\psi \quad \tilde{\tau}_\phi]^T$ . Then [7.17] can be rewritten as

$$\ddot{\Phi} = \tilde{\tau}.$$

Newton's second law is used to obtain the equations of translational motion in the inertial frame of reference as

$$\ddot{\mathbf{p}}_{CM/T}^n = C_{n/b} \frac{\mathbf{F}^b}{m} + \mathbf{g}^n,$$

where the position of  $CM$  in the North-East-Down (NED) coordinate system with respect to  $F_e$  is given by  $\mathbf{p}_{CM/T}^n = [x \quad y \quad z]^T$ . The aerodynamic and thrust force vector in the body system is represented by  $\mathbf{F}^b = [X \quad Y \quad Z]^T$ . From Figure 7.3, it follows that

$$\mathbf{F}^b = \begin{bmatrix} 0 \\ 0 \\ \mathcal{T} = \sum_{i=1}^4 \mathcal{F}_i \end{bmatrix},$$

where  $\mathcal{F}_i$  is the thrust force produced by the  $i$ th rotor.

Then, the simplified dynamic model for a quadrotor helicopter is given by

$$\ddot{x} = -\mathcal{T} \sin \theta, \tag{7.18}$$

$$\ddot{y} = \mathcal{T} \cos \theta \sin \phi, \tag{7.19}$$

$$\ddot{z} = \mathcal{T} \cos \theta \cos \phi - 1, \tag{7.20}$$

$$\ddot{\phi} = \tilde{\tau}_\phi, \tag{7.21}$$

$$\ddot{\theta} = \tilde{\tau}_\theta, \tag{7.22}$$

$$\ddot{\psi} = \tilde{\tau}_\psi, \tag{7.23}$$

where the constant “−1” is the normalized product of the vehicle mass and the gravitational acceleration.

### 7.5. Formation flying control based on coordination

A coordination algorithm assumes that there are  $n$  agents which have some kind of information exchange between them. In this section, it will be assumed that lateral, longitudinal, altitude, and heading subsystems of each mini rotorcraft are agents to be coordinated and to follow a desired trajectory. To do this, combined with a nonlinear control, we use an algebraic graph theoretical approach to model information exchange where every node in a graph is considered as an agent which can have information exchange with all or several agents. Then, we propose a coordination-based algorithm to solve the 3D formation flying problem. Two geometric formations, line and triangle, are considered in the formation flying problem.

Following [CAS 05a], we define the following control laws:

$$\tau = \frac{-k_z \dot{z} - k_z(z - z^d) + 1}{\cos \phi \cos \theta}, \quad [7.24]$$

$$\tilde{\tau}_\psi = -k_\psi \dot{\psi} - k_\psi(\psi - \psi^d), \quad [7.25]$$

where  $k_z$ ,  $k_\psi$  and  $k_z$ ,  $k_\psi$  are positive constants, and  $z^d$  and  $\psi^d$  represent the desired altitude and heading, respectively. Notice that controls [7.24] and [7.25] stabilize the altitude and heading in a closed-loop system, such that

$$\lim_{t \rightarrow \infty} \|z - z^d\| \rightarrow 0,$$

$$\lim_{t \rightarrow \infty} \|\psi - \psi^d\| \rightarrow 0.$$

Consequently, using [7.24] in [7.18]–[7.23], the lateral dynamic model is given by the following set of equations:

$$\ddot{y} = \tan \phi, \quad [7.26]$$

$$\ddot{\phi} = \tilde{\tau}_\phi. \quad [7.27]$$

Similarly, the longitudinal dynamic model is represented by the following set of equations:

$$\ddot{x} = -\frac{\tan \theta}{\cos \phi}, \quad [7.28]$$

$$\ddot{\theta} = \tilde{\tau}_\theta. \quad [7.29]$$

To stabilize the lateral and longitudinal subsystems of  $n$  aerial vehicles, a consensus algorithm which allows to stabilize the platform in altitude and position is proposed. The cases of having three aerial vehicles with cyclic and chain information flow topology are considered. It is assumed that pitch angle and roll angle are operated

in a neighborhood of the origin, i.e.  $|\phi| < \pi/10$ . Then, the lateral dynamical systems, i.e. equations [7.26] and [7.27], are reduced to

$$\ddot{y}_i = \phi_i, \quad [7.30]$$

$$\ddot{\phi}_i = \tilde{\tau}_{\phi_i}. \quad [7.31]$$

for all  $i \in \mathcal{G}$ . It is clear that systems [7.30] and [7.31] are four integrators in cascade. Then, we propose to use a high-order consensus algorithm to stabilize the rotorcraft. It will be shown that a group of quadrotor vehicles with longitudinal and lateral dynamics given by [7.28] and [7.27] reaches consensus to a desired position while maintaining stable altitude.

Consider the following multiple quadrotor system whose lateral dynamics is governed by:

$$y_i^{(iv)} = \tilde{\tau}_{\phi_i}, \quad [7.32]$$

for all  $i = 1, \dots, n$ .

A first change of variable is proposed

$$\xi_i \triangleq \dot{y}_i + \kappa y_i, \quad [7.33]$$

where  $\kappa$  is a positive constant. Then, the third derivative w.r.t. is

$$\begin{aligned} \xi_i^{(iii)} &= y_i^{(iv)} + \kappa y_i^{(iii)}, \\ \xi_i^{(iii)} &= \tilde{\tau}_{\phi_i} + \kappa y_i^{(iii)}. \end{aligned} \quad [7.34]$$

The control  $u_i$  is defined as

$$\tilde{\tau}_{\phi_i} \triangleq u_i' - \kappa y_i^{(iii)}. \quad [7.35]$$

Equation [7.34] can be rewritten as

$$\xi_i^{(iii)} = u_i'.$$

By following the iterative algorithm, we define a new variable

$$\bar{\xi}_i \triangleq \dot{\xi}_i + \kappa \xi_i. \quad [7.36]$$

The second derivative w.r.t. is given by

$$\begin{aligned} \ddot{\xi}_i &= \xi_i^{(iii)} + \kappa \ddot{\xi}_i, \\ \ddot{\xi}_i &= u_i' + \kappa \ddot{\xi}_i. \end{aligned} \quad [7.37]$$

The control  $u'_i$  is defined as

$$u'_i \triangleq \tilde{u}_i - \kappa \ddot{\xi}_i. \quad [7.38]$$

Equation [7.37] can be rewritten as

$$\ddot{\xi}_i = \tilde{u}_i.$$

By following the iterative algorithm, we define the last change of variable

$$\tilde{\xi}_i \triangleq \dot{\xi}_i + \kappa \bar{\xi}_i. \quad [7.39]$$

The first derivative w.r.t. is given by

$$\begin{aligned} \dot{\tilde{\xi}}_i &= \ddot{\xi}_i + \kappa \dot{\bar{\xi}}_i, \\ \dot{\tilde{\xi}}_i &= \tilde{u}_i + \kappa \dot{\bar{\xi}}_i. \end{aligned} \quad [7.40]$$

The control  $\tilde{u}_i$  is defined as

$$\tilde{u}_i \triangleq \bar{u}_i - \kappa \dot{\bar{\xi}}_i. \quad [7.41]$$

Equation [7.40] can be rewritten as

$$\dot{\tilde{\xi}}_i = \bar{u}_i.$$

We define the controller  $\bar{u}_i$  as

$$\bar{u}_i \triangleq - \sum_{j \in \mathcal{N}_i} (\tilde{\xi}_i - \tilde{\xi}_j). \quad [7.42]$$

Introducing [7.42] into [7.41], we have

$$\tilde{u}_i = - \sum_{j \in \mathcal{N}_i} (\tilde{\xi}_i - \tilde{\xi}_j) - \kappa \dot{\bar{\xi}}_i. \quad [7.43]$$

Introducing [7.43] into [7.38], we have

$$u'_i = - \sum_{j \in \mathcal{N}_i} (\tilde{\xi}_i - \tilde{\xi}_j) - \kappa \dot{\bar{\xi}}_i - \kappa \ddot{\xi}_i. \quad [7.44]$$

Introducing [7.44] into [7.35], we have

$$\tilde{\tau}_{\phi_i} = - \sum_{j \in \mathcal{N}_i} (\tilde{\xi}_i - \tilde{\xi}_j) - \kappa \dot{\bar{\xi}}_i - \kappa \ddot{\xi}_i - \kappa \xi_i^{(iii)}, \quad [7.45]$$

where

$$\begin{aligned}\ddot{\xi}_i &= y_i^{(iii)} + \kappa \ddot{y}_i, \\ \dot{\tilde{\xi}}_i &= y_i^{(iii)} + 2\kappa \ddot{y}_i + \kappa^2 \dot{y}_i.\end{aligned}$$

Then, the lateral control for the  $i$ th quadrotor is given by

$$\tilde{\tau}_{\phi_i} = - \sum_{j \in \mathcal{N}_i} (\tilde{\xi}_i - \tilde{\xi}_j) - 3\kappa y_i^{(iii)} - 3\kappa^2 \ddot{y}_i - \kappa^3 \dot{y}_i. \quad [7.46]$$

Introducing [7.46] into [7.32], we have

$$y_i^{(iv)} = - \sum_{j \in \mathcal{N}_i} (\tilde{\xi}_i - \tilde{\xi}_j) - 3\kappa y_i^{(iii)} - 3\kappa^2 \ddot{y}_i - \kappa^3 \dot{y}_i. \quad [7.47]$$

To show the stability of the lateral system, a positive-definite Lyapunov function is proposed as

$$V = 2(V_1 + \dots + V_N),$$

where  $V_i$  is the storage function for each quadrotor vehicle. Then, using Lemma 2.1,  $\dot{V}$  is given by

$$\dot{V} = -S_i(\tilde{\xi}_i) - K \sum_{i=1}^N \sum_{j \in \mathcal{N}_i} (\tilde{\xi}_i - \tilde{\xi}_j)^T (\tilde{\xi}_i - \tilde{\xi}_j).$$

Noting that  $S_i(\tilde{\xi}_i) = 0 \forall i$  and integrating the above equation we can see that  $(\tilde{\xi}_i - \tilde{\xi}_j) \in \mathcal{L}_2$ . Using [7.39], we have

$$\tilde{\xi}_i - \tilde{\xi}_j = (\dot{\tilde{\xi}}_i - \dot{\tilde{\xi}}_j) + \kappa(\bar{\xi}_i - \bar{\xi}_j). \quad [7.48]$$

Defining  $e_{ij} = \bar{\xi}_i - \bar{\xi}_j$  and differentiating w.r.t., we get  $\dot{e}_{ij} = \dot{\tilde{\xi}}_i - \dot{\tilde{\xi}}_j$ , then [7.48] is rewritten as

$$(\tilde{\xi}_i - \tilde{\xi}_j) = \dot{e}_{ij} + \kappa e_{ij}.$$

Then,  $\dot{e}_{ij} + \kappa e_{ij} \in \mathcal{L}_2$ , which guarantees the exponential convergence of  $e_{ij}$  to the origin. Assuming that the information flow graph is strongly connected then

$$\lim_{t \rightarrow \infty} |\bar{\xi}_i - \bar{\xi}_j| = 0.$$

Since  $(\bar{\xi}_i - \bar{\xi}_j) \in \mathcal{L}_2$ . Using [7.36], we have

$$(\bar{\xi}_i - \bar{\xi}_j) = (\dot{\xi}_i - \dot{\xi}_j) + \kappa(\xi_i - \xi_j). \quad [7.49]$$

Defining  $e'_{ij} = \xi_i - \xi_j$  and differentiating w.r.t., we get  $\dot{e}'_{ij} = \dot{\xi}_i - \dot{\xi}_j$ , then [7.49] is rewritten as

$$(\bar{\xi}_i - \bar{\xi}_j) = \dot{e}'_{ij} + \kappa e'_{ij}.$$

Then  $\dot{e}'_{ij} + \kappa e'_{ij} \in \mathcal{L}_2$ , which guarantees the exponential convergence of  $e'_{ij}$  to the origin. Assuming that the information flow graph is strongly connected then

$$\lim_{t \rightarrow \infty} |\xi_i - \xi_j| = 0.$$

We can see  $(\xi_i - \xi_j) \in \mathcal{L}_2$ . Using [7.33], we have

$$(\xi_i - \xi_j) = (\dot{y}_i - \dot{y}_j) + \kappa(y_i - y_j). \quad [7.50]$$

Defining  $\tilde{e}_{ij} = y_i - y_j$  and differentiating w.r.t., we get  $\dot{\tilde{e}}_{ij} = \dot{y}_i - \dot{y}_j$ , then [7.50] is rewritten as

$$(\xi_i - \xi_j) = \dot{\tilde{e}}_{ij} + \kappa \tilde{e}_{ij}.$$

Then  $\dot{\tilde{e}}_{ij} + \kappa \tilde{e}_{ij} \in \mathcal{L}_2$ , which guarantees the exponential convergence of  $\tilde{e}_{ij}$  to the origin. Assuming that the information flow graph is strongly connected then

$$\lim_{t \rightarrow \infty} |y_i - y_j| = 0.$$

After a time  $T > 0$ ,  $\sum_{j \in \mathcal{N}_i} (\tilde{\xi}_i - \tilde{\xi}_j) \rightarrow 0$ ; therefore, from [7.47] it can be seen that  $y_i^{(iii)}, \ddot{y}_i, \dot{y}_i \rightarrow 0$ . Then, from [7.28] and [7.29], the longitudinal subsystem is reduced to four integrators in cascade. Hence, the following control law is proposed:

$$\tau_{\theta,i} = - \sum_{j \in \mathcal{N}_i} (\tilde{\xi}_i - \tilde{\xi}_j) - 3\kappa x_i^{(iii)} - 3\kappa^2 \ddot{x}_i - \kappa^3 \dot{x}_i. \quad [7.51]$$

To achieve yaw angle synchronization, let us recall the yaw dynamics given by

$$\ddot{\psi}_i = \tilde{\tau}_{\psi_i}. \quad [7.52]$$

By using the same approach described above, we propose a control law such that

$$\tilde{\tau}_{\psi_i} \triangleq - \sum_{j \in \mathcal{N}_i} \left[ (\psi_i - \psi_j) + (\dot{\psi}_i - \dot{\psi}_j) \right] - \kappa \dot{\psi}_i, \quad [7.53]$$

$$\mathcal{T}_i \triangleq - \sum_{j \in \mathcal{N}_i} [(z_i - z_j) + (\dot{z}_i - \dot{z}_j)] - \kappa \dot{z}_i, \quad [7.54]$$

which ensures the consensus agreement in the sense that

$$\begin{aligned} \lim_{t \rightarrow \infty} |\psi_i - \psi_j| &= 0, \\ \lim_{t \rightarrow \infty} |z_i - z_j| &= 0. \end{aligned} \quad [7.55]$$

Control law [7.46], [7.51], [7.53], and [7.54] guarantees the 3D synchronization of all quadrotor vehicles. However, this makes no sense in practical situations. Instead, we propose a formation control described in next section.

### 7.5.1. Formation control

In this section, we propose a leader-relative position consensus (UAV formation) for the multirotor system, i.e. the quadrotor vehicles will converge to a desired position with respect to the leader of the group. We redefine the control  $\bar{u}_i$  as

$$\bar{u}_i = - \sum_{j \in \mathcal{N}_i} (\tilde{\xi}_i - \tilde{\xi}_j) - \sum_{j \in \mathcal{N}_i} \kappa^3 (y_i^{(d)}), \quad [7.56]$$

where the  $y_i^d$  are constants. Then, the formation control law is rewritten as:

$$y_i^{(iv)} = - \sum_{j \in \mathcal{N}_i} (\tilde{\xi}_i - \tilde{\xi}_j) - 3\kappa y_i^{(iii)} - 3\kappa^2 \ddot{y}_i - \kappa^3 \dot{y}_i - \sum_{j \in \mathcal{N}_i} \kappa^3 (y_i^{(d)}).$$

It is important to note that

$$(\tilde{\xi}_i - \tilde{\xi}_j) = y_i^{(iii)} - y_j^{(iii)} + 3\kappa \ddot{y}_i - 3\kappa \ddot{y}_j + 3\kappa^2 \dot{y}_i - 3\kappa^2 \dot{y}_j + \kappa^3 y_i - \kappa^3 y_j.$$

Then, [7.56] can be rewritten as

$$\begin{aligned} \bar{u}_i &= - \sum_{j \in \mathcal{N}_i} (y_i^{(iii)} - y_j^{(iii)}) - \sum_{j \in \mathcal{N}_i} 3\kappa (\ddot{y}_i - \ddot{y}_j) \\ &\quad - \sum_{j \in \mathcal{N}_i} 3\kappa^2 (\dot{y}_i - \dot{y}_j) - \sum_{j \in \mathcal{N}_i} \kappa^3 (y_i - y_j) - \sum_{j \in \mathcal{N}_i} \kappa^3 (y_i^{(d)}), \\ \bar{u}_i &= - \sum_{j \in \mathcal{N}_i} (y_i^{(iii)} - y_j^{(iii)}) - \sum_{j \in \mathcal{N}_i} 3\kappa (\ddot{y}_i - \ddot{y}_j) \\ &\quad - \sum_{j \in \mathcal{N}_i} 3\kappa^2 (\dot{y}_i - \dot{y}_j) - \sum_{j \in \mathcal{N}_i} \kappa^3 (y_i - y_i^{(d)} - y_j). \end{aligned}$$

The control law  $\bar{u}_i$  can be written as

$$\bar{u}_i = - \sum_{j \in \mathcal{N}_i} (\tilde{\xi}_i - \tilde{\xi}_j).$$

Then, [7.47] can be rewritten as

$$y_i^{(iv)} = - \sum_{j \in \mathcal{N}_i} (\tilde{\xi}_i - \tilde{\xi}_j) - 3\kappa y_i^{(iii)} - 3\kappa^2 \ddot{y}_i - \kappa^3 \dot{y}_i,$$

where  $\tilde{\xi}_i = y_i^{(iii)} + 3\kappa \ddot{y}_i + 3\kappa^2 \dot{y}_i + \kappa^3 (y_i - y_i^{(d)})$  and  $\tilde{\xi}_j = y_j^{(iii)} + 3\kappa \ddot{y}_j + 3\kappa^2 \dot{y}_j + \kappa^3 y_j$   $\forall j \in \mathcal{N}_i$ .

The change of variable [7.33] can be rewritten as

$$\xi_i = \dot{y}_i + \kappa(y_i - y_i^{(d)}).$$

Since  $y_i^{(d)}$  is a constant reference position for the  $i$ th vehicle, the third derivative is given by [7.34].

Defining  $\tilde{e}_{ij} = y_i - y_i^{(d)} - y_j$  and differentiating w.r.t., we get  $\dot{\tilde{e}}_{ij} = \dot{y}_i - \dot{y}_j$ , then [7.50] is rewritten as

$$(\xi_i - \xi_j) = \dot{\tilde{e}}_{ij} + \kappa\tilde{e}_{ij}.$$

Then,  $\dot{\tilde{e}}_{ij} + \kappa\tilde{e}_{ij} \in \mathcal{L}_2$ , which guarantees the exponential convergence of  $\tilde{e}_{ij}$  to the constant reference  $y_i^{(d)}$ . Assuming that the information flow graph is strongly connected then

$$\lim_{t \rightarrow \infty} |y_i - y_j| = y_i^{(d)}.$$

Note that the desired leader-relative positions  $x_i^{(d)}$  and  $y_i^{(d)}$  for triangular and line formations are given by equations [7.1]–[7.14].

A high-order coordination-based formation flying implies a synchronization of all states among aerial vehicles. Then, collision avoidance is somehow already included in this control algorithm. However, this control strategy has the disadvantage that a perturbation in the orientation automatically affects the orientation of its neighbors. This lead us to develop a second flight formation control algorithm which is presented next.

## 7.6. Formation flying control based on nested saturations

We recall the simplified quadrotor dynamic model given by equations [7.18]–[7.23]. In this section, we develop a quadrotor formation control algorithm based on nested saturations. Then, we introduce the following definition.

**DEFINITION 7.3.–** Given two positive constants  $L, M$ , with  $L \leq M$ , a function  $\sigma: R \rightarrow R$  is said to be a linear saturation for  $(L, M)$  if it is continuous, non-decreasing function satisfying:

- 1)  $s\sigma(s) > 0$  for all  $s \neq 0$ ;
- 2)  $\sigma(s) = s$  when  $|s| \leq L$ ;
- 3)  $|\sigma(s)| \leq M$  for all  $s \in R$ .

We begin by defining the following control law:

$$\mathcal{T} = \frac{-k_z \dot{z} - k_z(z - z^d) + 1}{\cos \phi \cos \theta}, \quad [7.57]$$

$$\tilde{\tau}_\psi = -k_\psi \dot{\psi} - k_\psi(\psi - \psi^d), \quad [7.58]$$

where  $k_z, k_\psi$  and  $k_z, k_\psi$  are positive constants, and  $z^d$  and  $\psi^d$  represent the desired altitude and heading, respectively. Notice that controls [7.57] and [7.58] stabilize the altitude and heading in a closed-loop system, such that

$$\lim_{t \rightarrow \infty} \|z - z^d\| \rightarrow 0,$$

$$\lim_{t \rightarrow \infty} \|\psi - \psi^d\| \rightarrow 0.$$

In order to introduce the consensus algorithm for the lateral and longitudinal subsystems, we will review the longitudinal control design, which is a simplified dynamic model consisting of four integrators in cascade. Then, we introduce the following variables:

$$\xi_1(x) = \xi_2 - x - 2\dot{x} + \theta, \quad [7.59]$$

$$\xi_2(x) = \xi_3 + \theta - \dot{x}, \quad [7.60]$$

$$\xi_3(x) = \xi_4 + \theta, \quad [7.61]$$

$$\xi_4(x) = \dot{\theta}. \quad [7.62]$$

To simplify the analysis, a recursive methodology is proposed. To do this, it is assumed that

$$\zeta_n = \xi_n(x) + \sigma_{n-1}(\zeta_{n-1}(x)), \quad [7.63]$$

$$\zeta_1 = \xi_1(x), \quad [7.64]$$

and

$$u = -\sigma_n(\zeta_n).$$

Let us define the following positive-definite function:

$$V_n = (1/2)\xi_n^2(x).$$

Differentiating  $V$  with respect to time, we obtain

$$\dot{V}_n = \xi_n(x)\dot{\xi}_n(x).$$

From the fact that  $\dot{\xi}_n(x) = -\sigma_n(\zeta_n)$ , we have

$$\dot{V}_n = \xi_n(x)u = -\xi_n(x)\sigma_n(\zeta_n).$$

Due to equation [7.63], we get

$$\dot{V}_n = -\xi_n(x)\sigma_n(\xi_n(x) + \sigma_{n-1}(\zeta_{n-1}(x))).$$

Using Definition 7.3, we can see that  $M_{n-1} < 0.5L_n$ ; it can be noted that if  $|\xi_n| > 0.5L_n$  then  $\dot{V}_n < 0$ . This means that there exists a time  $T_n$  such that  $|\xi_n| \leq 0.5L_n$  for  $\forall t > T_n$ , which implies that  $|\xi_n + \sigma_{n-1}(\zeta_{n-1}(x))| \leq 0.5L_n + M_{n-1} \leq L_n$ .

When  $n = 1$ , we have the base case of the recursion. This case is treated a little differently, let us propose

$$V_1 = (1/2)\xi_1^2(x).$$

Differentiating  $V$  with respect to time, we obtain

$$\dot{V}_1 = \xi_1(x)\dot{\xi}_1(x).$$

Using [7.59]–[7.62] it is possible to see that  $\dot{\xi}_1(x) = -\sigma_1(\xi_1(x))$ , then we have

$$\dot{V}_1 = -\xi_1(x)\sigma_1(\zeta_1).$$

From equation [7.64], we get

$$\dot{V}_1 = -\xi_1(x)\sigma_1(\xi_1(x)). \quad [7.65]$$

As in the recursive case, it can be noted that if  $|\xi_1(x)| > 0.5L_1$  then  $\dot{V}_1 < 0$ . This means that there exists a time  $T_1$  such that  $|\xi_1(x)| \leq 0.5L_1 \forall t > T_1$ . It is important to note that  $T_n < T_{n-1}$  for all  $n > 2$ .

Since  $\dot{V}_1 < 0$ , then from equations [7.64] and [7.65] it implies that  $\xi_1(x) = \zeta_1 \rightarrow 0$ . It can be noted that starting from  $i = 2$  until  $i = n$ , we have the following set of equations due to the recursion of the method:

$$\dot{V}_2 = -\xi_2(x)\sigma_2(\xi_2(x) + \sigma_1(\zeta_1(x))), \quad [7.66]$$

$$\dot{V}_3 = -\xi_3(x)\sigma_3(\xi_3(x) + \sigma_2(\zeta_2(x))), \quad [7.67]$$

$$\dot{V}_4 = -\xi_4(x)\sigma_4(\xi_4(x) + \sigma_3(\zeta_3(x))). \quad [7.68]$$

The recursion of equation [7.63] leads us to:

$$\zeta_2 = \xi_2(x) + \sigma_1(\zeta_1(x)), \quad [7.69]$$

$$\zeta_3 = \xi_3(x) + \sigma_2(\zeta_2(x)), \quad [7.70]$$

$$\zeta_4 = \xi_4(x) + \sigma_3(\zeta_3(x)). \quad [7.71]$$

After a time  $T_4$ , it can be seen that from [7.66],  $\xi_2 \rightarrow 0$ , [7.69] implies that  $\zeta_2 \rightarrow 0$ , in a recursive way [7.67],  $\xi_3 \rightarrow 0$ , from [7.70],  $\zeta_3 \rightarrow 0$ , and from [7.68],  $\xi_4 \rightarrow 0$ , and from [7.71],  $\zeta_4 \rightarrow 0$ . This means that from [7.62],  $\dot{\theta} \rightarrow 0$ , from [7.61],  $\theta \rightarrow 0$ , from [7.60],  $\dot{x} \rightarrow 0$ , and finally, from [7.59]  $x \rightarrow 0$ .

Due to the fact that the lateral dynamic model is also a four integrators in cascade, we use the same analysis to obtain a control design. Then, the lateral and longitudinal control laws are given by

$$\tilde{\tau}_\theta = -\sigma_4(\dot{\theta} + \sigma_3(\dot{\theta} + \theta + \sigma_2(\dot{\theta} + 2\theta - \dot{x} + \sigma_1(\dot{\theta} + 3\theta - 3\dot{x} - x)))) \quad [7.72]$$

$$\tilde{\tau}_\phi = -\sigma_4(\dot{\phi} + \sigma_3(\dot{\phi} + \phi + \sigma_2(\dot{\phi} + 2\phi + \dot{y} + \sigma_1(\dot{\phi} + 3\phi + 3\dot{y} + y)))) \quad [7.73]$$

Now, we consider the case of having a multiple quadrotor vehicle system with cyclic information flow topology. The longitudinal kinematic model for the multiple quadrotor system is given by

$$\dot{x} = -\mathcal{L}x, \quad [7.74]$$

where  $\mathcal{L}$  is the Laplacian matrix of the information flow graph. However, it must be noted that longitudinal dynamic model is a four-integrator dynamic. We will consider the use of a nonlinear control using a nested saturations strategy. This strategy drives all the states to the origin.

Since the control objective is to force the consensus of a set of quadrotor vehicles to a desired position and heading, we propose the following change of variables:

$$x \triangleq \sum_{j \in \mathcal{N}_i} (x_j - x_i), \quad [7.75]$$

$$y \triangleq \sum_{j \in \mathcal{N}_i} (y_j - y_i), \quad [7.76]$$

$$z \triangleq \sum_{j \in \mathcal{N}_i} (z_j - z_i), \quad [7.77]$$

$$\psi \triangleq \sum_{j \in \mathcal{N}_i} (\psi_j - \psi_i), \quad [7.78]$$

where  $x_i, y_i, z_i, \psi_i, x_j, y_j, z_j$ , and  $\psi_j$  represent the 3D position and heading of the  $i$ th quadrotor and the  $j$ th quadrotor to be coordinated.

**REMARK.–** On the one hand a, multiple quadrotor consensus can be achieved by means of a single integrator consensus algorithm, then [7.75]–[7.78] provide a simple way to solve the coordination problem. On the other hand, we may think of the neighbor's position of a mini rotorcraft as the position reference, and thus, the stability

of every mini rotorcraft is guaranteed using the nonlinear control based on nested saturations.

From the previous control analysis, we have that  $x \rightarrow 0$ ,  $y \rightarrow 0$ ,  $z \rightarrow z^d$ , and  $\psi \rightarrow \psi^d$ , and from [7.75] and [7.76], this implies that

$$\lim_{t \rightarrow \infty} \|x_j - x_i\| = 0,$$

$$\lim_{t \rightarrow \infty} \|y_j - y_i\| = 0.$$

From [7.57], [7.58], [7.77], and [7.78], we have that

$$\lim_{t \rightarrow \infty} \|z_j - z_i\| = z^d,$$

$$\lim_{t \rightarrow \infty} \|\psi_j - \psi_i\| = \psi^d.$$

Therefore, the control laws  $\tilde{\tau}_\theta$ ,  $\tilde{\tau}_\phi$ ,  $\tilde{\tau}_\psi$ , and  $\mathcal{T}$  for the longitudinal, lateral, heading, and altitude subsystems of the  $i$ th quadrotor becomes

$$\begin{aligned} \tilde{\tau}_{\theta_i} = & -\sigma_4 \left( \dot{\theta}_i + \sigma_3 \left( \dot{\theta}_i + \theta_i + \sigma_2 \left( \dot{\theta}_i + 2\theta_i - \dot{x}_i \right. \right. \right. \\ & \left. \left. \left. + \sigma_1 \left( \dot{\theta}_i + 3\theta_i - 3\dot{x}_i + \left( \sum_{j \in \mathcal{N}_i} (x_j - x_i) \right) \right) \right) \right) \right), \end{aligned} \quad [7.79]$$

$$\begin{aligned} \tilde{\tau}_{\phi_i} = & -\sigma_4 \left( \dot{\phi}_i + \sigma_3 \left( \dot{\phi}_i + \phi_i + \sigma_2 \left( \dot{\phi}_i + 2\phi_i + \dot{y}_i \right. \right. \right. \\ & \left. \left. \left. + \sigma_1 \left( \dot{\phi}_i + 3\phi_i + 3\dot{y}_i + \left( \sum_{j \in \mathcal{N}_i} (y_j - y_i) \right) \right) \right) \right) \right), \end{aligned} \quad [7.80]$$

$$\mathcal{T} = \frac{-k_z \dot{z} - k_z \left( \sum_{j \in \mathcal{N}_i} (z_j - z_i) - z^d \right) + 1}{\cos(\phi) \cos(\theta)}, \quad [7.81]$$

$$\tilde{\tau}_\psi = -k_\psi \dot{\psi} - k_\psi \left( \sum_{j \in \mathcal{N}_i} (\psi_j - \psi_i) - \psi^d \right). \quad [7.82]$$

Note that [7.79] and [7.82] enable us to stabilize the coordination for a group of  $n$  mini rotorcrafts. We also notice that, in practice, due to the fact that a coordination to a fixed position implies that every mini rotorcraft will converge to the same position in the 3D space producing the collision of all mini rotorcrafts. In order to solve this problem, a simple leader-relative position control is developed in the next section.

### 7.6.1. Formation control

We propose a relative position consensus (UAV formation) for the multiple quadrotor system, i.e. quadrotor vehicles will converge to a position with respect to

the leader of the group. Using a relative position reference for the flight formation of multiple mini rotorcraft, equations [7.79] and [7.82] are rewritten as

$$\begin{aligned}\tilde{\tau}_{\theta_i} = & -\sigma_4 \left( \dot{\theta}_i + \sigma_3 \left( \dot{\theta}_i + \theta_i + \sigma_2 \left( \dot{\theta}_i + 2\theta_i - \dot{x}_i \right. \right. \right. \\ & \left. \left. \left. + \sigma_1 \left( \dot{\theta}_i + 3\theta_i - 3\dot{x}_i + \left( \sum_{j \in \mathcal{N}_i} (x_j - x_i) - x_i^d \right) \right) \right) \right) \right),\end{aligned}\quad [7.83]$$

$$\begin{aligned}\tilde{\tau}_{\phi_i} = & -\sigma_4 \left( \dot{\phi}_i + \sigma_3 \left( \dot{\phi}_i + \phi_i + \sigma_2 \left( \dot{\phi}_i + 2\phi_i + \dot{y}_i \right. \right. \right. \\ & \left. \left. \left. + \sigma_1 \left( \dot{\phi}_i + 3\phi_i + 3\dot{y}_i + \left( \sum_{j \in \mathcal{N}_i} (y_j - y_i) - y_i^d \right) \right) \right) \right) \right),\end{aligned}\quad [7.84]$$

$$\mathcal{T} = \frac{-k_z \dot{z} - k_z \left( \sum_{j \in \mathcal{N}_i} (z_j - z_i) - z_i^d \right) + 1}{\cos(\phi) \cos(\theta)},\quad [7.85]$$

$$\tilde{\tau}_{\psi} = -k_{\psi} \dot{\psi} - k_{\psi} \left( \sum_{j \in \mathcal{N}_i} (\psi_j - \psi_i) - \psi_i^d \right).\quad [7.86]$$

where  $x_i^d, y_i^d, z_i^d$ , and  $\psi_i^d$  are the desired geometrical 3D position and heading reference with respect to the leader as shown in previous section. Thus, [7.83], [7.84], [7.85], and [7.86] are such that the geometric flight formation of the multiple quadrotor system is guaranteed.

## 7.7. Trajectory-tracking control

Now, we will consider the case of trajectory tracking of a multiple vehicle system. It is assumed that the leader of the group is always vehicle 1. Then, [7.74] is rewritten as (then,  $\bar{u}_i$  is rewritten as)

$$\dot{\mathbf{x}} = -\mathcal{L}\mathbf{x} + \mathbf{b}u_l,\quad [7.87]$$

where  $\mathbf{b}^T = [1 \ 0 \ \dots \ 0]$  and  $u_l$  is the input given to the leader. Define  $x_{CM} = 1/N \sum_{i=1}^N x_i$ , where  $N$  is the number of aerial vehicles in the formation. Let  $x_{CM}^d$  be the desired value for  $x_{CM}$ . Assume for simplicity that aerial vehicle 1 is the leader, i.e.  $\mathbf{c} = \mathbf{b}^T = [1 \ 0 \ \dots \ 0]$  and that the control law is

$$u_l(x) = Nk\sigma(x_{CM}^d - x_{CM}),\quad [7.88]$$

where  $k$  is a positive gain. Note that  $x_{CM}$  may not be directly measurable for the leader (vehicle 1). We assume the system is observable from the input and output of

the leader. The state can, therefore, be observed from the input and output of vehicle 1. Introducing [7.88] into [7.87], we get

$$\begin{aligned}\dot{x}_{CM} &= k\sigma(x_{CM}^d - x_{CM}), \\ \mathbf{v}_i^T \mathbf{x} &= -\lambda_i(\mathbf{v}_i^T \mathbf{x}) + \mathbf{v}_i^T \mathbf{b} u_1, \quad i = 2, \dots, N.\end{aligned}$$

The modes in the last equation above are all stable. When  $u_1 = 0$ , these modes converge to zero which means that  $(x_i - x_j) \rightarrow 0$ , for  $i \neq j$ . This property is obtained by using the coordinating control algorithm that leads the position dynamics to [7.74]. These modes are uncontrollable when  $\mathbf{v}_i^T \mathbf{b} = 0$ . There is a tradeoff in the choice of gain  $k$  in [7.88]. For smaller values of  $k$ , the speed of convergence of  $x_{CM}$  is slower, but the transient in the errors  $(x_i - x_j)$ , for  $i \neq j$ , will be smaller.

Note that for the coordination-based formation, [7.88] takes the form

$$u_l(\tilde{\xi}) = Nk\sigma(\tilde{\xi}_{CM}^d - \tilde{\xi}_{CM}). \quad [7.89]$$

Then, the trajectory tracking control law is such that  $\tilde{\xi}_{CM} \rightarrow \tilde{\xi}_{CM}^d$  as  $t \rightarrow \infty$ , which implies that  $\tilde{\xi} \rightarrow \tilde{\xi}_{CM}^d/\kappa$ , which, in turn, implies that  $\xi \rightarrow \tilde{\xi}_{CM}^d/\kappa^2$  and  $x \rightarrow \tilde{\xi}_{CM}^d/\kappa^3$  and  $\tilde{\xi}_{CM}^d = \kappa^3(x_{CM}^{(d)})$ , which implies that  $|x_i - x_i^d| \rightarrow x_i^d$

Again, we can use the same control strategy for the lateral dynamics by replacing  $x$  position and  $\theta$  angle by  $y$  position and  $\phi$  angle.

The trajectory tracking for the leader of the group using high-order consensus is given by

$$\tilde{\tau}_{\phi_i} = - \sum_{j \in \mathcal{N}_i} (\tilde{\xi}(y)_i - \tilde{\xi}(y)_j) - 3\kappa\dot{\phi} - 3\kappa^2\phi_i - \kappa^3\dot{y}_i - \sum_{j \in \mathcal{N}_i} \kappa^3(y_i^{(d)}) - u_l(y), \quad [7.90]$$

$$\tilde{\tau}_{\theta_i} = - \sum_{j \in \mathcal{N}_i} (\tilde{\xi}(x)_i - \tilde{\xi}(x)_j) - 3\kappa\dot{\theta} - 3\kappa^2\theta_i - \kappa^3\dot{x}_i - \sum_{j \in \mathcal{N}_i} \kappa^3(x_i^{(d)}) - u_l(x). \quad [7.91]$$

The trajectory tracking for the leader of the group using nested saturations is given by

$$\begin{aligned}\tilde{\tau}_{\theta_i} &= -\sigma_4 \left( \dot{\theta}_i + \sigma_3 \left( \dot{\theta}_i + \theta_i + \sigma_2 \left( \dot{\theta}_i + 2\theta_i - \dot{x}_i \right. \right. \right. \\ &\quad \left. \left. \left. + \sigma_1 \left( \dot{\theta}_i + 3\theta_i - 3\dot{x}_i + \left( \sum_{j \in \mathcal{N}_i} (x_j - x_i) - x_i^d - u_l(x) \right) \right) \right) \right) \right),\end{aligned} \quad [7.92]$$

$$\begin{aligned}\tilde{\tau}_{\phi_i} = & -\sigma_4 \left( \dot{\phi}_i + \sigma_3 \left( \dot{\phi}_i + \phi_i + \sigma_2 \left( \dot{\phi}_i + 2\phi_i + \dot{y}_i \right. \right. \right. \\ & \left. \left. \left. + \sigma_1 \left( \dot{\phi}_i + 3\phi_i + 3\dot{y}_i + \left( \sum_{j \in \mathcal{N}_i} (y_j - y_i) - y_i^d - u_l(y) \right) \right) \right) \right) \right). [7.93]\end{aligned}$$

### 7.7.1. Time-varying reference tracking

It has been shown in previous sections that control laws [7.90]–[7.93] ensure the convergence of the center of mass of a multivehicle system to a constant reference given to the leader. However, when the reference is varying in time there is a small bias in agents coordination.

In this section, we consider the case of multiagent trajectory tracking of a time-varying reference. We will prove that a multivehicle system converges to the position time-varying reference given only to the leader. Again, we are interested in the chain and cyclic topologies of information flow. Let us consider the double-integrator multivehicle system of the form

$$\ddot{x}_i = u_i.$$

We define a change of variable

$$\begin{aligned}\xi_i &\triangleq \dot{x}_i + \kappa x_i, \\ \xi_{CM}^d &\triangleq \dot{x}_{CM}^d + \kappa x_{CM}^d,\end{aligned}[7.94]$$

where  $\kappa$  is a positive constant. Control input  $u_i$  is defined as

$$u_i \triangleq \bar{u}_i - \kappa \dot{x}_i. [7.95]$$

Differentiating [7.94] and using control [7.95], we obtain

$$\dot{\xi}_i = \bar{u}_i.$$

Define the coordinating control  $\bar{u}_i$  as

$$\bar{u}_i \triangleq -\sum_{j \in \mathcal{N}_i} (\xi_i - \xi_j) + b_i \tilde{u}_i.$$

Thus, the following multiagent system is obtained

$$\dot{\xi} = -\mathcal{L}\xi + \mathbf{b}\tilde{u}, [7.96]$$

where  $\mathcal{L}$  is the Laplacian matrix and the control law

$$\tilde{u}_l \triangleq \tilde{u}_{CM}, [7.97]$$

$$\tilde{u}_i \triangleq \dot{\xi}_j \text{ for some } j \in \mathcal{N}_i, [7.98]$$

where  $u_l$  is the input given to the leader,  $u_i$  is the input given to the  $i$ th vehicle and  $\tilde{u}_{CM} = k_{CM} \sigma_{M_{CM}} (\xi_{CM} - \xi_{CM}^d)$ .

### 7.7.1.1. Chain topology

Let us consider the case of three agents with chain topology of information flow with agent 1 acting as the leader of the group

$$\dot{\xi}_1 = (\xi_2 - \xi_1) + \tilde{u}_{CM}, \quad [7.99]$$

$$\dot{\xi}_2 = (\xi_1 - \xi_2) + (\xi_3 - \xi_2) + \dot{\xi}_1, \quad [7.100]$$

$$\dot{\xi}_3 = (\xi_2 - \xi_3) + \dot{\xi}_2. \quad [7.101]$$

This system can also be represented as

$$\begin{bmatrix} \dot{\xi}_1 \\ \dot{\xi}_2 \\ \dot{\xi}_3 \end{bmatrix} = \begin{bmatrix} -1 & 1 & 0 \\ 1 & -2 & 1 \\ 0 & 1 & -1 \end{bmatrix} \begin{bmatrix} \xi_1 \\ \xi_2 \\ \xi_3 \end{bmatrix} + \begin{bmatrix} \tilde{u}_{CM} \\ \dot{\xi}_1 \\ \dot{\xi}_2 \end{bmatrix}. \quad [7.102]$$

Rewriting [7.101], we get

$$(\dot{\xi}_3 - \dot{\xi}_2) = -(\xi_3 - \xi_2), \quad [7.103]$$

then, from the above, this implies that  $(\xi_3 - \xi_2) \rightarrow 0$ .

From [7.100], it follows

$$(\dot{\xi}_2 - \dot{\xi}_1) = (\xi_1 - \xi_2) + (\xi_3 - \xi_2). \quad [7.104]$$

Since  $(\xi_3 - \xi_2) \rightarrow 0$ , then [7.104] reduces to  $(\dot{\xi}_2 - \dot{\xi}_1) = -(\xi_2 - \xi_1)$ , which also implies that  $(\xi_2 - \xi_1) \rightarrow 0$ .

Premultiplying [7.102] by 1 eigenvector, we get

$$\begin{aligned} \dot{\xi}_1 + \dot{\xi}_2 + \dot{\xi}_3 &= \tilde{u}_{CM} + \dot{\xi}_1 + \dot{\xi}_2, \\ \dot{\xi}_3 &= \tilde{u}_{CM}. \end{aligned} \quad [7.105]$$

Now, define

$$\tilde{u}_{CM} \triangleq -(\xi_{CM} - \xi_{CM}^d) + \dot{\xi}_{CM}^d, \quad [7.106]$$

then, introducing [7.106] into [7.105], it follows that  $(\dot{\xi}_3 - \dot{\xi}_{CM}^d) = -(\xi_{CM} - \xi_{CM}^d)$ , which implies that  $(\xi_3 - \xi_{CM}^d) \rightarrow 0$ . Assume that,  $\xi_{CM}^d = x_{CM}^d$  and  $\dot{x}_{CM}^d = 0$ , then  $(\xi_3 - x_{CM}^d) \rightarrow 0$  implies that  $(x_3 - x_{CM}^d) \rightarrow 0$ . Note that  $(x_i - x_j) \rightarrow 0$  and  $(x_3 - x_{CM}^d) \rightarrow 0$  then  $(x_i - x_{CM}^d) \rightarrow 0$  for all  $i = 1, 2, 3$ .

**LEMMA 7.1.-** Consider a multiagent system of the form [7.96] and control laws [7.97] and [7.98], then,  $\dot{x}_{CM} \rightarrow \dot{x}_{CM}^d$  as  $t \rightarrow \infty$ , moreover  $(\dot{x}_i - \dot{x}_{CM}^d) \rightarrow 0$ .

### 7.7.1.2. Cyclic topology

Let us consider the following equations that represent the case of three agents with chain topology of information flow and agent 1 acting as the leader of the group:

$$\dot{\xi}_1 = (\xi_2 - \xi_1) + \tilde{u}_{CM}, \quad [7.107]$$

$$\dot{\xi}_2 = (\xi_3 - \xi_2) + \dot{\xi}_3, \quad [7.108]$$

$$\dot{\xi}_3 = (\xi_1 - \xi_3) + \dot{\xi}_1, \quad [7.109]$$

or

$$\begin{bmatrix} \dot{\xi}_1 \\ \dot{\xi}_2 \\ \dot{\xi}_3 \end{bmatrix} = \begin{bmatrix} -1 & 1 & 0 \\ 0 & -1 & 1 \\ 1 & 0 & -1 \end{bmatrix} \begin{bmatrix} \xi_1 \\ \xi_2 \\ \xi_3 \end{bmatrix} + \begin{bmatrix} \tilde{u}_{CM} \\ \dot{\xi}_1 \\ \dot{\xi}_2 \end{bmatrix}. \quad [7.110]$$

From [7.109], we have

$$(\dot{\xi}_3 - \dot{\xi}_1) = -(\xi_3 - \xi_1), \quad [7.111]$$

which implies that  $(\xi_3 - \xi_1) \rightarrow 0$ .

Rewriting [7.108], we get

$$(\dot{\xi}_3 - \dot{\xi}_2) = -(\xi_3 - \xi_2), \quad [7.112]$$

then  $(\xi_3 - \xi_2) \rightarrow 0$ .

Premultiplying [7.110] by 1 eigenvector, we get

$$\dot{\xi}_1 + \dot{\xi}_2 + \dot{\xi}_3 = \tilde{u}_{CM} + \dot{\xi}_1 + \dot{\xi}_3, \quad [7.113]$$

$$\dot{\xi}_2 = \tilde{u}_{CM}. \quad [7.114]$$

Now, define

$$\tilde{u}_{CM} \triangleq -(\xi_{CM} - \xi_{CM}^d) + \dot{\xi}_{CM}^d, \quad [7.115]$$

thus, introducing [7.115] into [7.114] it follows that

$$(\dot{\xi}_2 - \dot{\xi}_{CM}^d) = -(\xi_{CM} - \xi_{CM}^d),$$

which implies that  $(\xi_2 - \xi_{CM}^d) \rightarrow 0$ . Assume that  $\xi_{CM}^d = x_{CM}^d$  and  $\dot{x}_{CM}^d = 0$ , then  $(\xi_2 - x_{CM}^d) \rightarrow 0$  implies that  $(x_2 - x_{CM}^d) \rightarrow 0$ . Due that  $(x_i - x_j) \rightarrow 0$  and  $(x_2 - x_{CM}^d) \rightarrow 0$ , then  $(x_i - x_{CM}^d) \rightarrow 0$  for all  $i = 1, 2, 3$ .

**LEMMA 7.2.-** Consider a multiagent system of the form [7.96] with coordinating control laws [7.97] and [7.98]. Then,  $\dot{x}_{CM} \rightarrow \dot{x}_{CM}^d$  as  $t \rightarrow \infty$ , moreover  $(\dot{x}_i - \dot{x}_{CM}^d) \rightarrow 0$ .

## 7.8. Simulation results

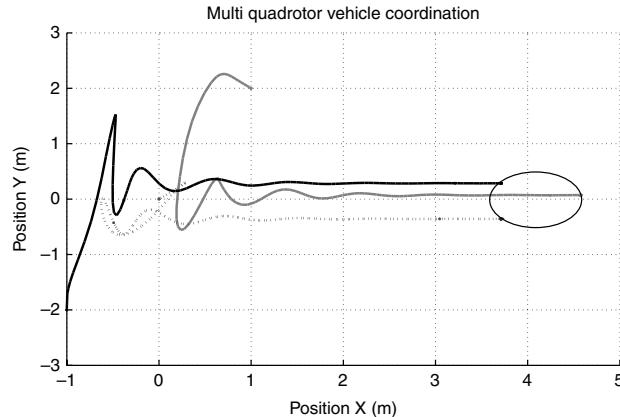
To illustrate the proposed control algorithms, this section presents the simulation results concerning the geometric formation and tracking of a time-varying reference for a quadrotor platoon. We consider three quadrotors evolving in the 3D space. Extensive simulations were run on a small platoon of three rotorcraft considering the 6-DOF nonlinear dynamical model. We have developed a simulation model using Matlab Simulink<sup>TM</sup> in which cyclic and chain topologies of information flow have been implemented.

### 7.8.1. High-order consensus-based formation

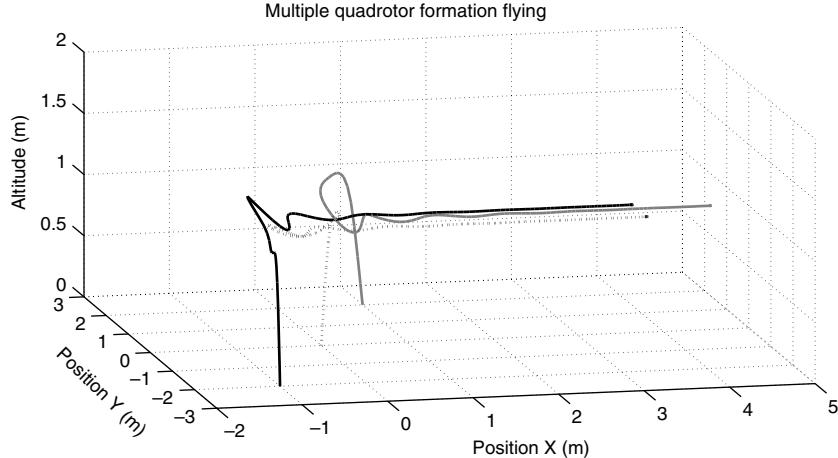
We have used different initial altitude and position values for each quadrotor. Regardless of the initial conditions, the proposed control law effectively stabilize the formation of multiple quadrotors. The initial conditions for inertial position and velocity are

$$\begin{aligned} [x_1, y_1, z_1] &= [1, 2, 0] \text{ m}; \quad [\dot{x}_1, \dot{y}_1, \dot{z}_1] = [-0.1, -0.1, 0.2] \text{ m/s} \\ [x_2, y_2, z_2] &= [0, 0, 0] \text{ m}; \quad [\dot{x}_2, \dot{y}_2, \dot{z}_2] = [-0.1, -0.2, 0.3] \text{ m/s} \\ [x_3, y_3, z_3] &= [-1, -2, 0] \text{ m}; \quad [\dot{x}_3, \dot{y}_3, \dot{z}_3] = [0.2, 0.3, -0.5] \text{ m/s} \end{aligned}$$

It is clear that the nonlinear coordinated control strategy can be used to synchronize 3D position of a group of quadrotor helicopters. Thus, using control inputs [7.46], [7.51], [7.53], and [7.54] on the quadrotor dynamical systems [7.18]–[7.23] in simulation we get the results shown in Figures 7.4 and 7.5.



**Figure 7.4.** Formation flying of multiple quadrotors using coordination control (top view)



**Figure 7.5.** Formation flying of multiple quadrotors using coordination control

### 7.8.2. Nested saturations based formation

Similarly to the first formation control approach, we have used different initial altitude and position values for each quadrotor. Regardless of the initial conditions, the proposed control law effectively stabilizes the formation of multiple quadrotors. The initial conditions for inertial position and velocity are

$$\begin{aligned} [x_1, y_1, z_1] &= [1, 2, 0] \text{ m}; \quad [\dot{x}_1, \dot{y}_1, \dot{z}_1] = [-0.1, -0.1, 0.2] \text{ m/s} \\ [x_2, y_2, z_2] &= [0, 0, 0] \text{ m}; \quad [\dot{x}_2, \dot{y}_2, \dot{z}_2] = [-0.1, -0.2, 0.3] \text{ m/s} \\ [x_3, y_3, z_3] &= [-1, -2, 0] \text{ m}; \quad [\dot{x}_3, \dot{y}_3, \dot{z}_3] = [0.2, 0.3, -0.5] \text{ m/s} \end{aligned}$$

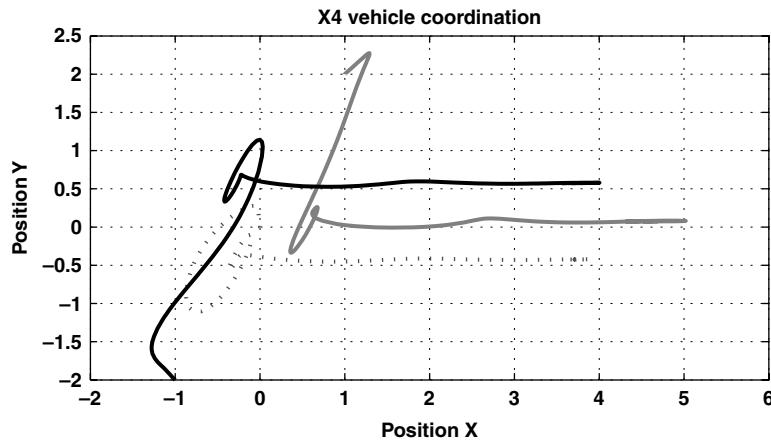
The results of simulation show that the proposed nonlinear control strategy can be used to achieve a geometric formation as well as formation flying of multiple quadrotor. Thus, using control inputs [7.83], [7.84], [7.85], and [7.86] on the mini rotorcraft acting as followers and [7.81], [7.82], [7.92], and [7.93] on the mini rotorcraft acting as leader, on the 6-DOF nonlinear dynamical models [7.18]–[7.23] in simulation, we get the results shown in Figures 7.6 and 7.7.

### 7.8.3. Time-varying tracking

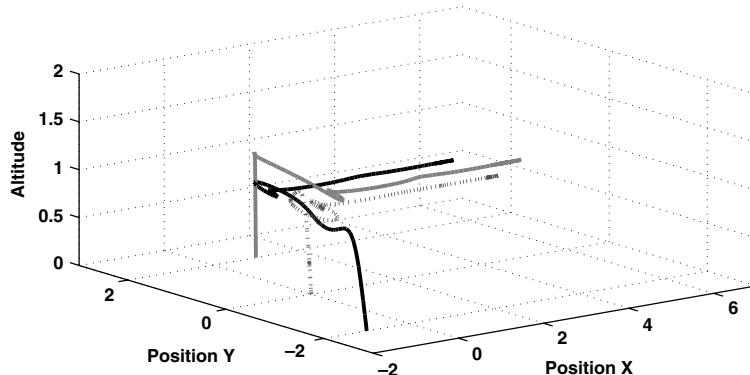
Initially, the platoon should converge to a triangular formation with center of mass at the (0, 0) position in the 2D plane. Once the platoon has converged to the triangular formation, the order to displace along the  $x$ -axis is given. After a short time, the constant reference is changed to a trigonometric function sinus while maintaining the

heading in the same direction all the time. Finally, the reference is changed again to a constant reference. The initial conditions for inertial position and velocity are:

$$\begin{aligned}[x_1, y_1, z_1] &= [0, 0.5, 0] \text{ m}; \quad [\dot{x}_1, \dot{y}_1, \dot{z}_1] = [0, 0, 0] \text{ m/s} \\ [x_2, y_2, z_2] &= [-2, 0.1, 0] \text{ m}; \quad [\dot{x}_2, \dot{y}_2, \dot{z}_2] = [0, 0, 0] \text{ m/s} \\ [x_3, y_3, z_3] &= [-2, -0.5, 0] \text{ m}; \quad [\dot{x}_3, \dot{y}_3, \dot{z}_3] = [0, 0, 0] \text{ m/s}\end{aligned}$$



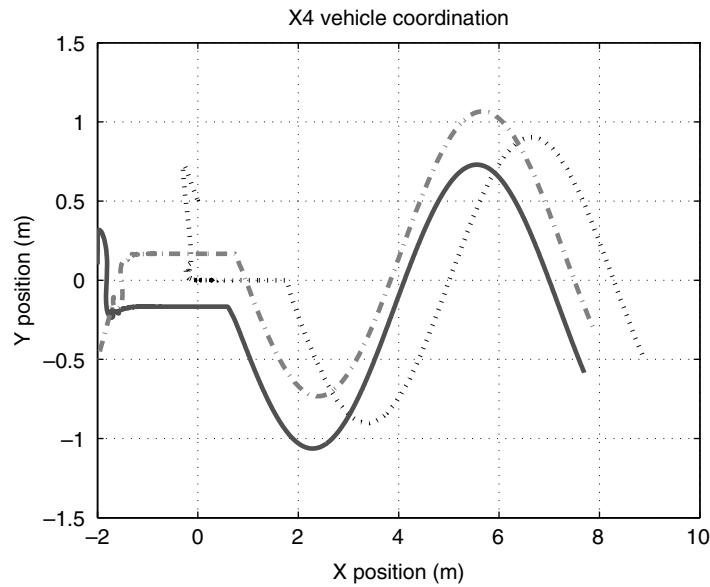
**Figure 7.6.** Formation flying of multiple quadrotors using nested saturations (top view)



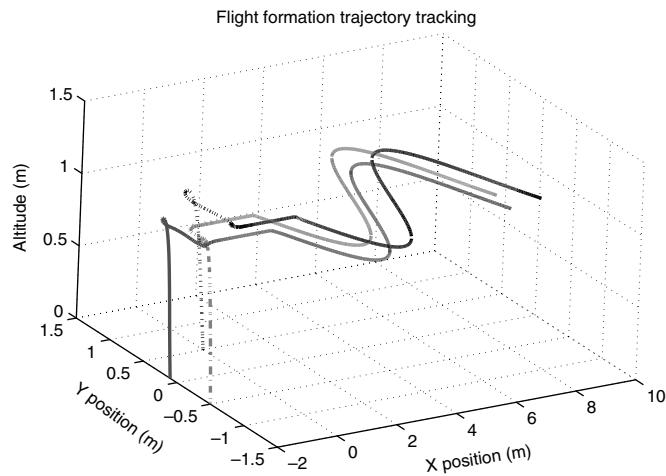
**Figure 7.7.** Formation flying of multiple quadrotors using nested saturations

The simulation results show that the proposed nonlinear control strategy can be used to achieve a geometric formation as well as tracking a time-varying reference

for a group of multiple mini rotorcrafts. Thus, using control inputs [7.83], [7.84], [7.85], and [7.86] on the mini rotorcraft acting as followers and [7.85], [7.86], [7.92], and [7.93] with  $u_l[x]$  and  $u_l[y]$  defined has in section 7.7.1, on the quadrotor acting as leader, on the 6-DOF nonlinear dynamical model in simulation, we get the result shown in Figures 7.8 and 7.9.



**Figure 7.8.** Trajectory tracking with cyclic topology



**Figure 7.9.** Trajectory tracking (3D view)

### 7.9. Conclusions

We have presented two nonlinear control approaches to formation flying of quadrotor vehicles. The first approach uses a coordination control algorithm that synchronizes every state to the origin, the position which synchronizes to a geometric formation reference. The second approach considers a position coordination to synchronize and keep the desired formation.

In the proposed formation flying control approaches, tracking of the center of mass of the quadrotors formation has been achieved by using a state feedback control applied to the leader. It is assumed that the leader has full information access from every agent. However, when the leader is not assumed to have direct information from all the agents, the state is observed from the input and output of the leader. This approach requires observability and controllability (or at least detectability and stabilizability) of the agents' network from the leader input and output.

The flight formation algorithms proposed in this chapter provide a promising method for formation flying and cooperative control of multiple quadrotor vehicles even in the presence of some physical limitations, information loss, and time delay. Despite the success in experimental tests, there are several limitations to think about. An important limitation is that the GPS may not be accurate enough to stabilize the formation flying of quadrotors. In theory, commercial communications systems are limited to 100 m but their real limits are much shorter due to environmental conditions.

### 7.10. Bibliography

- [ARR 06] ARRICIELLO F., CHIAVERINI S., FOSSEN T., "Formation control of marine vessels using the null-space-based behavioral control", *LNCIS 336 Group Coordination and Cooperative Control*, Springer-Verlag, pp. 1–19, 2006.
- [BAL 98] BALCH T., ARKIN A., "Behavior-based formation control for multirobot teams", *IEEE Trans. Robotics and Automation*, vol. 14, no. 6, pp. 926–939, 1998.
- [BEA 01] BEARD R., LAWTON J., HADAEGH F., "A coordination architecture for spacecraft formation control", *IEEE Transactions on Control Systems Technology*, vol. 9, no. 6, pp. 777–790, 2001.
- [BOU 04] BOUABDALLAH S., N. A., SIEGWART R., "PID vs LQ control techniques applied to an indoor micro quadrotor", *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, Sendai, Japan, pp. 2451–2456, 2004.
- [CAS 05a] CASTILLO P., LOZANO R., DZUL A., *Modelling and Control of Mini-Flying Machines*, Springer-Verlag, 2005.
- [CAS 05b] CASTILLO P., LOZANO R., DZUL A., "Stabilization of a mini rotorcraft with four rotors", *IEEE Control Systems*, vol. 25, no. 6, pp. 45–55, December 2005.

- [CHE 06] CHEN X., SERRANI A., “ISS-based robust leader/follower trailing control”, *LNCIS 336 Group Coordination and Cooperative Control*, Springer-Verlag, pp. 75–91, 2006.
- [ERG 07] ERGINER B., ALTUG E., “Modeling and PD control of a quadrotor VTOL vehicle”, *2007 IEEE Intelligent Vehicles Symposium*, Istanbul, pp. 894–899, June 2007.
- [GIU 00] GIULIETTI F., POLLINI L., INNOCENTI M., “Autonomous formation flight”, *IEEE Control Systems*, vol. 20, no. 6, pp. 34–44, December 2000.
- [GUE 10] GUERRERO J., FANTONI I., SALAZAR S., LOZANO R., “Flight formation of multiple mini rotorcraft via coordination control”, *2010 IEEE International Conference on Robotics and Automation (ICRA)*, Anchorage, pp. 620–625, May 2010.
- [ISI 03] ISIDORI A., MARCONI L., SERRANI A., “Robust nonlinear motion control of a helicopter”, *IEEE Transactions on Automatic Control*, vol. 48, no. 03, pp. 413–426, 2003.
- [LAC 03] LA CIVITA M., PAPAGEORGIOU G., MESSNER W., KANADE T., “Design and flight testing of a gain-scheduled  $H_\infty$  loop shaping controller for wide-envelope flight of a robotic helicopter”, *Proceedings of the 2003 American Control Conference*, Denver, pp. 4195–4200, 2003.
- [LAR 06] LARA D., SANCHEZ A., LOZANO R., CASTILLO P., “Real-time embedded control system for VTOL aircrafts: Application to stabilize a quad-rotor helicopter”, *Computer Aided Control System Design, 2006 IEEE International Conference on Control Applications, 2006 IEEE International Symposium on Intelligent Control, 2006 IEEE*, Munich, pp. 2553–2558, October 2006.
- [LEO 01] LEONARD N., FIORELLI E., “Virtual leaders, artificial potentials and coordinated control of groups”, *Proceedings of the 40th IEEE Conference on Decision and Control*, Orlando, pp. 2968–2973, 2001.
- [LOZ 07] LOZANO R., *Objets Volants Miniatures: Modélisation et Commande Embarquée*, Hermès-Lavoisier, 2007.
- [MIC 11] MICHAEL N., FINK J., KUMAR V., “Cooperative manipulation and transportation with aerial robots”, *Autonomous Robots*, vol. 30, pp. 73–86, 2011.
- [OUN 10] OUNG R., BOURGAULT F., DONOVAN M., D’ANDREA R., “The distributed flight array”, *2010 IEEE International Conference on Robotics and Automation (ICRA)*, Anchorage, pp. 601–607, May 2010.
- [STE 03] STEVENS B., LEWIS F., *Aircraft Control and Simulation*, John Wiley and Sons, 2003.



# Chapter 8

## Formation Based on Potential Functions

### 8.1. Introduction

The basic concept of the potential field method consists of filling the robot workspace with an artificial potential field, in which the robot is attracted to its goal position (the point of minimal potential energy) and it is repulsed away from the obstacles. This method is particularly attractive because of its mathematical elegance and simplicity [GE 00]. This method is, at least in concept, non-hierarchical. The potential functions do not depend on the vehicle's model, they just give the force vectors, then a control law will be responsible for moving each vehicle according to its computed forces.

A system of vehicles in formation, even when each vehicle has a tractable model and interacts with their neighbors in a simple and predictable way, the resulting systems present a rich and complex behavior when viewed as a whole [D'AN 03]. Interconnections, even though weak, can cause parameter drift and, therefore, instability of the scheme [IOA 86]. For our stability analysis, we consider the formation as an interconnected system of similar vehicles which, through potential forces, interact with their neighbors. In our case, the interconnection is well defined. We study this interconnected system as linear, where the stability analysis is performed through pole location which leads us to find a formation gain of the proposed controller.

---

Chapter written by L. GARCÍA and A. DZUL.

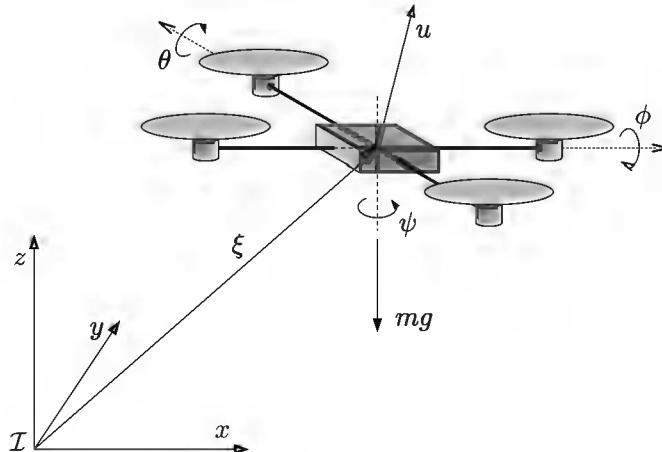
In this chapter, we address a two-dimensional (2D) formation control, using simple potential functions that generate the desired forces and a nested saturation controller to move the vehicles to their goal positions.

### 8.2. Dynamical model

In this section, the model of the quadrotor aircraft, using the Euler–Lagrange approach, is presented. The generalized coordinates for the quadrotor are the following:

$$\mathbf{q} = (x, y, z, \phi, \theta, \psi) \in \mathbb{R}^6$$

where  $\xi = (x, y, z) \in \mathbb{R}^3$  denotes the position of the center of mass of the rotorcraft, relative to the inertial frame  $\mathcal{I}$ , and  $\eta = (\phi, \theta, \psi) \in \mathbb{R}^3$  represents the Euler angles (roll, pitch, and yaw, respectively) that describe the rotorcraft orientation (Figure 8.1).



**Figure 8.1.** Quadrotor rotorcraft

Based on [SÁN 08], we have used the next dynamical model:

$$m\ddot{x} = u(\cos \psi \sin \theta \cos \phi + \sin \psi \sin \phi) \quad [8.1]$$

$$m\ddot{y} = u(\sin \psi \sin \theta \cos \phi - \cos \psi \sin \phi) \quad [8.2]$$

$$m\ddot{z} = u \cos \theta \cos \phi - mg \quad [8.3]$$

$$\ddot{\psi} = \tau_\psi \quad [8.4]$$

$$\ddot{\theta} = \tau_\theta \quad [8.5]$$

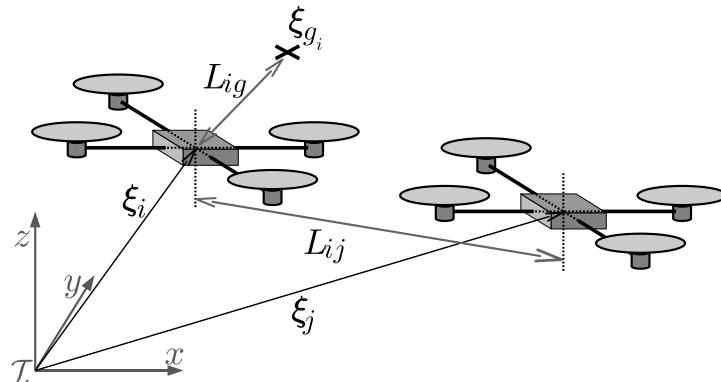
$$\ddot{\phi} = \tau_\phi \quad [8.6]$$

where the inertia terms are implicit in the yaw, pitch, and roll torques ( $\tau_\psi$ ,  $\tau_\theta$ , and  $\tau_\phi$ , respectively) [CAS 04].  $u$  denotes the total thrust,  $m$  represents the mass of the vehicle, and  $g$  is the gravitational acceleration.

### 8.3. Formation control

The objective consists of achieving a desired formation and, at the same time, to be able to guarantee both collision avoidance with another quadrotor and obstacle avoidance. To solve the previous problem, we will use artificial potential functions [GAR 11]. We start by assuming that the altitude  $z$  is maintained constant, such that the formation is going to be realized in a 2D environment. With this assumption, the translational vector expressing the position of the rotorcraft  $i$  is represented by:

$$\xi_i = \begin{bmatrix} x_i \\ y_i \end{bmatrix}$$



**Figure 8.2.** Positions and distances in a formation of two quadrotors

#### 8.3.1. Interactive potential energy and force

The target of each rotorcraft in the formation is to reach its goal position ( $\xi_{gi}$ ). Hence, for the  $i$ th rotorcraft, we consider the distance between the actual position and its goal position, which is computed as follows:

$$L_{ig} = \|\xi_{gi} - \xi_i\|$$

For convenience, the goal position error is obtained as  $\tilde{L}_{ig} = -L_{ig}$ . Now, define the distance between the  $i$ th and the  $j$ th rotorcrafts as:

$$L_{ij} = \|\xi_j - \xi_i\|$$

and the vehicle's distance error as:

$$\tilde{L}_{ij} = L_{ijd} - L_{ij}$$

where  $L_{ijd}$  denotes the desired separation between the  $i$ th and the  $j$ th rotorcrafts.

The artificial potential energy function chosen between the aircrafts  $i$  and  $j$ , and with respect to the goal position, are expressed, respectively, as:

$$\mathcal{U}_{ij} = \frac{1}{2}k_f \tilde{L}_{ij}^2$$

$$\mathcal{U}_{ig} = \frac{1}{2}k_f \tilde{L}_{ig}^2$$

where  $k_f$  is the gain of the formation and it is a constant value. The total structural potential energy from all vehicles, around the  $i$ th vehicle, is then

$$\mathcal{U}_i = \sum_{j \neq i}^n \mathcal{U}_{ij} + \mathcal{U}_{ig}$$

The corresponding structural force is given by the negative gradient of the structural potential energy. This force, in the case of the vehicle  $i$  with respect to the vehicle  $j$ , can be expressed as:

$$\mathbf{f}_{ij} = -\nabla \mathcal{U}_{ij} = k_f \tilde{L}_{ij} \frac{\boldsymbol{\xi}_j - \boldsymbol{\xi}_i}{L_{ij}} \quad [8.7]$$

where  $(\boldsymbol{\xi}_j - \boldsymbol{\xi}_i)/L_{ij}$  denotes a unitary vector pointing toward the current desired position with respect to the rotorcraft  $j$ . In the same way, we can obtain the force of the rotorcraft  $i$  with respect to its goal:

$$\mathbf{f}_{ig} = -\nabla \mathcal{U}_{ig} = k_f \tilde{L}_{ig} \frac{\boldsymbol{\xi}_g - \boldsymbol{\xi}_i}{L_{ig}}$$

then the total structural attractive force acting on the  $i$ th rotorcraft  $\mathbf{F}_{att,i}$  can be computed by:

$$\begin{aligned} \mathbf{F}_{att,i} &= -\nabla \mathcal{U}_i = \sum_{j \neq i}^n \mathbf{f}_{ij} + \mathbf{f}_{ig} \\ &= \sum_{i \neq j}^n k_f \tilde{L}_{ij} \frac{\boldsymbol{\xi}_j - \boldsymbol{\xi}_i}{L_{ij}} + k_f \tilde{L}_{ig} \frac{\boldsymbol{\xi}_g - \boldsymbol{\xi}_i}{L_{ig}} \end{aligned} \quad [8.8]$$

where  $n$  is the number of vehicles in the formation, and its time derivative is given by:

$$\dot{\mathbf{F}}_{att,i} = -\sum_{i \neq j}^n k_f (\dot{\boldsymbol{\xi}}_j - \dot{\boldsymbol{\xi}}_i) + k_f \dot{\boldsymbol{\xi}}_i = k_f n \dot{\boldsymbol{\xi}}_i - k_f \sum_{i \neq j}^n \dot{\boldsymbol{\xi}}_j$$

### 8.3.2. Collision avoidance

The attractive function [8.7] between two vehicles, by its definition, makes the vehicles experience a repulsion when the distance  $L_{ij}$  is smaller than  $L_{ijd}$ , in order to maintain them at the desired separation. With the addition of the attractive force toward the goal position  $\mathbf{f}_{ig}$  and even more attractive functions to other vehicles, the repulsion between two vehicles is not enough to avoid collisions. The strategy is to add a force of the same magnitude that is attractive, but in the opposite sense.

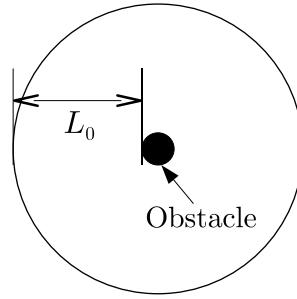
To ensure collision avoidance, let us define a repulsive force between vehicles as follows:

$$\mathbf{f}_{r,ij} = \begin{cases} \|\mathbf{F}_{att,i}\| \frac{\xi_i - \xi_j}{L_{ij}}, & L_{ij} \leq L_0 \\ 0, & L_{ij} > L_0 \end{cases}$$

This repulsive force has the same magnitude as the attractive force, but it goes in the opposite way.  $L_0$  is a distance large enough to ensure no collisions. Thus, the separation desired distance between each pair of rotorcrafts should be greater than  $L_0$ .

### 8.3.3. Obstacle avoidance

Let us assume a static environment, i.e. the obstacles are fixed in location. In addition, it is assumed that the obstacles have a round shape. A repulsive potential energy has to be defined in order to reach obstacle avoidance, which only acts in a region nearby to the obstacle, as shown in Figure 8.3.



**Figure 8.3.** Length of action of the repulsive force

Let us take into account the forces acting on the vehicle to generate the repulsive function. For one rotorcraft  $i$ , close to the  $k$ th obstacle, the repulsive force is given by:

$$\mathbf{f}_{r,io_k} = \begin{cases} (\|\mathbf{F}_{att,i}\| + \|\sum_{j \neq i}^n \mathbf{f}_{r,ij}\| + b_1) \frac{\xi_i - \xi_{o_k}}{L_{io_k}}, & L_{io_k} \leq L_0 \\ 0, & L_{io_k} > L_0 \end{cases}$$

where  $b_1$  is the smallest bound of the nested saturation control,  $L_{io_k} = \|\xi_{o_k} - \xi_i\|$  corresponds to the distance between the center of mass of the quadrotor and the nearest border of the obstacle  $k$ ,  $\xi_{o_k}$  denotes the obstacle position, and  $L_0$  is the distance at which the repulsive function begins to act.

#### 8.3.4. Total structural force

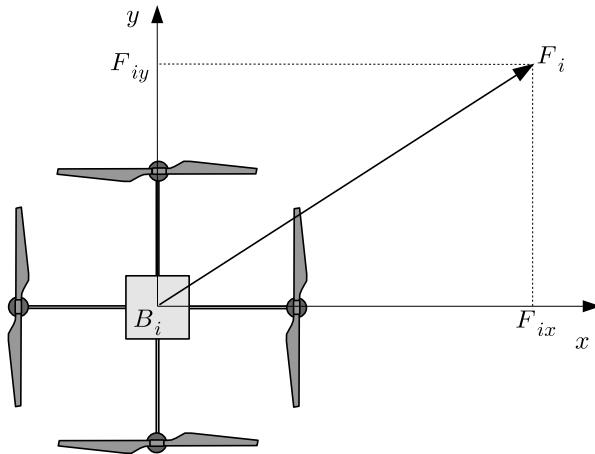
Using the same idea of the attractive forces, the repulsive forces are added as:

$$\mathbf{F}_{rep,i} = \sum_{j \neq i}^n \mathbf{f}_{r,ij} + \sum_{k=1}^K \mathbf{f}_{r,io_k}$$

where  $K$  corresponds to the number of obstacles obstructing the free trajectory of the  $i$ th rotorcraft. Then, the total structural force is given by:

$$\mathbf{F}_i = \mathbf{F}_{att,i} + \mathbf{F}_{rep,i}$$

The total structural force  $\mathbf{F}_i$  represents a vector containing a component in the  $x$ -axis, denoted by  $F_{ix}$ , and a component in the  $y$ -axis, denoted by  $F_{iy}$ . The  $x$  and  $y$  components of the structural force  $\mathbf{F}_i$  are reflected on the body's frame of the  $i$ th rotorcraft.



**Figure 8.4.** Representation of the force vector

#### 8.4. Position control

The previous analysis considers that, before and during the formation process, all aircrafts achieve and maintain a desired fixed altitude  $z_d$  and a desired fixed yaw angle (e.g.  $\psi_d = 0$ ). Then, we start controlling such states.

### 8.4.1. Altitude and yaw control

Assume that  $\psi = 0 \forall t > 0$ , that is each quadrotor maintains its yaw orientation during the formation. The yaw control is then obtained by:

$$\tau_{\psi_i} = -k_{p\psi}\psi_i - k_{v\psi}\dot{\psi}_i \quad [8.9]$$

where  $k_{p\psi}$  and  $k_{v\psi}$  denote the proportional and derivative (PD) constants for this PD yaw control. Also, if it is supposed that each rotorcraft achieves and maintains a desired altitude, then the vertical position  $z$  can be achieved through the PD control input:

$$u_i = [(k_{pz}\tilde{z}_i - k_{vz}\dot{z}_i) + mg]\frac{1}{\cos\theta_i \cos\phi_i} \quad [8.10]$$

where  $\tilde{z}_i = z_{id} - z_i$  denotes the altitude error,  $k_{pz}$  and  $k_{vz}$  are positive constants related with a PD altitude controller. The angles  $\theta_i$  and  $\phi_i$  should be maintained in small values to avoid singularities in [8.10].

Using [8.10] and [8.9] after a finite time, we have  $\tilde{z}_i, \dot{z}_i, \psi_i$ , and  $\dot{\psi}_i \rightarrow 0$ . Hence, the system [8.1]–[8.6] can be considered as:

$$\ddot{x}_i = g \tan \theta_i \quad [8.11]$$

$$\ddot{y}_i = -g \frac{\tan \phi_i}{\cos \theta_i} \quad [8.12]$$

$$\ddot{\theta}_i = \tau_{\theta_i} \quad [8.13]$$

$$\ddot{\phi}_i = \tau_{\phi_i} \quad [8.14]$$

Imposing very small bounds on  $\theta_i$  and  $\phi_i$ , in such a way that  $\tan(\theta_i) \approx \theta_i$ , and so on with  $\phi_i$ , the subsystem for the  $i$ th rotorcraft is reduced to:

$$\ddot{x}_i = g\theta_i \quad [8.15]$$

$$\ddot{y}_i = -g\phi_i \quad [8.16]$$

$$\ddot{\theta}_i = \tau_{\theta_i} \quad [8.17]$$

$$\ddot{\phi}_i = \tau_{\phi_i} \quad [8.18]$$

which can be seen as two independent systems, where each one is composed of four integrators in cascade.

### 8.4.2. Nested saturation control

This control method was proposed to globally asymptotically stabilize a chain of  $n$  integrators with one input [TEE 92]. Further, it was applied to stabilize a quadrotor

system [CAS 04, SÁN 08]. In this subsection, we present the way to link the computed force with the rotorcraft's position control. The responsibility of the potential force is to generate the direction that each rotorcraft must follow to achieve its desired position to avoid collision, meanwhile the nested saturation control is responsible for executing the movement of each aircraft.

#### 8.4.2.1. Change of variables for the nested saturation

To use the nested saturation method, it is necessary to apply a change of variables. Based on [JOH 03] we will explain the way to realize the transformation.

LEMMA 8.1.– Consider a chain of  $n$  integrators, as

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= x_3 \\ &\vdots \\ \dot{x}_{n-1} &= x_n \\ \dot{x}_n &= u \end{aligned} \tag{8.19}$$

that may be represented as  $\dot{\mathbf{x}} = \mathbf{A}_x \mathbf{x} + \mathbf{B}_x u$ , with  $\mathbf{x} \in \mathbb{R}^n$ ,  $u \in \mathbb{R}$ , and

$$\mathbf{A}_x = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ \vdots & & & & \vdots \\ 0 & \cdots & \cdots & \cdots & 1 \\ 0 & \cdots & \cdots & \cdots & 0 \end{bmatrix}, \quad \mathbf{B}_x = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix}$$

then there exists a linear transformation  $\mathbf{z} = \mathbf{T}_{zx} \mathbf{x}$  which transforms [8.19] into  $\dot{\mathbf{z}} = \mathbf{A}_z \mathbf{z} + \mathbf{B}_z u$ , where

$$\mathbf{A}_z = \begin{bmatrix} 0 & k_2 & k_3 & \cdots & k_m \\ 0 & 0 & k_3 & \cdots & k_m \\ \vdots & & & & \vdots \\ 0 & 0 & 0 & \cdots & k_m \\ 0 & 0 & 0 & \cdots & 0 \end{bmatrix}, \quad \mathbf{B}_z = \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \\ 1 \end{bmatrix}$$

and the elements  $k_1, \dots, k_m \in \mathbb{R} \setminus 0$ .

PROOF.– Consider the set of coefficients

$$K = \{k_m, k_{m-1}, \dots, k_2, k_1\}$$

Let  $K_l \subseteq K$  represent a subset containing the first  $l$  elements of  $K$ . Define a function  $F_k^h(K_l)$  which acts over the set  $K_l$ .  $F_k^h(K_l)$  is used to generate the product of

combination of elements taking  $h$  at time from  $K_l$ . The number of such combinations is given by the binomial coefficient:

$$\bar{C}_h^l = \binom{l}{h}$$

then  $F_k^h(K_l)$  generates the  $k$ th combination of the products of  $m$  elements taken from the set  $K_l$  without repetition. Note that  $F_k^0 = 1$ .

In order to generate the transformation  $T_{zx}$ , define the function  $C(l, h)$ , with  $l \in [0, \dots, m]$ ,  $h \in [0, \dots, l]$ , and  $h \leq l$ , over the set of coefficients  $K$ .

$$C(l, h) = \sum_{k=1}^{\bar{C}_h^l} F_k^h(K_l)$$

$$C(l, 0) = 1$$

The new coordinate system is characterized by:

$$z_{m-i} = \sum_{j=0}^i C(i, j) x_{n-j}$$

where  $i \in [0, \dots, m - 1]$ , and the elements of the transformation matrix  $\mathbf{T}_{zx}$  are:

$$\mathbf{T}_{zx(m-i),(m-j)} = \begin{cases} C(i, j), & i \geq j \\ 0, & i < j \end{cases}$$

EXAMPLE.– In the case of four integrator in cascade:

$$\dot{x}_1 = x_2$$

$$\dot{x}_2 = x_3$$

$$\dot{x}_3 = x_4$$

$$\dot{x}_4 = u$$

The elements of the transformation matrix  $\mathbf{T}_{zx}$  are:

$$T_{zx_{1,1}} = k_2 k_3 k_4$$

$$T_{zx_{1,2}} = k_2 k_3 + k_2 k_4 + k_3 k_4$$

$$T_{zx_{1,3}} = k_2 + k_3 + k_4$$

$$T_{zx_{1,4}} = 1$$

$$T_{zx_{2,2}} = k_3 k_4$$

$$T_{zx_{2,3}} = k_3 + k_4$$

$$T_{zx_{2,4}} = 1$$

$$T_{zx_{3,3}} = k_4$$

$$T_{zx_{3,4}} = 1$$

and the new coordinates:

$$\begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \end{bmatrix} = \begin{bmatrix} k_2 k_3 k_4 & k_2 k_3 + k_2 k_4 + k_3 k_4 & k_2 + k_3 + k_4 & 1 \\ 0 & k_3 k_4 & k_3 + k_4 & 1 \\ 0 & 0 & k_4 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix}$$

#### 8.4.2.2. Nested saturation formation control

For the  $i$ th quadrotor, we regroup the equations [8.15]–[8.18] as:

$$\ddot{\xi}_i = \bar{g}\eta_i \quad [8.20]$$

$$\ddot{\eta}_i = \tau_i \quad [8.21]$$

where  $\xi_i = [x_i \ y_i]^T$ ,  $\eta_i = [\theta_i \ \phi_i]^T$ ,  $\tau_i = [\tau_{\theta i} \ \tau_{\phi i}]^T$ , and

$$\bar{g} = \begin{cases} g & \text{for } x - \theta \text{ coordinates} \\ -g & \text{for } y - \phi \text{ coordinates} \end{cases}$$

To apply the nested saturation method to the system [8.20]–[8.21], it is necessary to transform coordinates according to the method of the transformation matrix  $T_{zx}$ . Let us call, in this case, the transformation matrix  $T$ , then the new coordinates are:

$$\begin{bmatrix} z_{1i} \\ z_{2i} \\ z_{3i} \\ z_{4i} \end{bmatrix} = \begin{bmatrix} \frac{k_2 k_3 k_4}{\bar{g}} & \frac{k_2 k_3 + k_2 k_4 + k_3 k_4}{\bar{g}} & k_2 + k_3 + k_4 & 1 \\ 0 & \frac{k_3 k_4}{\bar{g}} & k_3 + k_4 & 1 \\ 0 & 0 & k_4 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \xi_i \\ \xi_i \\ \eta_i \\ \dot{\eta}_i \end{bmatrix} \quad [8.22]$$

However, to stabilize the vehicles at a desired position of a formation of  $n$  vehicles, let us introduce the computed force  $F_i$  and assume now that the force is computed by considering the other vehicles fixed (afterward we will consider moving vehicles). Then, we have

$$\dot{F}_i = -k_f n \dot{\xi}_i$$

$$\ddot{\xi}_i = \bar{g}\eta_i$$

$$\dot{\eta}_i = \dot{\eta}_i$$

$$\ddot{\eta}_i = \tau_i$$

and we can observe that it is still a system of integrators in cascade. With the nested saturation method, consider a diffeomorphism in the form  $\mathbf{z} = \mathbf{T}_f \mathbf{q}$  as follows:

$$\begin{bmatrix} z_{1i} \\ z_{2i} \\ z_{3i} \\ z_{4i} \end{bmatrix} = \begin{bmatrix} -\frac{k_2 k_3 k_4}{\bar{g} n} & \frac{k_2 k_3 + k_2 k_4 + k_3 k_4}{\bar{g}} & k_2 + k_3 + k_4 & 1 \\ 0 & \frac{k_3 k_4}{\bar{g}} & k_3 + k_4 & 1 \\ 0 & 0 & k_4 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \mathbf{F}_i \\ \dot{\xi}_i \\ \eta_i \\ \dot{\eta}_i \end{bmatrix} \quad [8.23]$$

and  $\mathbf{z}_{pi} = [z_{pix} \ z_{piy}]^T$  with  $p = 1, \dots, 4$ . The controller gains are positive constants and they are represented by  $k_1, \dots, k_4$ . The difference between  $\mathbf{T}$  [8.22] and  $\mathbf{T}_f$  [8.23] is in the division of the number of vehicles  $n$  in the first element of the transformation matrix ( $T_{1,1}$ ) to reduce the effect of the number of vehicles and the minus sign; nevertheless, we had not divided the gain  $k_f$  because it has been introduced in the force to change the behavior of the system.

Define a saturation functions of the form:

$$\sigma_b(s) = \begin{cases} -b & \text{for } s < -b \\ s & \text{for } -b \leq s \leq b \\ b & \text{for } s > b \end{cases}$$

where  $b$  is a positive constant value.

The signal for *pitch control* ( $\theta, x$ ) is given by [CAS 04]:

$$\tau_{\theta i} = -\sigma_{b_4}(k_4 z_{4ix} + \sigma_{b_3}(k_3 z_{3ix} + \sigma_{b_2}(k_2 z_{2ix} + \sigma_{b_1}(k_1 z_{1ix})))) \quad [8.24]$$

and for the *roll control* ( $\phi, y$ ), we have

$$\tau_{\phi i} = -\sigma_{b_4}(k_4 z_{4iy} + \sigma_{b_3}(k_3 z_{3iy} + \sigma_{b_2}(k_2 z_{2iy} + \sigma_{b_1}(k_1 z_{1iy})))) \quad [8.25]$$

#### 8.4.3. Stability analysis

We will begin our analysis based on [JOH 03]. Considering the fact that [8.24] and [8.25] are similar, then we focus our analysis for the pitch control  $\tau_{\theta i}$ . For simplicity, we rename  $z_{m_{ix}}$  as  $z_m$

By using the coordinate transformation [8.23] on the system [8.11], through the control law [8.24], we have

$$\dot{z}_2 = k_4 z_4 + k_3 z_3 - \sigma_{b_4}(k_4 z_4 + \sigma_{b_3}(\cdot)) \quad [8.26]$$

$$\dot{z}_3 = k_4 z_4 - \sigma_{b_4}(k_4 z_4 + \sigma_{b_3}(\cdot)) \quad [8.27]$$

$$\dot{z}_4 = -\sigma_{b_4}(k_4 z_4 + \sigma_{b_3}(\cdot)) \quad [8.28]$$

Choosing a function:

$$V_m = \frac{1}{2}z_m^2$$

where  $m = 4$ , whose time derivative is given by:

$$\dot{V}_m = z_m \dot{z}_m = -z_m \sigma_{b_m}(k_m z_m + \sigma_{b_{(m-1)}}(\cdot))$$

The terms  $z_m$  and  $\sigma_{b_m}(\cdot)$  will have the same sign only if the addition  $k_m z_m + \sigma_{b_{(m-1)}}(\cdot)$  has the same sign as  $z_m$ . Having chosen  $b_{m-1} < 1/2b_m$ , it can be verified that  $\dot{V}_m < 0$  for all  $z_m \in Q_m = \{z_m : |k_m z_m| \leq 1/2b_m\}$ . If the vehicle starts outside  $Q_m$ , the trajectory of  $z_m$  eventually enters  $Q_m$  in a finite time. Once  $z_m$  has entered  $Q_m$ ,  $\sigma_{b_m}(\cdot)$  it operates in its linear region because the argument to  $\sigma_{b_m}(\cdot)$  is bounded by:

$$|k_n z_m + \sigma_{b_{(m-1)}}(\cdot)| < \frac{1}{2}b_m + b_{m-1} < b_m$$

The equation for the evolution of  $z_{m-1}$  is now given by:

$$\dot{z}_{m-1} = -\sigma_{b_{m-1}}(k_{m-1} z_{(m-1)} + \sigma_{b_{m-2}}(\cdot)) \quad [8.29]$$

where [8.29] is similar to the expression for  $z_m$ . As observed for  $z_m$ , we can verify that once  $z_{m-1}$  enters in a set  $Q_{m-1}$ , in a finite time, it stays in  $Q_{m-1}$  thereafter. This process remains valid for  $V_2$  and  $z_2$ , because  $z_1$  depends on the potential force computed with respect to the other vehicles.

Then, the Lyapunov function to stabilize the system [8.26]–[8.28] is  $V = V_4 + V_3 + V_2$ . According to [JOH 03] in Corollary 7.1, during periods when the outermost saturated element is the first saturator,  $\sigma_{b_1}$ , the poles of the resulting closed loop linear system reside at  $\{-k_4, -k_3, -k_2, 0\}$ , which produces a stable system. Then, the term  $k_1 z_1$  decreases below the bound  $b_1$ . The remaining stability analysis is finished by considering the system as an interconnected large-scale system.

#### 8.4.4. Stability analysis for the interconnected system

Under the bounds of the controller saturations, and with the introduction of the force  $F_{ix}$  as the position error, the system:

$$\ddot{x}_i = g\theta_i \quad [8.30]$$

$$\ddot{\theta}_i = \tau_{\theta_i} \quad [8.31]$$

could be seen as a linear and controllable system of the form:

$$\begin{bmatrix} \dot{F}_{ix} \\ \dot{x}_i \\ \dot{\theta}_i \\ \ddot{\theta}_i \end{bmatrix} = \begin{bmatrix} 0 & nk_f & 0 & 0 \\ 0 & 0 & g & 0 \\ 0 & 0 & 0 & 1 \\ -\mathcal{A}_n & -\mathcal{B} & -\mathcal{C} & -\mathcal{D} \end{bmatrix} \begin{bmatrix} x_i \\ \dot{x}_i \\ \theta_i \\ \dot{\theta}_i \end{bmatrix} + \begin{bmatrix} -k_f \sum_{i \neq j} \dot{x}_j \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad [8.32]$$

where  $\mathcal{A}_n, \mathcal{B}, \mathcal{C}, \mathcal{D} \in \mathbb{R}$  and, from the transformation matrix in [8.23]

$$\begin{aligned}\mathcal{A}_n &= k_1 T_{1,1} \\ &= \frac{k_1 k_2 k_3 k_4}{g n}\end{aligned}\quad [8.33]$$

$$\begin{aligned}\mathcal{B} &= k_2 T_{2,2} + k_1 T_{1,2} \\ &= \frac{k_1 k_2 k_3 + k_2 k_3 k_4 + k_1 k_2 k_4 + k_1 k_3 k_4}{g}\end{aligned}\quad [8.34]$$

$$\begin{aligned}\mathcal{C} &= k_3 T_{3,3} + k_2 T_{2,3} + k_1 T_{1,3} \\ &= k_1 k_2 + k_2 k_3 + k_3 k_4 + k_1 k_3 + k_3 k_4 + k_2 k_4\end{aligned}\quad [8.35]$$

$$\begin{aligned}\mathcal{D} &= k_4 T_{4,4} + k_3 T_{3,4} + k_2 T_{2,4} + k_1 T_{1,4} \\ &= k_1 + k_2 + k_3 + k_4\end{aligned}\quad [8.36]$$

Because the dynamics of each vehicle are interconnected with the other vehicles, we have to study the stability of the whole formation as one system. Let  $\mathbf{A}_n \in \mathbb{R}^{4 \times 4}$  be the matrix of the dynamics of one vehicle:

$$\mathbf{A}_n = \begin{bmatrix} 0 & nk_f & 0 & 0 \\ 0 & 0 & g & 0 \\ 0 & 0 & 0 & 1 \\ -\mathcal{A}_n & -\mathcal{B} & -\mathcal{C} & -\mathcal{D} \end{bmatrix}$$

with  $\mathbf{I}$  as the interconnection matrix:

$$\mathbf{I} = \begin{bmatrix} 0 & -k_f & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

and let  $\mathbf{x}_i \in \mathbb{R}^4$  be the vector of states,  $x - \theta$ , of the vehicle  $i$  [8.32], that is  $\mathbf{x}_i = [x_i \ \dot{x}_i \ \theta_i \ \dot{\theta}_i]^T$ . Then, the whole interconnected system for  $n$  vehicles can be represented as:

$$\dot{\mathbf{X}}_n = \mathbf{J}_n \mathbf{X}_n \quad [8.37]$$

where

$$\mathbf{X}_n = [\mathbf{x}_1 \ \mathbf{x}_2 \ \cdots \ \mathbf{x}_n]^T$$

and

$$\mathbf{J}_n = \begin{bmatrix} \mathbf{A}_n & \mathbf{I} & \cdots & \mathbf{I} \\ \mathbf{I} & \mathbf{A}_n & \cdots & \mathbf{I} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{I} & \mathbf{I} & \cdots & \mathbf{A}_n \end{bmatrix}$$

Consider, first, the case when  $n = 1$ , then

$$\dot{\mathbf{x}}_1 = \mathbf{A}_1 \mathbf{x}_1 \quad [8.38]$$

where  $\mathbf{x}_1 \in \mathbb{R}^4$  and  $\mathbf{A}_1 \in \mathbb{R}^{4 \times 4}$ . By the method's definition, if the gain  $k_f = 1$ , at the moment when  $k_1 z_1 < b_1$ , the poles of closed-loop system are located at  $\{-k_1, -k_2, -k_3, -k_4\}$  [JOH 03], which gives a stable system. By increasing the value of  $k_f$ , the system becomes faster, but with an under damped response. By increasing  $k_f$ , the system tends to be non-stable.

**LEMMA 8.2.-** Consider the controllable system [8.38] for the case of one vehicle, the value of the gain  $k_f$  that makes the system critically stable is:

$$k_{f1} = \frac{\mathcal{B}(\mathcal{CD} - g\mathcal{B})}{\mathcal{A}_1 \mathcal{D}^2}$$

**PROOF.-** In the system [8.38], the eigenvalues of the matrix  $\mathbf{A}_1$  are:

$$|\lambda \mathbf{I} - \mathbf{A}_1| = \lambda^4 + \lambda^3 \mathcal{D} + \lambda^2 \mathcal{C} + \lambda g\mathcal{B} + k_{f1} g \mathcal{A}_1 = 0. \quad [8.39]$$

On the other hand, the expression to identify the closed-loop poles is given by:

$$0 = (\lambda + \lambda_1)(\lambda + \lambda_2)(\lambda + \lambda_3)(\lambda + \lambda_4) \quad [8.40]$$

$$\begin{aligned} &= \lambda^4 + (\lambda_1 + \lambda_2 + \lambda_3 + \lambda_4)\lambda^3 + [\lambda_1\lambda_2 + (\lambda_1 + \lambda_2)(\lambda_3 + \lambda_4) + \lambda_3\lambda_4]\lambda^2 \\ &\quad + [\lambda_1\lambda_2\lambda_3 + \lambda_1\lambda_2\lambda_4 + \lambda_1\lambda_3\lambda_4 + \lambda_2\lambda_3\lambda_4]\lambda + \lambda_1\lambda_2\lambda_3\lambda_4 \end{aligned} \quad [8.41]$$

By using [8.39] and [8.41], we have

$$\mathcal{D} = \lambda_1 + \lambda_2 + \lambda_3 + \lambda_4 \quad [8.42]$$

$$\mathcal{C} = \lambda_1\lambda_2 + (\lambda_1 + \lambda_2)(\lambda_3 + \lambda_4) + \lambda_3\lambda_4 \quad [8.43]$$

$$g\mathcal{B} = (\lambda_1 + \lambda_2)\lambda_3\lambda_4 + (\lambda_3 + \lambda_4)\lambda_1\lambda_2 \quad [8.44]$$

$$k_{f1} g \mathcal{A}_1 = \lambda_1\lambda_2\lambda_3\lambda_4 \quad [8.45]$$

In order to find the critically stable point, let us define the critically stable poles:

$$\lambda_3 = 0 + j\alpha, \quad \lambda_4 = 0 - j\alpha$$

then, we obtain

$$\alpha = \sqrt{\frac{g\mathcal{B}}{\mathcal{D}}}$$

and the value  $k_f$ , for the critic stability point, is given by:

$$k_{f1} = \frac{\mathcal{B}(\mathcal{C}\mathcal{D} - g\mathcal{B})}{\mathcal{A}_1\mathcal{D}^2}$$

We verify that, for one vehicle, the system will be stable if:

$$k_f < \frac{\mathcal{B}(\mathcal{C}\mathcal{D} - g\mathcal{B})}{\mathcal{A}_1\mathcal{D}^2}$$

**LEMMA 8.3.-** *In an interconnected system of the form [8.37], the value of the gain  $k_f$  that makes the system critically stable can be computed as:*

$$k_{fn} = \frac{n\mathcal{B}(\mathcal{C}\mathcal{D} - g\mathcal{B})}{(n+1)\mathcal{A}_1\mathcal{D}^2} = \frac{\mathcal{B}(\mathcal{C}\mathcal{D} - g\mathcal{B})}{(n+1)\mathcal{A}_n\mathcal{D}^2} \quad [8.46]$$

**PROOF.-** First, let us analyze the interconnected system, for  $n = 2$ , that is  $\dot{\mathbf{X}}_2 = \mathbf{J}_2 \mathbf{X}_2$ , then

$$\begin{bmatrix} \dot{\mathbf{x}}_1 \\ \dot{\mathbf{x}}_2 \end{bmatrix} = \begin{bmatrix} \mathbf{A}_2 & \mathbf{I} \\ \mathbf{I} & \mathbf{A}_2 \end{bmatrix} \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{bmatrix}$$

where  $\mathbf{X}_2 \in \mathbb{R}^8$  and  $\mathbf{J}_2 \in \mathbb{R}^{8 \times 8}$ . The eigenvalues of the matrix  $\mathbf{J}_2$  are determined through

$$|\lambda\mathbf{I} - \mathbf{J}_2| = 0$$

On the other hand, we have

$$0 = \prod_{i=1}^8 (\lambda + \lambda_i)$$

Then, using  $|\lambda\mathbf{I} - \mathbf{J}_2| = \prod_{i=1}^8 (\lambda + \lambda_i)$ , we get

$$k_{f2} = \frac{2k_{f1}}{3}$$

Now, we analyze the system for  $n = 3$ . The linear interconnected system is given by:

$$\dot{\mathbf{X}}_3 = \mathbf{J}_3 \mathbf{X}_3$$

where  $\mathbf{X}_3 \in \mathbb{R}^{12}$  and  $\mathbf{J}_3 \in \mathbb{R}^{12 \times 12}$ .

By making equal

$$|\lambda \mathbf{I} - \mathbf{J}_3| = \prod_{i=1}^{12} (\lambda + \lambda_i)$$

we obtain

$$k_{f3} = \frac{3k_{f1}}{4}$$

In general, we can observe that, for  $n$  interconnected vehicles, the formation will be stable if:

$$k_f < \frac{n\mathcal{B}(\mathcal{CD} - g\mathcal{B})}{(n+1)\mathcal{A}_1\mathcal{D}^2} = \frac{\mathcal{B}(\mathcal{CD} - g\mathcal{B})}{(n+1)\mathcal{A}_n\mathcal{D}^2}$$

Hence, if we use a value of  $k_f$  lower than  $k_{fn}$  (given in [8.46]), then the applied control law to the formation is globally stable. Note that as  $n \rightarrow \infty$ ,  $k_{fn} \rightarrow k_{f1}$ .

**LEMMA 8.4.-** Consider an interconnected system in the form [8.37], the minimum value of the gain  $k_f$  that makes the system critically stable is given by:

$$k_{f2} = \frac{10}{3} = 3.33333.$$

**PROOF.-** For an interconnected system in the form [8.37], the gain that makes critically stable the system is:

$$k_{fn} = \frac{\mathcal{BCD} - g\mathcal{B}^2}{(n+1)\mathcal{A}_n\mathcal{D}^2}$$

Given [8.33]–[8.36], we have

$$\mathcal{BCD} - g\mathcal{B}^2 = 3n\mathcal{A}_n\mathcal{D}^2 + \gamma$$

where

$$\begin{aligned} \gamma = \frac{1}{g} & [2(k_1^2 k_2^2 k_3^2 + k_1^2 k_2^2 k_4^2 + k_1^2 k_3^2 k_4^2 + k_2^2 k_3^2 k_4^2) \\ & + (k_1 + k_2)(k_3 + k_4)(k_1^2 k_2^2 + k_3^2 k_4^2) \\ & + (k_1 + k_4)(k_2 + k_3)(k_1^2 k_4^2 + k_2^2 k_3^2) \\ & + (k_1 + k_3)(k_2 + k_4)(k_1^2 k_3^2 + k_2^2 k_4^2)] \end{aligned} \quad [8.47]$$

Moreover, each addend of [8.47] is equivalent to  $n/2\mathcal{A}_n\mathcal{D}^2$ , if and only if  $k_1 = k_2 = k_3 = k_4$ , and is greater than  $n/2\mathcal{A}_n\mathcal{D}^2$  if the constants  $k_1, \dots, k_4$  are different. Therefore,

$$\gamma \geq \frac{4n}{2}\mathcal{A}_n\mathcal{D}^2 = 2n\mathcal{A}_n\mathcal{D}^2$$

This implies that

$$k_{fn} \geq \frac{3n\mathcal{A}_n\mathcal{D}^2}{(n+1)\mathcal{A}_n\mathcal{D}^2} + \frac{2n\mathcal{A}_n\mathcal{D}^2}{(n+1)\mathcal{A}_n\mathcal{D}^2} \geq \frac{5n}{(n+1)}$$

The case for the minimum critical constant  $k_{fn}$  is when  $n = 2$ , then

$$k_{f2} \geq \frac{5n}{n+1} = \frac{10}{3}$$

Hence, we can always have a constant  $k_f > 1$  that makes the system faster, but stable.

#### 8.4.5. Bounded force

The attractive force is a vector that points toward the place of minimal potential energy, and the vehicle should move in the direction of that force every time. Nevertheless, given that the control signals [8.24]–[8.25] are bounded, the vector of force in the control is bonded too. Not considering a bounded force could cause the repulsive force to repel the vehicles in an inappropriate way, and could cause collisions. The following corollary, taken from [JOH 03], is useful to find the bound of the force.

**COROLLARY 8.1.–** *During periods when  $\sigma_{b_k}$  is the outermost saturated element in the control laws [8.24], [8.25] and the coordinate transformation used is given by [8.23], then, in steady-state, the magnitude of the  $k$ th derivative,  $\dot{x}_k$ , is given by:*

$$|\dot{x}_k| = \left| \frac{b_k}{T_{k,k}} \right|$$

**PROOF.–** If  $\sigma_{b_k}$  is saturated, the closed-loop system may be written as:

$$\dot{x}_m + k_m z_m + k_{m-1} z_{m-1} + \dots + b_k = 0$$

according to the transformation [8.22],

$$\begin{aligned} 0 &= \dot{x}_m + k_m T_{m,m} x_m + k_{m-1} \sum_{j=0}^1 T_{m-1,m-j} x_{m-j} + \dots \\ &\quad + k_{m-(k+1)} \sum_{j=0}^{m-(k+1)} T_{m-(k+1),m-j} x_{m-j} \pm b_k \end{aligned}$$

When the outermost saturated element is  $\sigma_k$ , the dynamics eventually reach a *saturated-equilibrium*, i.e. a region where higher order derivatives reach zero. Then  $\dot{x}_m, x_m, x_{m-1}, \dots, x_{m-k+2}$  go to zero. Thus, the non-zero terms are:

$$k_{m-(k+1)} T_{m-(k+1),(k+1)} x_{(k+1)} \pm b_k = 0$$

The term  $k_{m-(k+1)} T_{m-(k+1),(k+1)} = T_{k,k}$  and  $x_{k+1} = \dot{x}_k$ . After this

$$T_{k,k} \dot{x}_k = \pm b_k$$

To find the bound of the force, the term  $\sigma_{b_1}(k_1 z_1)$  of the controller should be saturated. We look for the saturation force at maximum speed of the vehicle  $\dot{x}_{max}$ , considering for this state the first element of the matrix  $T_s$  [8.22], then

$$\dot{x}_{max} = \frac{b_1}{\frac{k_2 k_3 k_4}{g}} \quad [8.48]$$

then we have:

$$\begin{aligned} -b_1 &= k_1 z_1 \\ -\frac{b_1}{k_1} &= -\frac{k_2 k_3 k_4}{g n} F_{xb} + \frac{k_2 k_3 + k_2 k_4 + k_3 k_4}{g} x_{max} \\ -\frac{b_1}{k_1} &= -\frac{k_2 k_3 k_4}{g n} F_{xb} + \frac{k_2 k_3 + k_2 k_4 + k_3 k_4}{k_2 k_3 k_4} b_1 \end{aligned}$$

where the bound of the force is:

$$b_F = F_{xb} = g n b_1 \left( \frac{1}{k_1 k_2 k_3 k_4} + \frac{k_2 k_3 + k_2 k_4 + k_3 k_4}{(k_2 k_3 k_4)^2} \right)$$

The attractive force [8.8] must be saturated by:

$$\mathbf{F}_{att,i} = \sigma_{b_F}(\mathbf{F}_{att,i})$$

#### 8.4.6. Repulsive distance

Given that the system has a dynamic of four integrators in cascade, turning back direction of the movement is not instantaneous, due to its velocity. It implies that at the moment that one vehicle enters into a repulsive zone, the vehicle will continue advancing a distance toward the obstacle before changing its direction in the opposite way.

To compute the maximum distance that the system [8.30]–[8.31] can advance at the moment of turning back direction, assume that the vehicle comes at a maximum speed over  $x$ -axis. At the moment that the vehicle crosses the origin, the position reference will be 0, then the maximum overshoot in the position (from the origin),  $L_{max}$ , is the maximum distance that the quadrotor can advance overpassing some repulsive zone.

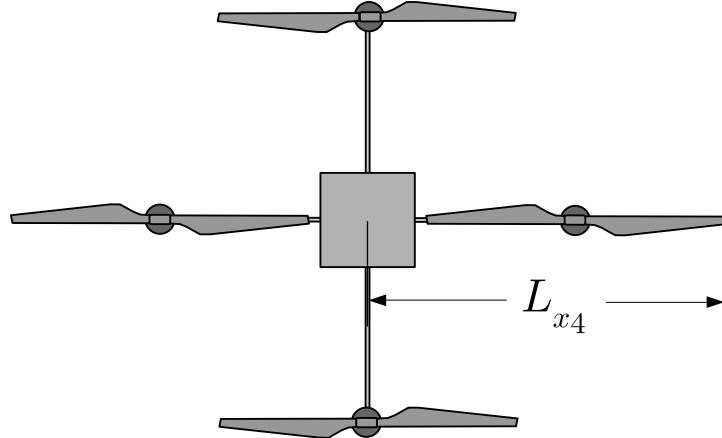
Considering that the system [8.30]–[8.31] comes at a maximum speed, we have to analyze the moment that the system leaves the saturation  $b_2$ , which is

$$\begin{bmatrix} \dot{x}_i \\ \ddot{x}_i \\ \dot{\theta}_i \\ \ddot{\theta}_i \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & g & 0 \\ 0 & 0 & 0 & 1 \\ 0 & -\mathcal{B}' & -\mathcal{C}' & -\mathcal{D}' \end{bmatrix} \begin{bmatrix} x_i \\ \dot{x}_i \\ \theta_i \\ \dot{\theta}_i \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} u \quad [8.49]$$

with  $\mathcal{B}' = k_2 k_3 k_4 / g$ ,  $\mathcal{C}' = k_2 k_3 + k_3 k_4 + k_2 k_4$ ,  $\mathcal{D}' = k_2 + k_3 + k_4$ , and  $u = -\sigma_{b_1}(k_1 z_1)$ . We just need to solve this system for initial conditions  $\mathbf{x}(0) = [0 \ \dot{x}_{max} \ 0 \ 0]^T$ . The value of  $\dot{x}_{max}$  is obtained from [8.48]. Very close results can be obtained considering the input  $u$  in [8.49] as constant:  $u = -b_1$ . This system should be solved numerically.

The distance  $L_{max}$  was obtained just in one axis, but in the diagonals of the  $x - y$  axes, the maximum distance is  $\sqrt{2}L_{max}$ . Now, just remains to add the distance  $L_{x4}$  from the center of mass of one vehicle to the tip of its propeller, as shown in Figure 8.5. Then, the safe distance  $L_0$  is:

$$L_0 > \sqrt{2}L_{max} + L_{x4}$$



**Figure 8.5.** Distance from the center of mass to the tip of the propeller

### 8.5. Simulation results

Simulations were carried out for all rotorcrafts having the same model parameters (see Table 8.1). We have the value  $L_{x4} = 0.4$  m, and, with the parameters of the table,  $L_{max} = 1.629$  m, then  $L_0 = 2.8$  m  $> \sqrt{2}L_{max} + L_{x4}$ .

Type	Parameter	Value
Rotorcraft model	$m$	1 kg
	$g$	9.81 m/s <sup>2</sup>
Controller constants	$k_1 = k_2 = k_3 = k_4$	1
Controller boundaries	$b_1$	0.1
	$b_2$	0.2
	$b_3$	0.4
	$b_4$	0.8

**Table 8.1.** Parameter and constant values

The first simulation describes a Delta formation for three rotorcrafts. The desired separation distance between rotorcrafts was chosen as  $L_{ijd} = 1.5L_0$ , and the gain of the formation  $k_f = 2$ .

Figure 8.6 shows the trajectories followed by the rotorcrafts. The sign  $\times$  represents the initial position, meanwhile the final position is marked with a quadrotor representation. The trajectories of the vehicles are not a straightline toward their goal positions because the attractive forces also act with respect to the other vehicles.

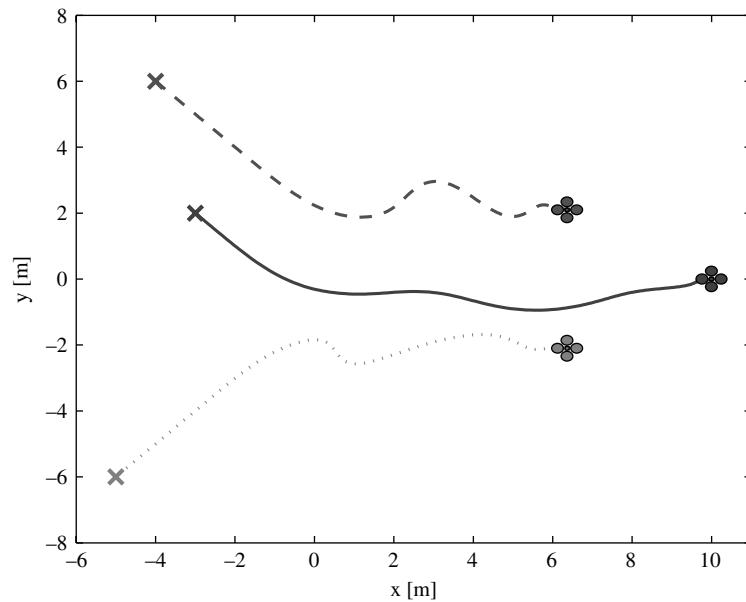
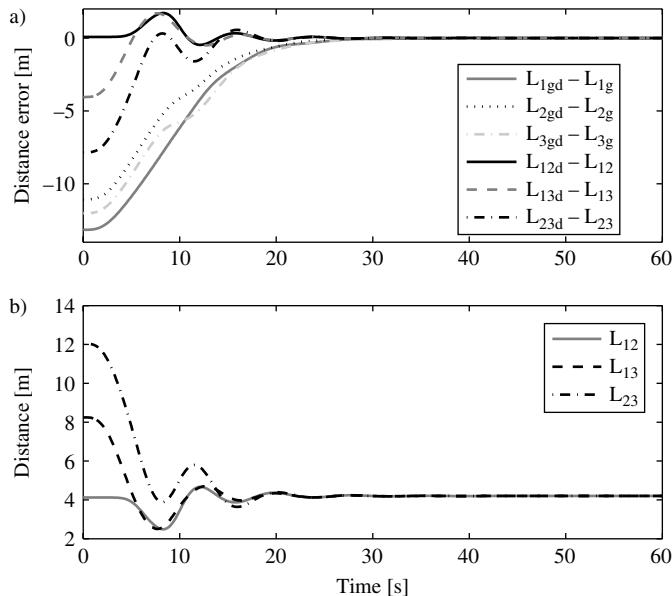
**Figure 8.6.** Formation trajectory without presence of obstacles

Figure 8.7(a) shows the distance errors  $\tilde{L}_{ij} = L_{i,jd} - L_{ij}$  and  $\tilde{L}_{ij} = L_{i,jd} - L_{ij}$ . We observe that all errors converge to zero, meanwhile Figure 8.7(b) shows the distances among vehicles; it can be seen that the measured minimum distance was greater than 2 m, clearly far from the collision distance  $2L_{x4}$ .

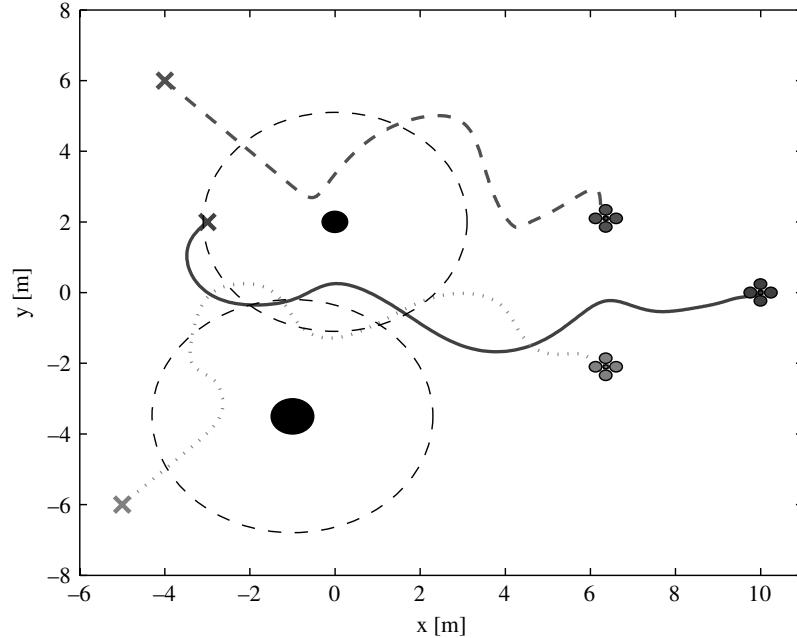


**Figure 8.7.** (a) Error distances  $\tilde{L}_{ij}$ ; (b) Distances  $L_{ij}$

### 8.5.1. Obstacle avoidance

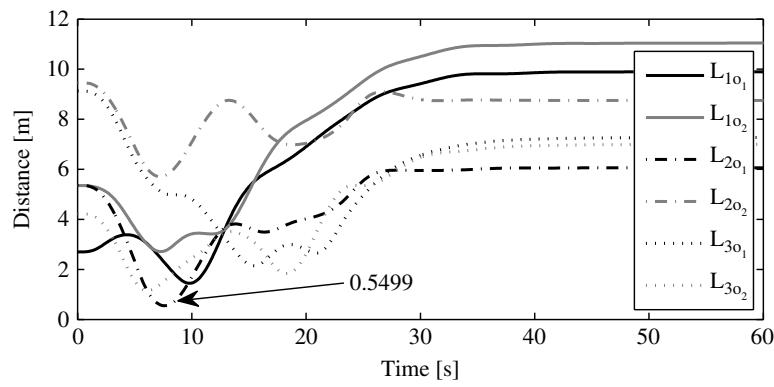
To validate the performance of the repulsive functions and also the repulsive distance, we have included obstacles in the workspace of the vehicles.

Figure 8.8 shows the trajectories followed by the rotorcrafts when they start from the same initial conditions as the first simulation, and considering the same goal positions, but in the presence of two obstacles. It seems strange that the vehicle of the top (dashed line) approaches toward the obstacle instead of avoiding it since the moment it enters into the repulsive zone, this behavior is caused by two facts: the total attractive force in that vehicle at that moment is saturated in the  $x$  and  $y$  axis; therefore, the force vector points in a direction with an angle of  $-45^\circ$  from its body axis, and due to the current velocity at the moment that the vehicle enters into the repulsive zone, it does not turn immediately to avoid the obstacle (see subsection 8.4.6) thus it continues approximating to the obstacle for a moment even though the repulsive force now points in the opposite direction.



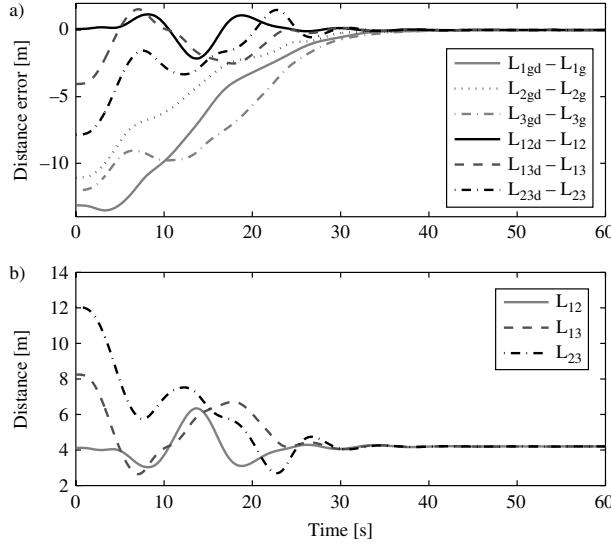
**Figure 8.8.** Formation trajectory in presence of obstacles

From Figure 8.9 we observe that vehicle 2 is very close to obstacle 1 for a while, this could be appreciated in Figure 8.8. That distance is measured from the center of mass of vehicle 2 to the border of obstacle 1. In this case the measured minimum distance is 0.5499 m, while the colliding distance is lower than  $L_{x4} = 0.4$  m, then the vehicle does not collide.



**Figure 8.9.** Distances  $\tilde{L}_{ij}$

The distance errors are shown in Figure 8.10(a), and we can see that the errors tend to zero. Figure 8.10(b) shows the distances among vehicles. We observe that the measured minimum distance is far from the collision distance  $2L_{x4}$ .



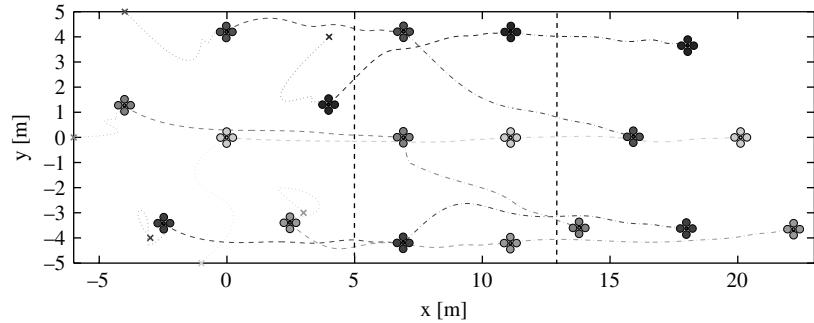
**Figure 8.10.** (a) Error distances  $\tilde{L}_{ij}$ ; (b) distances  $L_{ij}$

### 8.5.2. Multiple formations

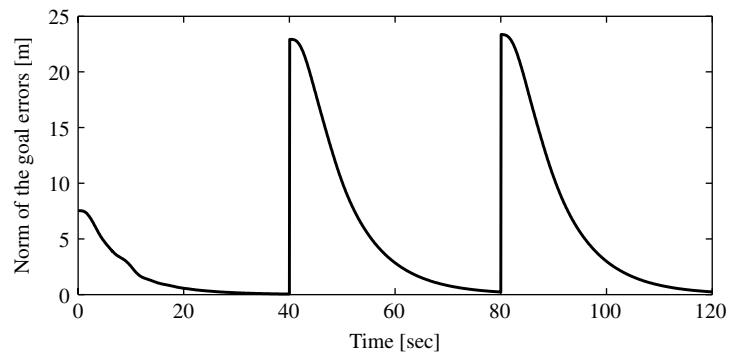
Figure 8.11 shows three different formations for six vehicles. The first formation is a star shape, the next target is a rectangle, and, once the rectangle has been formed, the last shape to be achieved is a pyramid. The gain of the formation  $k_f = 2.5$ . The executed trajectories from the initial condition to the first formation shape are indicated by dotted lines. From the first formation to the second formation, the trajectories are marked with dashed lines, and from the second to the third formation, the trajectories are shown by dashed-dotted lines.

Figure 8.12 shows the norm of the goal position errors  $\tilde{L}_{ig}$ . The first formation is defined from 0 to 40 seconds. We can see that the norm of the errors tends to 0. After the first 40 sec, the goal positions change in order to form a rectangle, then the norm of errors instantaneously goes up, but we observe that it tends to 0. Finally, in order to form the third shape, the transition occurs at 80 seconds, once again the norm of errors goes up and then it tends to 0.

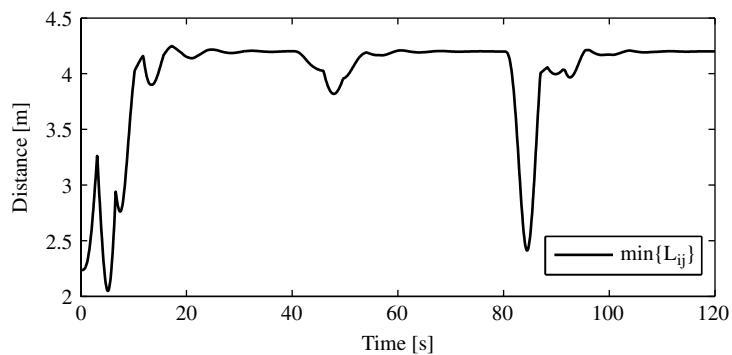
Figure 8.13 plots the smallest element of all  $L_{ij}$  for each simulation step. The smallest distances were always far from  $2L_{x4}$ , this means that there was no risk of collisions.



**Figure 8.11.** Three different formations for six vehicles



**Figure 8.12.** Norm of the goal position errors



**Figure 8.13.**  $\min\{L_{ij}\}$

### 8.6. Conclusions

The potential artificial functions are an effective way to reach formations among vehicles. In this chapter, the computed potential forces with a controller based on nested saturation functions are linked; a stability proof of the closed-loop system is also shown. With the stability analysis the conditions to choose the formation gain are obtained. Also repulsive functions that ensure collision and obstacle avoidance are proposed.

### 8.7. Bibliography

- [CAS 04] CASTILLO P., DZUL A., LOZANO R., “Real-time stabilization and tracking of a four-rotor mini rotorcraft”, *IEEE Transactions on Control Systems Technology*, vol. 12, no. 4, pp. 510–516, July 2004.
- [D'AN 03] D'ANDREA R., DULLERUD G.E., “Distributed control design for spatially interconnected systems”, *IEEE Transactions on Automatic Control*, vol. 48, no. 9, pp. 1478–1495, September 2003.
- [GAR 11] GARCÍA-DELGADO L., DZUL A., NEZ V.S., LLAMA M., “Quad-rotors formation control as an interconnected system”, *18th IFAC World Congress*, Milano, Italy, 28 August–2 September 2011.
- [GE 00] GE S.S., CUI Y.J., “New potential functions for mobile robot path planning”, *IEEE Transactions on Robotics and Automation*, vol. 16, no. 5, pp. 615–620, October 2000.
- [IOA 86] IOANNOU P.A., “Decentralized adaptive control of interconnected systems”, *IEEE Transactions on Automatic Control*, vol. AC-31, no. 4, pp. 291–298, April 1986.
- [JOH 03] JOHNSON E.N., KANNAN S.K., “Nested saturation with guaranteed real poles”, *American Control Conference*, Denver, Colorado, pp. 497–502, 4–6 June 2003.
- [SÁN 08] SÁNCHEZ A., GARCIA P., CASTILLO P., LOZANO R., “Simple real-time stabilization of a VTOL aircraft with bounded signals”, *AIAA Journal of Guidance, Control and Dynamics*, vol. 31, no. 4, pp. 1166–1176, 2008.
- [TEE 92] TEEL A.R., “Global stabilization and restricted tracking for multiple integrators with bounded controls”, *Systems and Control Letters*, vol. 18, pp. 165–171, 1992.



# Chapter 9

## Quadrotor Vision-Based Control

In this chapter, two vision-based coordination schemes for autonomous hovering and tracking of a miniature quadrotor are developed. In both the approaches, we use two cameras. One camera to compute the three-dimensional (3D) position of quadrotor with respect to a known object using image rectification. The second camera pointing downward has been mounted on the miniature quadrotor to estimate the translational velocity of the vehicle. Vanishing points techniques are used to estimate the rotation matrix and translation vector of the camera mounted on the quadrotor in order to obtain the 3D position. A nonlinear controller based on visual-based coordination strategy for a miniature quadrotor is presented. To validate the theoretical results, an embedded control system for the miniature quadrotor has been developed. These methods have been tested using real-time images; they are efficient, accurate, and robust.

### 9.1. Introduction

The study and the development of unmanned aerial vehicles (UAVs) have matured over recent years due to the fact that they can be used in a wide variety of applications, ranging from environmental monitoring in civil applications to surveillance and homeland security in the military field. Hovering platforms have the operational flexibility of being able to hover as well as to takeoff and land vertically. Currently, many researchers from various disciplines are working to improve the performance of autonomous hovering rotorcrafts.

---

Chapter written by J.E. GOMEZ-BALDERAS, J.A. GUERRERO, S. SALAZAR, R. LOZANO and P. CASTILLO.

An important issue in UAV research is the localization in unknown environments. Computer vision systems have been recently used as a solution to three-dimensional (3D) position estimation of the mini-UAVs. Many visual tasks require recovering 3D information from sequences of images, taking into account the world modeled as a projective space and determining how the projective invariant information can be recovered from the images and used in robotics applications (see [ROB 95]). The main problem in natural landmark navigation is to detect and match characteristic features from sensory inputs. The selection of features is important since it will determine the complexity in feature description, detection, and matching. Researchers have used different kinds of patterns to be recognized by systems based on computer vision [BOR 07], [ALH 10], [QUI 05]. The use of a target for autonomous landing has been applied in [SCH 07], [YU 07], and [WU 04]. A detailed summary of progress and several open problems in vision-based estimation for UAVs may be found in [KUR 04], [KEM 06], and [CON 09]. A different approach that uses video sequences to estimate the motion of UAV flight has been described in [HE 06]. Some tracking strategies using different targets are presented in [MON 07] and [DUC 09]. Caballero *et al.* [CAB 09] have shown that it is possible to determine the pose of UAV using online mosaicing over the same area. In [ALT 05], a two-camera method has been introduced for estimating the full six-degree-of-freedom pose of the helicopter. One of these cameras is located onboard the helicopter, and the other camera is located on the ground.

The relationships among camera parameters, structures in 3D scenes, and vanishing points (VPs) have been established in [HAR 80]. Also, VPs are widely employed for camera calibrations and recovery of rotational component of motion and real-time pose estimation in an urban environment [ZUW 06]. The usefulness of VP in motion analysis is the 3D orientation representation and the 3D translations between the camera and the scene. In addition, Wang and Tsai [WAN 90] and Guillou *et al.* [GUI 00] have proposed different approaches to camera calibration based upon the use of vanishing lines. This technique requires only a single view of a cube, because in the 3D space the parallel lines meet only at infinity, and a VP being, of course, equivalent to the projection of a point at infinity. On the other hand, a method that does not utilize any 3D measurements of the circle or lines in many scenarios can be found in [WAN 08a].

Several approaches to nonlinear control of a miniature quadrotor can be found in [CAS 04], [CAS 05], [SAL 09], and [ROM 09]. In this approach, the dynamics is decoupled into lateral and longitudinal dynamical subsystems. Nested saturations control was used to stabilize each subsystem. In this chapter, we propose a control scheme capable of working at a sampling rate compatible with respect to the desired bandwidth of the closed-loop system. We have a visual system with slow sampling rate and the inertial measurements unit (IMU) which operates at a fast sampling rate,

and also we present a vision-based control approach for the autonomous hovering of a quadrotor shown in Figure 9.1. The nonlinear controller is based on the separated saturations approach which takes into account the *a priori* bound on the input amplitude.



**Figure 9.1.** Miniature quadrotor with embedded control system

We are interested in the stabilization of a miniature quadrotor using a vision-based control approach. A nonlinear controller based on separated saturations using vision-based position and velocity estimation is proposed in this chapter.

The first approach uses a single camera to compute the 3D position of the vehicle with respect to a reference object placed on the wall. A second camera pointing downward has been mounted on the miniature quadrotor to estimate the translational velocity of the vehicle using a Lucas–Kanade optical flow (OF) algorithm. Due to the limited computing capability on the UAV, onboard camera image processing was not an option. Therefore, the image-processing computation has been carried out on the ground station. The approach used in this chapter uses a webcam attached at the top of the UAV airframe pointing forward. The UAV position algorithms used in our experimental test was a red color square like a target in the image plane as input and computes the position of the UAV. The distance to the target on the wall is obtained with an accuracy of  $\pm 2$  cm. A real-time platform was developed and the miniature quadrotor was able to hover autonomously.

The second approach uses a vision system to observe walls by tracking a line painted on it. To follow a trajectory, we propose an algorithm that uses the VPs from the line and then obtain the 3D position of the quadrotor.

## 9.2. Quadrotor dynamic model and control

### 9.2.1. Dynamic model

The dynamic model is obtained from [GOM 11b], and the aerodynamic and thrust forces in the body frame of reference is given by:

$$\mathcal{T} = \sum_{i=1}^4 f_i Z_{A,T} \quad [9.1]$$

where  $f_i = k_i \omega'_i{}^2$ , where  $k_i$  is a constant related to each propeller and  $\omega'_i$  is rotational velocity of each propeller. Thus, we have:

$$m\ddot{x} = -\mathcal{T} \sin \theta \quad [9.2]$$

$$m\ddot{y} = \mathcal{T} \cos \theta \sin \phi \quad [9.3]$$

$$m\ddot{z} = \mathcal{T} \cos \theta \cos \phi - mg \quad [9.4]$$

$$\ddot{\phi} = \tilde{\tau}_\phi \quad [9.5]$$

$$\ddot{\theta} = \tilde{\tau}_\theta \quad [9.6]$$

$$\ddot{\psi} = \tilde{\tau}_\psi \quad [9.7]$$

where  $m$  is the mass of the vehicle and  $g$  is the gravitational force.

### 9.2.2. Nonlinear control

In this section, a nonlinear controller based on separated saturations [TEE 92] is presented to stabilize the quadrotor in hover flight. We define

$$e_z = z - z_d$$

$$e_\psi = \psi - \psi_d$$

like the errors of the altitude and the yaw angle with respect to the desired constant values,  $z_d$  and  $\psi_d$ , respectively. In addition,  $\dot{e}_z = \dot{z}$  and  $\dot{e}_\psi = \dot{\psi}$ . Define now the following control strategies:

$$\mathcal{T} = \frac{-a_{z_1}\dot{e}_z - a_{z_2}e_z + mg}{\cos \theta \cos \phi} \quad [9.8]$$

$$\tau_{\bar{\psi}} = -a_{\psi_1}\dot{e}_\psi - a_{\psi_2}e_\psi \quad [9.9]$$

where  $a_{z_1}$ ,  $a_{z_2}$ ,  $a_{\psi_1}$ , and  $a_{\psi_2}$  are positive constants.

Using the above equations, the lateral dynamic model is represented by

$$\ddot{y} = mg \tan \phi \quad [9.10]$$

$$\ddot{\phi} = \tilde{\tau}_\phi \quad [9.11]$$

Assuming that  $\tan \phi \approx \phi$ , the above system is a four integrators in cascade. The technique based on separated saturations is used to stabilize the four integrators in cascade. The lateral controller is given by

$$\tilde{\tau}_\phi = -\sigma_{g_1}(\dot{\phi}) - \sigma_{g_2}(\dot{\phi} + \phi) - \sigma_{g_3}(\dot{\phi} + 2\phi + \dot{y}) - \sigma_{g_4}(\dot{\phi} + 3\phi + 2\dot{y} + y) \quad [9.12]$$

where  $\sigma_{g_i}$  represents a saturation function. This control law guarantees that  $\dot{\phi}, \phi, \dot{y}$ , and  $y$  converge to zero.

Using [9.8], the longitudinal dynamic model is represented by the following set of equations:

$$\ddot{x} = \frac{-mg \tan \theta}{\cos \phi} \quad [9.13]$$

$$\ddot{\theta} = \tilde{\tau}_\theta \quad [9.14]$$

It is worth mentioning that using control law [9.12] implies that  $\phi \rightarrow 0$ , which, in turn, implies that  $\cos \phi \rightarrow 1$ . Then, assuming that  $\tan \theta \approx \theta$ , the above system is again four integrators in cascade and the corresponding longitudinal controller is given as

$$\tilde{\tau}_\theta = -\sigma_{g_1}(\dot{\theta}) - \sigma_{g_2}(\dot{\theta} + \theta) - \sigma_{g_3}(\dot{\theta} + 2\theta + \dot{x}) - \sigma_{g_4}(\dot{\theta} + 3\theta + 2\dot{x} + x) \quad [9.15]$$

The proposed control law guarantees that  $\dot{\theta}, \theta, \dot{x}$ , and  $x$  converge to zero.

### 9.2.3. Trajectory-tracking control

In this section, we focus on estimating the 3D position of a quadrotor while it tracks a line painted on a wall with no marks. To solve this problem, we use the VPs of the line to obtain the rotation matrix and the translation vector of a camera placed onboard of the helicopter and pointing forward. Additionally, another camera, placed also onto the helicopter and pointing downward, is employed to calculate the OF and then the translational velocity of the quadrotor. The image information is thus introduced inside the closed-loop control of the vehicle to follow, autonomously a line. The goal of this section is to show how we can control the helicopter following a line using the vision system. The control law is based on saturation functions and is obtained employing the Lyapunov analysis.

Moreover, the control input for the lateral and longitudinal dynamics will be obtained using the Lyapunov analysis and the saturation functions in such a way that the bounds can be chosen small enough to guarantee that  $\cos \theta \cos \phi \neq 0$ .

Recall that lateral and longitudinal dynamics are given by

$$\begin{aligned}\ddot{x} &= -\frac{mg \tan \theta}{\cos \phi} \\ \ddot{\theta} &= \tau_{\bar{\theta}} \\ \ddot{y} &= mg \tan \phi \\ \ddot{\phi} &= \tau_{\bar{\phi}}\end{aligned}\tag{9.16}$$

The control objective is to follow a trajectory in the  $x$ -axis, while the  $z, \psi, \phi$ , and  $y$  are stabilized around to desired constant values. In order to stabilize the lateral dynamics, we propose a nonlinear control law bounding every state in such a way that the algorithm will guarantee an arbitrary bound for  $\phi, \dot{\phi}, \ddot{\phi}$ , and  $y$ . To further simplify the analysis, we will propose a very small upper bound on  $|\phi|$  in such a way that the difference  $\tan \phi - \phi$  is arbitrarily small. Thus, the lateral dynamics could be rewritten as

$$\begin{aligned}\ddot{y} &\simeq mg\phi \\ \ddot{\phi} &= \tau_{\bar{\phi}}\end{aligned}$$

Propose

$$\tau_{\bar{\phi}} = -\sigma_{\phi_4}(k_{\phi_4}\dot{e}_y) - \sigma_{\phi_3}(k_{\phi_3}e_y) - \sigma_{\phi_2}(k_{\phi_2}\dot{e}_{\phi}) - \sigma_{\phi_1}(k_{\phi_1}e_{\phi})\tag{9.17}$$

where  $e_{\phi} = \phi - \phi_d$  and  $e_y = y - y_d$ , with  $\phi_d$  and  $y_d$  being the desired constant values and  $\sigma_{\phi_i}(\cdot)$  is a saturation function (see [SAN 10]). From [SAN 10], it follows that  $\dot{e}_{\phi}, \dot{e}_y, e_{\phi}, e_y \rightarrow 0$ . This implies that  $\phi \rightarrow \phi_d$  and  $y \rightarrow y_d$ . In addition, the bound for the  $\theta$  angle will be chosen such that  $\tan \theta \approx \theta$ .

For the longitudinal dynamics, let us propose  $k_g e_x = x - x_d$ , with  $x_d(t) = b_0 + b_1 t + b_2 t^2$  is defined like the desired trajectory,  $k_g = g$  and  $b_i$  are constants. Thus,  $k_g \dot{e}_x = \dot{x} - \dot{x}_d$ ,  $k_g \ddot{e}_x = \theta - \ddot{x}_d$ ,  $k_g e_x^{(3)} = \dot{\theta}$ , and

$$e_x^{(4)} = \tau_{\bar{\theta}}$$

Again, similarly for the lateral dynamics, we propose the following nonlinear control:

$$\tau_{\bar{\theta}} = \sigma_{\theta_4}(k_{\theta_4}\dot{e}_x) + \sigma_{\theta_3}(k_{\theta_3}e_x) - \sigma_{\theta_2}(k_{\theta_2}\dot{e}_{\theta}) - \sigma_{\theta_1}(k_{\theta_1}e_{\theta})\tag{9.18}$$

The constants  $k_{\phi_i}$  and  $k_{\theta_i}$  are chosen as given in [SAN 10] and [GOM 11a] in such a way that the convergence to zero is ensured. Thus, this implies that  $e_x^{(3)}, \dot{e}_x, \dot{e}_x, e_x \rightarrow 0$ , and by consequence  $\phi \rightarrow \ddot{x}_d$ ,  $\dot{x} \rightarrow \dot{x}_d$ , and  $x \rightarrow x_d$ .

### 9.3. Computer vision preliminaries

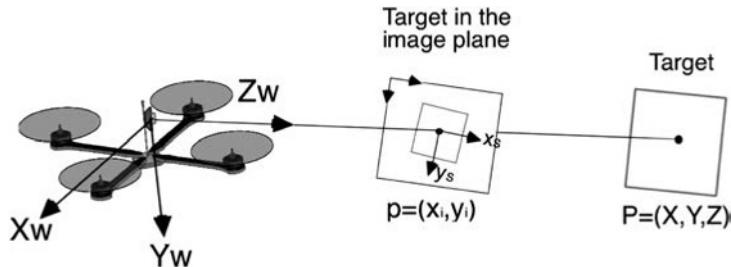
#### 9.3.1. Camera model

The classical geometric model of a camera is based on the pure perspective projection concepts. Camera calibration is composed basically in the computation of the extrinsic and intrinsic parameters of the camera. The extrinsic parameters of a camera indicate the position and the orientation of the camera with respect to the coordinate system, and the intrinsic parameters characterize the inherent properties of the camera optics, including the focal length, the image center, the image scaling factor, and the lens distortion coefficients. In the pinhole camera model, a projection point  $\mathbf{p}_i = (x_i, y_i)^T$  in the camera image is related to a 3D point  $\mathbf{P} = (X_W, Y_W, Z_W)^T$  relative to the world reference frame with coordinates  $\mathbf{P}_c = (X, Y, Z)^T$  in the camera frame by the perspective relations

$$x_i = S_x f \frac{X}{Z} + c_x$$

$$y_i = S_y f \frac{Y}{Z} + c_y$$

where  $x_i$  and  $y_i$  are the picture elements (pixels) coordinates on the image plane,  $f$  is the focal distance,  $S_x, S_y$  are two scale parameters expressed in units of pixel per millimeter, and  $(c_x, c_y)^T$  is the origin of the image coordinate system. The coordinates  $\mathbf{P}_c$  of the same point  $p_i$  relative to the camera frame are given by a rigid-body transformation of  $\mathbf{P}$ . These coordinates system are shown in Figure 9.2.



**Figure 9.2.** Camera model of the quadrotor

In the pinhole camera model, the image  $x = (x, y)$  of the point  $p$  is given by:

$$x = f \frac{X}{Z}$$

$$y = f \frac{Y}{Z}$$

where  $(x, y)$  are specified in terms of metric units and  $(x_s, y_s)$  are scaled versions that correspond to coordinates of the pixel. In homogeneous coordinates, this relationship can be written as:

$$Zx = \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \quad [9.19]$$

Since the coordinate  $Z$  is usually unknown, we may simply write it as an arbitrary positive scalar  $\lambda$ . In general, the actual origin of the camera coordinate system is conventionally specified relative to the upper left corner. Therefore, we need to translate the origin of the reference frame to this corner

$$x_i = x_s + c_x$$

$$y_i = y_s + c_y$$

where  $(c_x, c_y)$  are the coordinates (in pixels) of the principal point relative to the image reference frame. In the homogeneous representation,

$$\begin{aligned} \begin{pmatrix} x_i \\ y_i \\ 1 \end{pmatrix} &= \begin{bmatrix} S_x f & -S_x f \cot \eta & c_x \\ 0 & \frac{S_y f}{\sin \eta} & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} x_s \\ y_s \\ 1 \end{pmatrix} \\ &= \mathbf{K} \begin{pmatrix} x_s \\ y_s \\ 1 \end{pmatrix} \\ \lambda \begin{pmatrix} x_i \\ y_i \\ 1 \end{pmatrix} &= (\mathbf{K} \quad \mathbf{0}) \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \end{aligned} \quad [9.20]$$

$(X, Y, Z, 1)^T$  denotes the homogeneous coordinate vector of  $\mathbf{P}_c$  in the camera coordinate system. Homogeneous coordinates have allowed us to represent the perspective projection mapping by the  $3 \times 4$  matrix:

$$\begin{aligned} \mathbf{M} &= (\mathbf{K} \quad \mathbf{0}) \\ &= \begin{bmatrix} f_x & f_s & c_x & 0 \\ 0 & f_y & c_y & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \end{aligned}$$

where the parameters  $(f_x, f_y)$  are the horizontal and the vertical scale factors expressed in pixels/unit length, converting metric distance along the  $(x_s, y_s)$  directions into pixels units.  $f_x = S_x f$ ,  $f_y = S_y f$ , and  $f_s = f_x \cot \eta$ , where the angle  $\eta$  models the non-orthogonality between the rows and columns in the image.

The transformation process between the world frame and the camera frame requires both rotation  $\mathbf{R}$  and translation  $\mathbf{t}$  in the camera coordinate system. The transformation can be written as

$$\mathbf{T} = \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0}^T & 1 \end{bmatrix}$$

$$\mathbf{t} = \begin{pmatrix} t_x \\ t_y \\ t_z \end{pmatrix}; \mathbf{R} = \begin{bmatrix} R_{11} & R_{12} & R_{13} \\ R_{21} & R_{22} & R_{23} \\ R_{31} & R_{32} & R_{33} \end{bmatrix}$$

From the coordinate transformation, we have for  $\mathbf{P}_c = [X, Y, Z, 1]^T$

$$\begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} = \mathbf{T} \begin{bmatrix} X_W \\ Y_W \\ Z_W \\ 1 \end{bmatrix}$$

The perspective transformation matrix, which relates 3D world coordinates and 2D image coordinates, is given by

$$\lambda \begin{pmatrix} x_i \\ y_i \\ 1 \end{pmatrix} = \mathcal{M} \begin{pmatrix} X_W \\ Y_W \\ Z_W \\ 1 \end{pmatrix}$$

where  $\mathcal{M} = K(\mathbf{R} - \mathbf{t})$  is the standard projective matrix, also defined up to scale, with 11 free coefficients, five intrinsic parameters ( $f_x, f_y, c_x, c_y, \eta$ ), and six extrinsic parameters (the three angles defining  $\mathbf{R}$  and the three coordinates of  $\mathbf{t}$ ), which matches the number of independent coefficients of  $\mathcal{M}$ .

The matrix  $\mathcal{M}$  can be rewritten explicitly as a function of the intrinsic and extrinsic parameters of the camera:

$$\mathcal{M} = \begin{bmatrix} f_x \mathbf{R}_1^T - f_x \cot \eta \mathbf{R}_2^T + c_x \mathbf{R}_3^T & f_x t_x - f_x \cot \eta t_y + c_x t_z \\ \frac{f_y}{\sin \eta} \mathbf{R}_2^T + c_y \mathbf{R}_3^T & \frac{f_y}{\sin \eta} t_y + c_y t_z \\ \mathbf{R}_3^T & t_z \end{bmatrix}$$

where  $\mathbf{R}_1^T, \mathbf{R}_2^T$ , and  $\mathbf{R}_3^T$  denote the three rows of the  $\mathbf{R}$  matrix, and  $t_x, t_y$ , and  $t_z$  are the coordinates of the vector  $\mathbf{t}$  in the frame attached to the camera.

### 9.3.2. Projective distortion removal

It is possible to recover the affine properties from images by using the transformation matrix  $\mathbf{H}_1$ , which maps the vanishing line back into a line at

infinity  $\mathbf{l}^\infty$ . We take two pairs of images lines  $\mathbf{l}^{(1)}, \mathbf{l}^{(2)}, \mathbf{m}^{(1)}, \mathbf{m}^{(2)}$ , which correspond to the lines connecting the points  $p_1, p_2$  parallel to  $p_3, p_4$ , and  $p_1, p_3$  parallel to  $p_2, p_4$ . In a perspective distorted image, these two sets of parallel lines will intersect at points  $\mathbf{p}^{(1)}$  and  $\mathbf{p}^{(2)}$ . The line formed by connecting  $\mathbf{p}^{(1)}$  and  $\mathbf{p}^{(2)}$  is the vanishing line  $\mathbf{l} = (l_1, l_2, l_3)^T$ , in homogeneous coordinates:

$$\begin{aligned}\mathbf{p}^{(1)} &= \mathbf{l}^{(1)} \times \mathbf{l}^{(2)} \\ \mathbf{p}^{(2)} &= \mathbf{m}^{(1)} \times \mathbf{m}^{(2)} \\ \mathbf{l} &= \mathbf{p}^{(1)} \times \mathbf{p}^{(2)}\end{aligned}$$

Therefore, by applying transformation  $\mathbf{H}_1$

$$\mathbf{H}_1 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ l_1 & l_2 & l_3 \end{pmatrix}$$

the vanishing line will be mapped into the line at infinity  $\mathbf{l}^\infty = (0, 0, 1)^T$ . This can be verified as

$$\mathbf{H}_1^{-T} = \begin{pmatrix} 1 & 0 & -l_1/l_3 \\ 0 & 1 & -l_2/l_3 \\ 0 & 0 & 1/l_3 \end{pmatrix}$$

and  $\mathbf{H}_1^{-T}\mathbf{l} = (0, 0, 1)^T$ , applying  $\mathbf{H}_1$  to camera images we obtain the affinely rectified image  $\mathbf{X}_a = \mathbf{H}_1\mathbf{X}_c$ .

### 9.3.3. Affine distortion removal

If we obtain the affinely rectified image  $\mathbf{X}_a$ , we should find the affine transformation matrix

$$\mathbf{H}_2 = \begin{pmatrix} \mathbf{A} & \mathbf{t} \\ \mathbf{0} & 1 \end{pmatrix}$$

such that  $\mathbf{X}_a = \mathbf{H}_2\mathbf{X}_s$ , where  $\mathbf{X}_s$  is the scene image in the real world. In the target, we have a pair of real orthogonal lines,  $\mathbf{l} \perp \mathbf{m}$  and  $\mathbf{l}' \perp \mathbf{m}'$  are the transformed lines under affine transformation  $\mathbf{H}_2$ . By orthogonality, we know that

$$(l_1/l_3, l_2/l_3)(m_1/m_3, m_2/m_3)^T = 0$$

then

$$l_1m_1 + l_2m_2 = \mathbf{l}^T \mathbf{C}_\infty^* \mathbf{m} = 0$$

where  $\mathbf{C}_\infty^*$  is the dual degenerate conic. Since  $\mathbf{C}_\infty^{*\prime} = \mathbf{H}_2 \mathbf{C}_\infty^* \mathbf{H}_2^T$ , we have

$$\mathbf{l}'^T \mathbf{C}_\infty^* \mathbf{m} = \mathbf{l}'^T \mathbf{H}_2 \mathbf{H}_2^{-1} \mathbf{C}'_\infty^* \mathbf{H}_2^{-T} \mathbf{H}_2^T \mathbf{m}' = \mathbf{l}'^T \mathbf{C}'_\infty^* \mathbf{m}' = 0$$

Therefore,

$$\begin{aligned} \mathbf{l}'^T \mathbf{C}'_\infty^* \mathbf{m}' &= \mathbf{l}'^T \mathbf{H}_2 \mathbf{C}_\infty^* \mathbf{H}_2^T \mathbf{m}' \\ &= \mathbf{l}'^T \begin{pmatrix} \mathbf{A} & \mathbf{t} \\ \mathbf{0} & 1 \end{pmatrix} \begin{pmatrix} I & \mathbf{0} \\ \mathbf{0} & 0 \end{pmatrix} \begin{pmatrix} \mathbf{A}^T & \mathbf{0} \\ \mathbf{t}^T & 1 \end{pmatrix} \mathbf{m}' \\ &= \mathbf{l}'^T \begin{pmatrix} \mathbf{A}\mathbf{A}^T & \mathbf{0} \\ \mathbf{0} & 0 \end{pmatrix} \mathbf{m}' \end{aligned}$$

We have

$$(l'_1, l'_2) \mathbf{A}\mathbf{A}^T (m'_1, m'_2)^T = 0 \quad [9.21]$$

In order to obtain  $\mathbf{A}$ , let  $\mathbf{S} = \mathbf{A}\mathbf{A}^T$ , where

$$\mathbf{S} = \begin{pmatrix} S_{11} & S_{12} \\ S_{12} & 1 \end{pmatrix} \quad [9.22]$$

since  $\mathbf{S}$  is a symmetric matrix considering the scale problem. Introducing [9.22] into [9.21], we get:

$$(l'_1 m'_1, l'_1 m'_2 + l'_2 m'_1) \begin{pmatrix} s_{11} \\ s_{12} \end{pmatrix} = -l'_2 m'_2 \quad [9.23]$$

From [9.23], it is clear that in order to solve the matrix  $S$  with two unknown parameters, two pairs of orthogonal lines are required  $\mathbf{l}^{(1)} \perp \mathbf{m}^{(1)}$  and  $\mathbf{l}^{(2)} \perp \mathbf{m}^{(2)}$ .

Since the symmetric matrix  $\mathbf{S}$  can be written as  $\mathbf{S} = \mathbf{U}\mathbf{D}\mathbf{U}^T$ , where  $\mathbf{U}^{-1} = \mathbf{U}^T$ , we can get  $\mathbf{A} = \mathbf{U}\sqrt{\mathbf{D}}\mathbf{U}^T$ . Now, we know  $\mathbf{H}_2$  and the restored image is  $\mathbf{X} = \mathbf{H}_2^{-1} X_a$

#### 9.4. Tracking of a visual target

Computer vision plays an important role in a wide variety of applications such as text finding, color image segmentation [STE 03], robotics [GRE 04, ZHA 05], etc. In this section, we propose a computer vision algorithm to obtain the position of the miniature quadrotor with respect to a visual target which in our case is a  $28 \times 28$  cm red square target in real-time test. The algorithm proposed works in a 3D Euclidean space, in which measurements of lengths and angles do make sense [FAU 95]. We use a color extractor such that the detection of the target from the image is robust to noise and changes in luminance conditions. Using color segmentation, we avoid the problems caused by motion blur [BAR 04] in the image. In particular, we search the corner of our target and the image blur does not affect our estimation. Then, an edge-detector algorithm is used to find the edges of the target in the image plane which is described next.

#### 9.4.1. Edge-detection algorithm

The Canny edge-detection algorithm [CAN 86] produces thin fragments of image contours that can be controlled by a single smoothing parameter known as  $\sigma$ . The image is first smoothed with a Gaussian filter of spread  $\sigma$  and then the gradient magnitude and direction are computed at each pixel of the smoothed image. Gradient direction is used to thin the edges by suppressing any pixel response that is not higher than the two neighboring pixels on either side of it along the direction of the gradient, this is called non-maximum suppression. Once the gradient magnitudes are thinned, high-magnitude contours are tracked. In the final aggregation phase, continuous contour segments are sequentially followed. Contour following is initiated only on edge pixels where the gradient magnitude meets a high threshold; however, once started, a contour may be followed through pixels whose gradient magnitude meet a lower threshold, usually about half of the higher starting threshold. After the edges are known, image regions can be detected from the segmented image. An image region is detected when edge segments or boundary segments close on themselves forming a convex region. In particular, a rectangular object results in four straight line segments, connected such that they form a convex region. So far we have settled the basis for image region detection. We next describe the polygons obtained after color- and edge-detection algorithms, and some useful polygon properties.

#### 9.4.2. Polygons properties

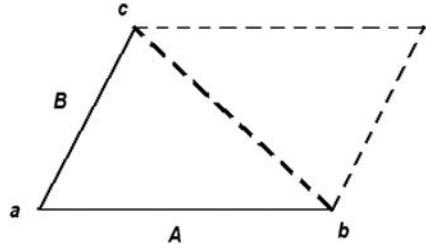
We define a polygon as a region of a plane bounded by a finite collection of line segments forming a simple closed curve, let  $v_0, v_1, v_2, \dots, v_{n-1}$  be  $n$  points in the plane. Let  $e_0 = v_0v_1, e_1 = v_1v_2, \dots, e_i = v_iv_{i+1}, \dots, e_{n-1} = v_{n-1}v_0$  be  $n$  segments connecting the points. Then these segments bound a convex polygon iff:

- 1) The intersection of each pair of adjacent segments in the cyclic ordering is the single point shared between them:  $e_i \cap e_{i+1} = v_{i+1}$ , for all  $i = 0, \dots, n - 1$ .
- 2) Non-adjacent segments do not intersect: i.e.  $e_i \cap e_j = \emptyset$ , for all  $j \neq i + 1$ .

The points  $v_i$  are called the vertices of the polygon and the segments  $e_i$  are called its edges. Note that a polygon of  $n$  vertices has  $n$  edges.

The area of a polygon with vertices  $v_0, v_1, \dots, v_{n-1}$  labeled counterclockwise can be calculated as  $A_P = A(v_0, v_1, v_2) + A(v_0, v_2, v_3) + \dots + A(v_0, v_{n-2}, v_{n-1})$ .

From linear algebra, we know that the magnitude of the cross product of two vectors is the area of the parallelogram they determine. If  $A$  and  $B$  are vectors with coordinates  $(A_x, A_y, A_z)$  and  $(B_x, B_y, B_z)$  in the  $x, y$ , and  $z$  directions, respectively, then  $|A \times B|$  is the area of the parallelogram with sides  $A$  and  $B$ , as shown in Figure 9.3.



**Figure 9.3.** Cross product parallelogram

For two-dimensional vectors,  $A_z = B_z = 0$ , the cross product reduces to  $(A_0B_1 - A_1B_0)$  and the area of a triangle  $A_T$  is given by:

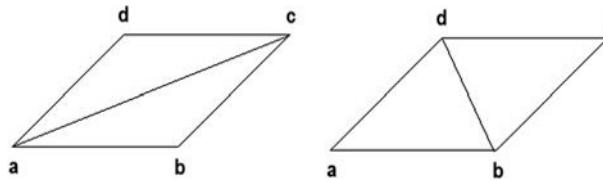
$$A_T = \frac{1}{2}(A_x B_y - A_y B_x) \quad [9.24]$$

Note that  $A = b - a$  and  $B = c - a$ , where  $(a, b, c)$  are three arbitrary vertices, then [9.24] can be rewritten as:

$$\begin{aligned} 2A_T &= a_x b_y - a_y b_x + a_y c_x - a_x c_y + b_x c_y - c_x b_y \\ &= (b_x - a_x)(c_y - a_y) - (b_y - a_y)(c_x - a_x) \end{aligned} \quad [9.25]$$

This is an expression for the area of the triangle as a function of the coordinates of its vertices. The area of a convex quadrilateral  $Q = (a, b, c, d)$  may be written in two ways, depending on the two different triangulations, as shown in Figure 9.4.

$$A_Q = A(a, b, c) + A(a, c, d) = A(d, a, b) + A(d, b, c) \quad [9.26]$$



**Figure 9.4.** The two triangulation of a convex quadrilateral

Writing out the expressions for the areas using equation [9.26] for the two terms of the first triangulation, we get

$$\begin{aligned} 2A_Q &= a_x b_y - a_y b_x + a_y c_x - a_x c_y + b_x c_y - c_x b_y \\ &\quad + a_x c_y - a_y c_x + a_y d_x - a_x d_y + c_x d_y - d_x c_y \end{aligned} \quad [9.27]$$

#### 9.4.3. Square-detection algorithm

Once we have obtained four edge segments in the image plane, which intersect at four points, it is required to verify that the segments form a square. To verify this, the angle in each corner of the intersection of two edges is computed. By applying the triangulation of polygons, it is always possible to obtain two similar triangles from a square. Then given the triangle  $\Delta$  with edges  $A, B$ , and  $C$ , the angle  $\varphi$  between the two edges  $B$  and  $C$  can be obtained by the following formula:

$$\varphi = \arccos \frac{|B|^2 + |C|^2 - |A|^2}{2 \cdot |B| \cdot |C|} \quad [9.28]$$

where  $|A|, |B|$ , and  $|C|$  denote the lengths of the edges  $A, B$ , and  $C$ , respectively. Applying this formula at each vertex of the square, we obtain the four angles in the polygon. If these edges are perpendicular, we then verify the conditions for a convex region to be a square and its area can be obtained using formula [9.27]. Once we have located the target on the image plane, we are interested in computing the distance to it and the translational velocity with respect to the target. Estimation of the distance to the target needs full knowledge of the camera parameters. The translational velocity is computed using an OF approach. The camera model and the OF estimation are described in the following sections.

#### 9.4.4. Image rectification

The square target has four points named  $P_1, P_2, P_3$ , and  $P_4$ . The vector from the origin of the world coordinate system to the point  $P_i$  is labeled  $\mathbf{P}_i$ . The Euclidean distance between  $P_i$  and  $P_j$  is denoted by  $s_{ij}$ . The vector  $(\mathbf{P}_i - \mathbf{P}_j)$  is denoted by  $\mathbf{P}_{ij}$  for  $i, j = 1, \dots, 4, i \neq j$ . The areas  $A(P_i, P_j, P_k)$  of the triangles in this square can be computed as:

$$A_1 = A(P_2, P_1, P_4) = |\mathbf{P}_{21} \times \mathbf{P}_{24}| / 2 = A(P_2, P_3, P_4) = |\mathbf{P}_{23} \times \mathbf{P}_{24}| / 2.$$

$$A_2 = A(P_1, P_4, P_3) = |\mathbf{P}_{14} \times \mathbf{P}_{13}| / 2 = A(P_1, P_2, P_3) = |\mathbf{P}_{12} \times \mathbf{P}_{13}| / 2.$$

We call  $\bar{\mathbf{p}}_i^f$  the vector from the center of projection  $C$  to the point  $p_i$  in the image plane and the vector joining  $C$  to  $P_i$  is denoted by  $\bar{\mathbf{P}}_i^f$ . The unit vector collinear to  $\bar{\mathbf{P}}_i^f = (x_i, y_i, f)^t$  is given by:

$$\mathbf{u}_i = (u_{ix}, u_{iy}, u_{iz})^t = \bar{\mathbf{p}}_i^f / \|\bar{\mathbf{p}}_i^f\| = (x_i, y_i, f)^t / F_i,$$

where

$$F_i = \sqrt{x_i^2 + y_i^2 + f^2}$$

Using the image coordinates of the points  $p_i$ , we can find the area of the triangle in the image plane using

$$\begin{aligned} B_1 &= x_2(y_4 - y_1) + y_2(x_1 - x_4) + y_1x_4 - x_1y_4 \\ B_2 &= x_2(y_3 - y_1) + y_2(x_1 - x_3) + y_1x_3 - x_1y_3 \end{aligned}$$

The  $B_i$ 's are equal to twice the area of the triangle formed by three of the four image points corresponding to the subscripts of the coordinates. If the optical axis of the vision system is normal to the target plane, the target and its image are similar, i.e.

$$\begin{aligned} \frac{r_{12}}{s_{12}} &= \frac{r_{13}}{s_{13}} = \frac{r_{14}}{s_{14}} = \frac{r_{23}}{s_{23}} = \frac{r_{24}}{s_{24}} = \frac{r_{34}}{s_{34}} = \frac{f}{f - Z_W}, \\ \frac{B_1}{A_1} &= \frac{B_2}{A_2}, \\ Z_1^c &= Z_2^c = Z_3^c = Z_4^c = Z_W. \end{aligned}$$

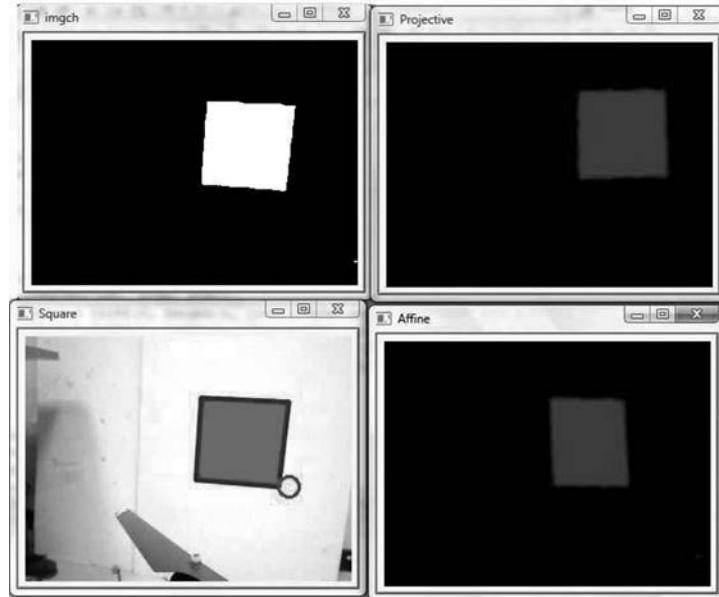
The parameter  $r_{ij}$  denotes the distance between  $p_i$  and  $p_j$ . We have assumed that the vision system is always normal to the target plane; i.e. we have neglected projective and affine distortions from camera images. In our approach, the projective distortion was removed first and then the affine distortions were removed, as explained in previous sections (see Figure 9.5).

#### 9.4.5. Solving the 3D localization problem

In order to estimate the miniature quadrotor position and velocity, we propose to use two cameras orthogonally mounted on the miniature quadrotor structure. The miniature quadrotor position is obtained as follows. In real experience, we use a color segmentation carried out on the original RGB image taken by the frontal camera. Next, the Canny algorithm is applied for edge detection to obtain the boundaries of the regions of interest. The boundaries are formed by a finite collection of  $n$  line segments  $e_0 = v_0v_1, e_1 = v_1v_2, \dots, e_i = v_iv_{i+1}, \dots, e_{n-1} = v_{n-1}v_0$ , connected by  $n$  image points  $v_0, v_1, v_2, \dots, v_{n-1}$ , where  $v_i = (x_i, y_i)$ . Using these points, we can obtain four vertices and four edges. The next step is to verify the angle on each corner of the quadrilateral using [9.28]. If this quadrilateral is a square, it is necessary to compute the square area using [9.24]. Once the area of the square is known, and the intrinsic parameters values of the camera [9.20] are also known, we use similar triangles' properties to obtain the distance  $Z_W$  between the camera and the target. Using [9.20], we know that  $f_x = fS_x$  and  $f_y = fS_y/\sin\eta$ , then  $S_x = f_x/f$  and  $S_y = f_y \sin\eta/f$ , the four 3D points of our target are called  $P_1, P_2, P_3$ , and  $P_4$  in the clockwise sense. These points have corresponding points  $p_1 = (x_i, y_i)$ ,  $p_2 = (x_{i+1}, y_i)$ ,  $p_3 = (x_{i+1}, y_{i+1})$ , and  $p_4 = (x_i, y_{i+1})$  in image coordinates.

Assuming the knowledge of  $h$ , the height of the square target, and the focal length  $f$ , we can obtain:

$$Z = \frac{fh}{(y_i - y_{i+1})S_y} \quad [9.29]$$



**Figure 9.5.** Several steps to remove projective and affine distortion. Bottom-left image displays original image taken by frontal camera. Top-left image shows output image after color filter method. Top-right image displays the effect of removing projective distortion. Bottom-right image presents the result of removing affine distortion

Other information can also be obtained such as the central point of the square, which is equal to the point where the two diagonals of the square intersect, called  $(x_{ci}, y_{ci})$  in the image plane. Using perspective projection properties in frontal pinhole camera model, we obtain:

$$X = \frac{x_{ci}Z}{f} \quad [9.30]$$

and

$$Y = \frac{y_{ci}Z}{f} \quad [9.31]$$

The miniature quadrotor translational velocity is estimated by using a Lucas–Kanade OF algorithm on the camera that is pointing downward. The proposed position algorithm estimates the 3D position of miniature quadrotor  $\mathbf{p}_{CM}^n = [x \ y \ z]^T = [X \ Z \ Y]^T$ .

#### 9.4.6. OF measurement

We use the OF from image sequences to estimate the translational velocity of the quadrotor. The estimate of the horizontal velocity will be used in the closed-loop feedback control to perform autonomous hover flights. In [MUR 05], the rotational components of OF are compensated and Stowers *et al.* [STO 09] use an OF technique of phase correlation on images captured with a down-facing camera which is fitted with an omnidirectional lens and the images are transformed into the logpolar domain before the main computational step. There exist many different methods for computing the OF, such as intensity-based differential methods [BEA 95], frequency-based filtering, and correlation-based methods. In this chapter, we have used the Lucas–Kanade pyramidal method, which is an intensity-based differential method [BOU 99]. This method has been chosen for its temporal persistence. To circumvent the problem of large and non-coherent motions, we can first track over larger spatial scales using an image pyramid and then refine the initial motion velocity assumptions by working our way down the levels of the image pyramid until we arrive at the raw image pixels. Furthermore, due to spatial coherence, the neighboring points in a scene on the floor belong to the same surface. They also have similar motion and project onto nearby points on the image plane. The clear advantage of this method is that each residual OF vector  $\mathbf{d}^L$  can be kept very small while computing a large overall pixel displacement vector  $\mathbf{d}$ . Assuming that each elementary OF computation step can handle pixel motions up to  $d_{\max}$ , then the overall pixel motion that the pyramidal implementation can handle becomes  $d_{\max-\text{final}} = (2^{Lm+1} - 1)d_{\max}$ .

Consider two discrete functions  $I_1, I_2 \in R^{m_x \times n_y}$  representing two gray scale images at different time instants, and let  $G_{p_i}$  be the gray-scale value of a particular pixel  $p = (x_i, y_i)^T$ . Then, the gray-scale values for  $p_i$  which appear in two consecutive images are defined, respectively, as

$$G_{p_1} = I_1(x_1, y_1), \quad G_{p_2} = I_2(x_2, y_2)$$

where  $x_i$  and  $y_i$  are the row and column pixel coordinates, respectively. Given a specific image point  $p_1 \in I_1$ , the aim of the approach is to find another image point  $p_2 \in I_2$  such that  $G_{p_1} \approx G_{p_2}$ . Moreover, the relationship between matched pixels  $p_1$  and  $p_2$  is given by

$$p_2 = p_1 + r = [x_1 + r_x \quad y_1 + r_y]^T$$

where  $r = [r_x \ r_y]^T$  defines the image displacement or OF and minimizes the following residual function:

$$\epsilon(r) = \sum_{x_p=x_{p1}-w_x}^{x_{p1}+w_x} \sum_{y_p=y_{p2}-w_y}^{y_{p2}+w_y} (I_1(p_1) - I_2(p_1 + r))^2$$

where  $w_x$  and  $w_y$  are two integers that define the size of the integration window. The Lucas–Kanade OF algorithm has an adaptive integration window; it means that it is able to handle large pixel motions and this representation acts as a low-pass filter. Figure 9.6 shows the OF algorithm using a windows size  $5 \times 5$ .



**Figure 9.6.** OF measurement

Let us assume that the camera is moving with translational velocity  $v$  and angular velocity  $\omega$  while viewing an object at distance  $d$  and offset  $\beta$  from the direction of travel, as depicted in Figure 9.7. The OF can be expressed as:

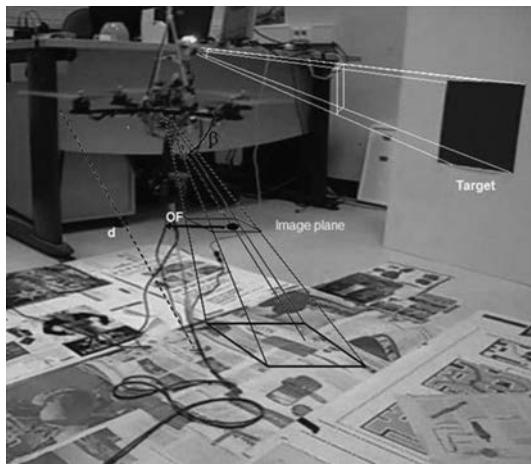
$$OF = \frac{v}{d} \sin \beta - \omega \quad [9.32]$$

The maximal OF is obviously generated when the plane that contains the features is perpendicular to translational motion direction ( $\beta = 90^\circ$ ). The velocity can be estimated by [9.32] as follows:

$$v = \frac{(OF + \omega)}{\sin \beta} d$$

Notice that singularities in the above equation appear when  $\beta = 0^\circ$ . Nevertheless in our case the pitch and roll angles are very close to zero which implies that

$\beta = 90^\circ \pm 3^\circ$  is a maximal interval when the quadrotor is appropriately stabilized at hover. The measurement of the angular velocity  $\omega$  is obtained by using the gyro information on each axis. An altitude stabilization algorithm in closed loop is used to keep the distance  $d$  constant and equal to some desired value. Since the IMU's sampling period is smaller ( $\sim 10$  ms) than the OF sampling period ( $\sim 60$  ms), a zero-order holder on the OF has been used in order to synchronize the estimation of the translational velocity. Note that no data fusion between the IMU and the visual system has been implemented.



**Figure 9.7.** Quadrotor vision system using two cameras

## 9.5. Tracking of a visual line

Visual tracking is a very important notion in the world of quadrotor which gives it a precise navigation scheme. In real life, some quadrotor tasks include tube verification, walls exploration, building exploration, and power lines verification; these tasks can be risky for human operators. To prevent all human risks, we propose a visual system capable of using the parallelism in this structure to navigate and explore these human-made constructions.

### 9.5.1. Vanishing point detection

Some objects in 3D space can be expressed in terms of parallelism, orthogonality, and coplanarity; these constraints can be used to obtain three VPs in the image plane that correspond to three mutually orthogonal directions in space. Using these VPs it is possible to obtain the intrinsic and extrinsic parameters of a camera.

In this section, we propose to use the VP detection technique to estimate the rotation matrix  $\mathbf{R}$  and translation vector  $\mathbf{t}$  of the camera by scaled orthographic projection and perspective projection. A VP in a perspective geometry is a single point where all the projected parallel segment lines in the 3D world, which are projected to non-parallel lines in the image plane, converge. Notice that the VPs can be found outside the field of view of the camera. If a set of lines is parallel to one of the three principal axes ( $x$ ,  $y$ , or  $z$ ), then the point is called a principal VP. In the 3D world, there exist three principal VPs, one for each axis named  $V_x$ ,  $V_y$ , and  $V_z$ . Computing one or more of these VPs from an image, helps us to obtain an estimation of image depth, object dimension, and 3D structure [WAN 08b]. This approach will be described in the following sections and some assumptions are considered:

- 1) The target is located in front of the camera, i.e. in the field of view of the camera.
- 2) To simplify the analysis, the target will be a rectangle with vertices  $(A, B, C, D)$  in a 3D world.
- 3) In the image plane, this rectangle is seen like a four-sided polygon with vertices  $(a, b, c, d)$  (see Figure 9.8).

Knowing the coordinates of each vertex in the image plane, we define

- $\vec{l}^1$  as the projected lines formed by the pair of vertex  $(b, c)$ ;
- $\vec{l}^2$ , is the projected line formed by the pair of vertex  $(a, d)$ ;
- $\vec{m}^1$  is the projected line formed by the pair of vertex  $(a, b)$ ; and
- $\vec{m}^2$  is the projected line formed by the pair of vertex  $(c, d)$ .

Let  $V_x$  be a VP obtained for intersecting lines  $\vec{l}^1$  and  $\vec{l}^2$  similarly,  $V_y$  is a VP obtained for intersecting lines  $\vec{m}^1$  and  $\vec{m}^2$  (see Figure 9.8).

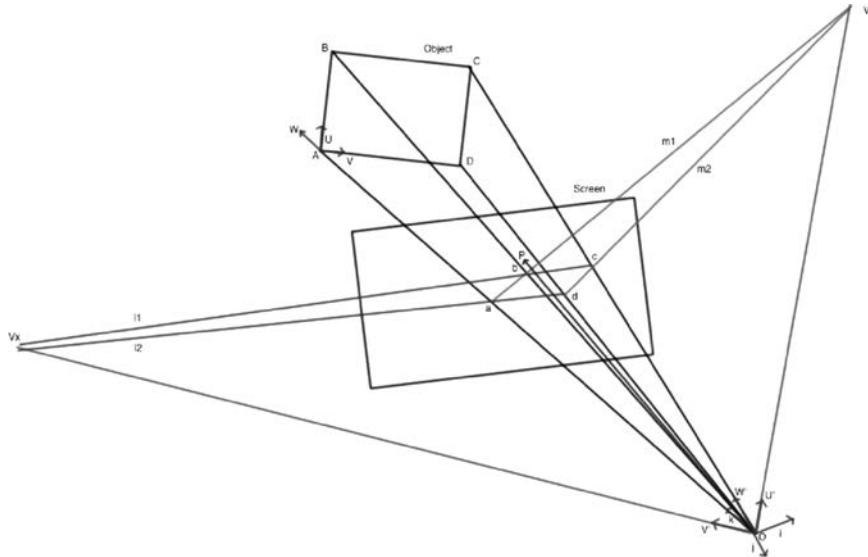
Once the intrinsic calibration parameters are known, in order to obtain the external calibration parameters all we need is the VPs corresponding to any two axes of the reference of camera coordinate system. The other axis is simply the cross product of the two. In mathematical form, it yields

$$V_x = \vec{l}^1 \times \vec{l}^2$$

$$V_y = \vec{m}^1 \times \vec{m}^2$$

and where  $V_x = [V_{xx}, V_{xy}, f]^T / \sqrt{V_{xx}^2 + V_{xy}^2 + f^2}$  and  $V_y = [V_{yx}, V_{yy}, f]^T / \sqrt{V_{yx}^2 + V_{yy}^2 + f^2}$ , and  $f$  is the focal length. Knowing that  $V_x$  and  $V_y$  are orthogonal, then the VP  $V_z$  over  $z$ -axis of the camera is obtained by:

$$V_z = V_x \times V_y \quad [9.33]$$



**Figure 9.8.** VP in a four-sided polygon

The rotation matrix which describes the rigid motion between the world and the camera coordinate system is:

$$\mathbf{R} = [V_x \quad V_y \quad V_z] \quad [9.34]$$

Observe that, in order to obtain the VPs, every image needs to be preprocessed. The pretreatment of the image involves a color-segmentation process to cancel the image blur and a boundary-extraction process to locate, in the image plane, four coplanar points which are the corners of a four-sided polygon.

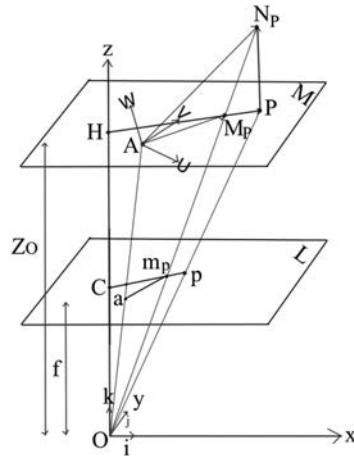
Using the classic pinhole camera model, with its center of projection  $O$ , its image plane  $L$  is at a distance  $f$  from  $O$ , with its axis  $O_x$  and  $O_y$  pointing along the row and the columns of the camera sensor and  $O_z$  pointing along the optical axis. The unit vectors for these three axes are  $\vec{i}$ ,  $\vec{j}$ , and  $\vec{k}$  with the coordinate system  $R_c = (O, \vec{i}, \vec{j}, \vec{k})$ . The 2D coordinate system associated with the screen plane can be denoted  $R_s = (P, \vec{i}, \vec{j})$ . For calibration operation, we know  $f$  and the intersection of the optical axis with the image plane.

An object with characteristic points  $A, B, C, D$  is located in the field of view of the camera, in a plane  $M$  at a distance  $Z_A$  from the center of projection  $O$ , its coordinates system is  $R_o$  with a reference point  $A$  for the object, and we can write  $R_o = (A, U, V, W)$ . Every point has a coordinate in the object coordinate frame  $B = (U_B, V_B, W_B)$ ,  $C = (U_C, V_C, W_C)$ , and  $D = (U_D, V_D, W_D)$ , with image points  $a, b, c, d$ . Each image point has a known image coordinate

$p_i = (x_i, y_i) \forall i = A, B, C, D$  in the image plane. The coordinates  $(X_i, Y_i, Z_i)$  of the points  $A, B, C, D$  are unknown, because the pose of the object in the camera coordinate system is unknown. Note that the depths  $Z_i$  of different points are not very different from one other, and can all be set to the depth  $Z_A$  of the reference point  $A$  of the object. The coordinates  $(x'_i, y'_i)$  of the points  $p'_i$  are the scaled orthographic projections of the feature points  $P \in [B, C, D]$ , which are called  $N_P$ . Knowing the height of the line  $\ell$  and the focal length  $f$ , we can calculate  $Z_A$  using triangles properties.

Consider points  $A, N_P$  (see Figure 9.9) of the object and the plane  $M$  parallel to the image plane  $L$  through the reference point  $A$ . The line of sight for  $N_P$  intersects plane  $M$  in  $M_P$  and  $N_P$  projects onto plane  $M$  in  $P$ . The vector  $\vec{AN}_P$  is equal to:

$$\vec{AN}_P = \vec{AM}_P + \vec{M}_P \vec{P} + \vec{PN}_P \quad [9.35]$$



**Figure 9.9.** Perspective projection  $m_P$  and scaled orthographic projection  $p$  for an object  $N_P$  and a reference point  $A$

The vector  $\vec{AN}_P$  and its image  $\vec{am}_P$  are proportional in the ratio  $Z_A/f$ . The vectors  $\vec{M}_P \vec{P}$  and  $\vec{Cm}_P$  are also proportional in the two similar triangles  $Cm_P O$  and  $M_P P N_P$ , in a ratio equal to the ratio of the  $z$  coordinates of the other two corresponding vectors of these triangles  $\vec{PN}_P$  and  $\vec{OC}$ . This ratio is  $\vec{PN}_P \cdot \vec{k}/f$ . The sum of the three vectors can be expressed as:

$$\vec{AN}_P = \frac{Z_A}{f} \vec{am}_P + \vec{PN}_P \cdot \frac{\vec{k}}{f} \vec{Cm}_P + \vec{PN}_P \quad [9.36]$$

We take the dot product of this expression with the unit vector  $\vec{i}$  of the camera coordinate system. The dot product  $\vec{PN}_P \cdot \vec{i} = 0$ , and the dot product  $\vec{am}_P \cdot \vec{i}$  is the  $x$ -coordinate,  $x_P - x_A$ , of the vector  $\vec{x}_A \vec{x}_P$ . The dot product  $\vec{Cm}_P \cdot \vec{i}$  is the coordinate  $x_P$  of  $\vec{Cm}_P$ . Defining  $\beta_P = 1/Z_A \vec{AP} \cdot R_3$  as the  $z$ -coordinate of  $\vec{AP}$ , and  $R_3 = R_1 \times R_2$ , where  $R_j$  is the  $j$ th row of rotation matrix  $\mathbf{R}$ , substituting, we have:

$$\vec{AP} \cdot \frac{f}{Z_A} R_1 = x_P(1 + \beta_P) - x_A \quad [9.37]$$

Similarly, we obtain the following equation:

$$\vec{AP} \cdot \frac{f}{Z_A} R_2 = y_P(1 + \beta_P) - y_A \quad [9.38]$$

taking the dot product of [9.36] with the unit vector  $\vec{j}$ .

Now, consider the points  $A, N_P$  (see Figure 9.9), the projection  $P$  of  $N_P$  on the plane  $M$ , and its image  $p'_P$  with coordinates  $(x'_P, y'_P)$ . In scaled orthographic projection, the vector  $\vec{AN}_P$  is the sum of two vectors

$$\vec{AN}_P = \vec{AP} + \vec{PN}_P \quad [9.39]$$

The vector  $\vec{AP}$  and its image  $\vec{ap}_P$  are proportional in the ratio  $Z_A/f$ . Consequently,

$$\vec{AN}_P = \frac{Z_A}{f} \vec{ap}_P + \vec{PN}_P \quad [9.40]$$

We take the dot product of this vector with unit vector  $\vec{i}$  of the camera coordinate system; the dot product  $\vec{PN}_P \cdot \vec{i} = 0$ , and the dot product  $\vec{ap}_P \cdot \vec{i}$  is the  $x$  coordinate,  $x'_P - x_A$ , of the vector  $\vec{ap}_P$ . We obtain

$$\vec{AP} \cdot \frac{f}{Z_A} R_1 = x'_P - x_A \quad [9.41]$$

$$\vec{AP} \cdot \frac{f}{Z_A} R_2 = y'_P - y_A \quad [9.42]$$

Notice that the coordinates of  $p_P$  can be written as:

$$x'_P = x_P(1 + \beta_P) \quad [9.43]$$

$$y'_P = y_P(1 + \beta_P) \quad [9.44]$$

Knowing that  $A, B, C, D$  are coplanar, we must use  $R_1 \cdot R_2 = 0$ . The translation vector  $\mathbf{T}$  is aligned with vector  $OA$  between the center of projection  $O$  and the

reference point  $A$ . Since the image of  $A$  is the known image point  $a$  then  $\mathbf{T}$  is aligned with vector  $Oa$  and is equal to  $(Z_A/f)Oa$ . Then, we get:

$$X_A = \frac{Z_A}{f}x_A \quad [9.45]$$

$$Y_A = \frac{Z_A}{f}y_A \quad [9.46]$$

## 9.6. Embedded architecture

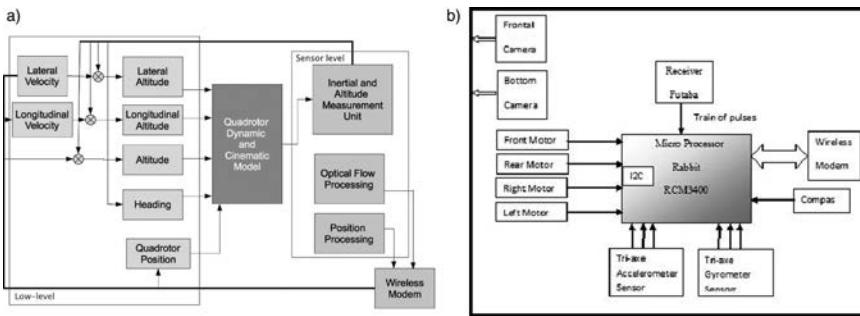
The miniature quadrotor is powered by four brushless motors with fixed pitch propellers. Altitude is controlled by the collective thrust of the motors. The total weight of the vehicle is about 380 g, with a flight endurance of 10 min approximately. Theoretical results obtained were incorporated into an autopilot control system using an architecture based on a 29-MHz Rabbit microcontroller with 512 kB Flash and 512 kB RAM. These microcontrollers are capable of handling floating-point operations and multitasking processing virtually due to the enhancement compiler Dynamic C [RAB]. We have built our own inertial measurement unit (IMU) using accelerometers, gyros, and a compass to obtain the roll, pitch, and yaw angles, and angular rates. The IMU information is sent to the microcontroller which also reads reference control inputs from an operator through a serial wireless modem. The microcontroller subsequently combines this information to compute the control input and sends the control corrections to the motors through a I2C serial port. Hence, the brushless motor speed controller or booster is handled by the I2C port of the microcontroller. Visual sensor is composed of two web cameras placed orthogonally in the quadrotor, with a camera pointing downward to process an OF algorithm to obtain the velocity estimation, and the other camera placed in front of the X4 looking forward, to get images of a line painted on a wall in front of the quadrotor vehicle. The images coming from the frontal camera are used to estimate the 3D position of the quadrotor using VPs. The proposed vision algorithm processes images with a size of  $640 \times 480$  pixels. All the vision algorithms are executed in a ground station composed of a PC with Intel® Core 2 Duo processor 2.10 GHz using OpenCV libraries, with a frequency of 15 frames per second approximately. Figure 9.10 shows a block diagram of the basic architecture.

## 9.7. Experimental results

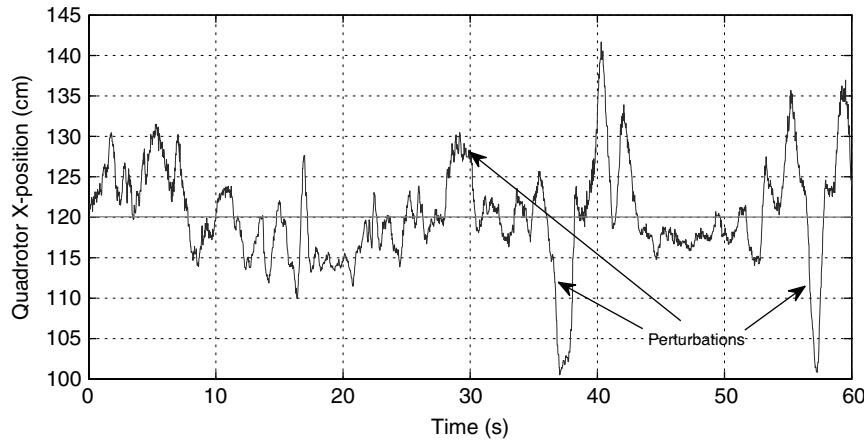
### 9.7.1. Visual target position stabilization

As was seen in previous sections, the control strategy for this vehicle is a state feedback control law where the angular position variables and its derivatives are obtained from the IMU. The position and velocity of the vehicle are estimated using

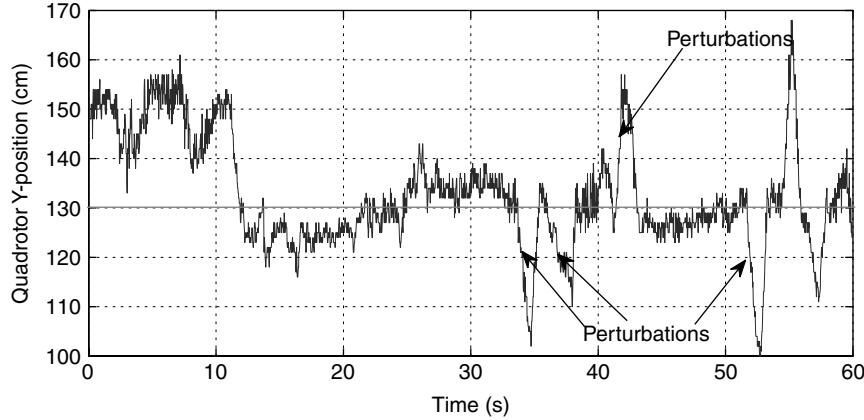
a vision-based algorithm explained in the previous sections. The closed-loop system includes a low-pass filter of the translational velocity and a saturation function to limit the translational velocity. We have modified the contrast of the image to obtain an improved estimation of characteristic points. In these experiments, the reference value for  $y^d = Z^d$  in frontal camera coordinates was set to 130 cm, which is the desired horizontal distance between the target and the miniature quadrotor vehicle. The lateral reference was set to  $x^d = X^d = 120$  cm, which is the desired displacement of the miniature quadrotor over the roll axis. The desired altitude was set to  $z^d = Y^d = 130$  cm. To adjust the control parameters several flight tests were carried out until a good performance of the vehicle is obtained. Figures 9.11–9.13 show the performance of the longitudinal and lateral subsystems using the nonlinear controller with separated saturations.



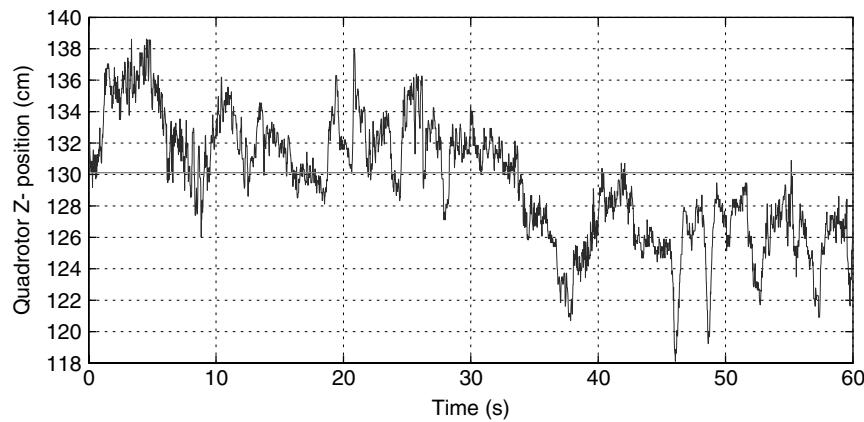
**Figure 9.10.** (a) Block diagram of the quadrotor controller; (b) embedded control architecture



**Figure 9.11.** X position of quadrotor. The quadrotor is manually perturbed over its x-axis



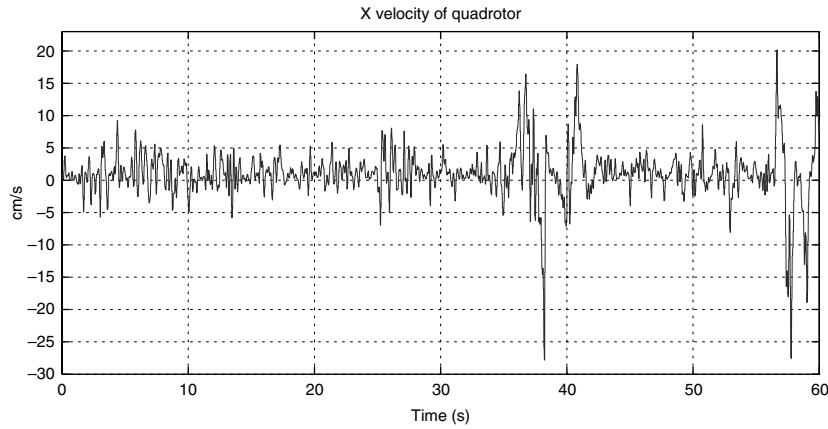
**Figure 9.12.** *Y position of quadrotor.  
The quadrotor is manually perturbed over its y-axis*



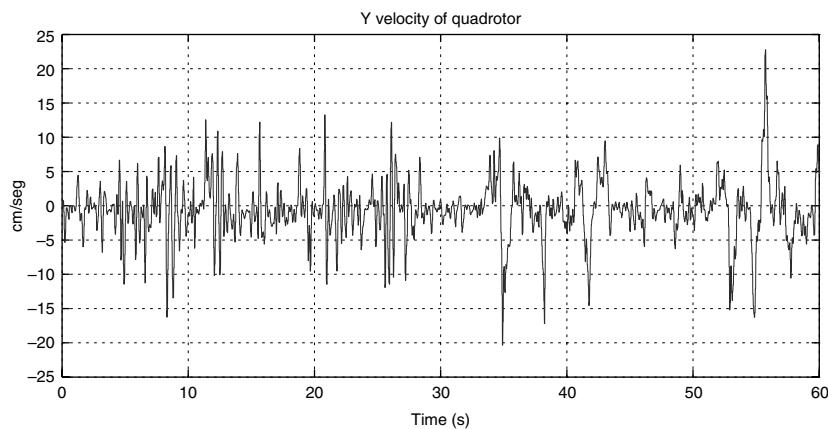
**Figure 9.13.** *Z position of quadrotor. Quadrotor altitude performance in presence of manual perturbations*

Several perturbations have been induced to test the robustness of our system to mild perturbations such as light winds and uncertainties due to the camera and the environment. The experimental validation of the proposed method can be seen in [GOMB] and [GOMa]. In Figures 9.11–9.13, a horizontal line represents the desired positions of quadrotor, and we show the result of our position algorithm in 60 s of hover flight. Figure 9.11 shows different disruptions caused manually over the  $x$ -axis of the quadrotor. Figure 9.12 presents different disruptions caused manually over the  $y$ -axis of the quadrotor. Figure 9.13 shows the quadrotor manually perturbed

in altitude. The velocities in the  $x$ -axis and  $y$ -axis of the miniature quadrotor were estimated using OF and are shown in Figures 9.14 and 9.15. Figure 9.14 shows the estimation velocity over the  $x$ -axis of the quadrotor. Figure 9.15 shows velocity estimation over miniature quadrotor  $y$ -axis.

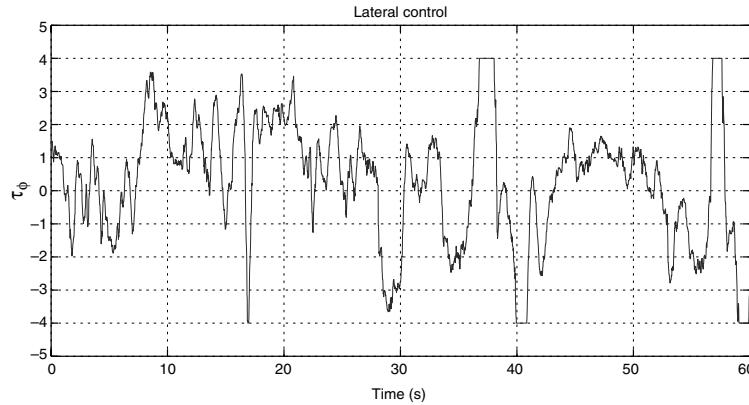


**Figure 9.14.** *X velocity of miniature quadrotor. The velocity estimations over  $x$ -axis are shown with some perturbations during quadrotor hover (cm/sec)*

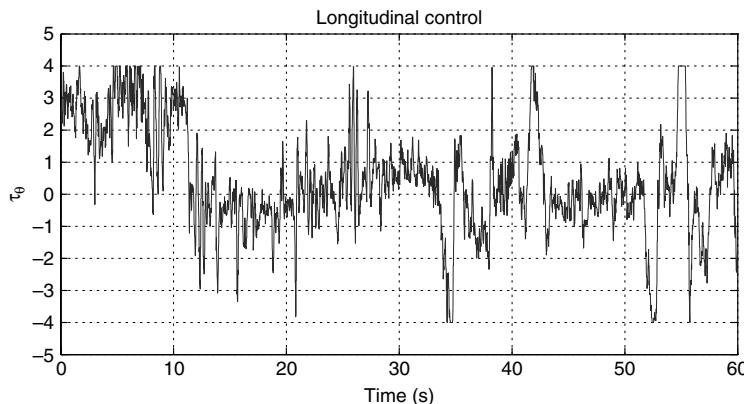


**Figure 9.15.** *Y velocity of miniature quadrotor. The velocity estimations over  $y$ -axis are shown with some perturbations during quadrotor hover (cm/sec)*

The pitch and roll angles are shown in Figures 9.16 and 9.17. This experiment shows that the nonlinear control law can recover from large orientation perturbations.



**Figure 9.16.** Lateral control  $\tau_\phi$  in presence of manual perturbations



**Figure 9.17.** Longitudinal control  $\tau_\theta$  in presence of manual perturbations

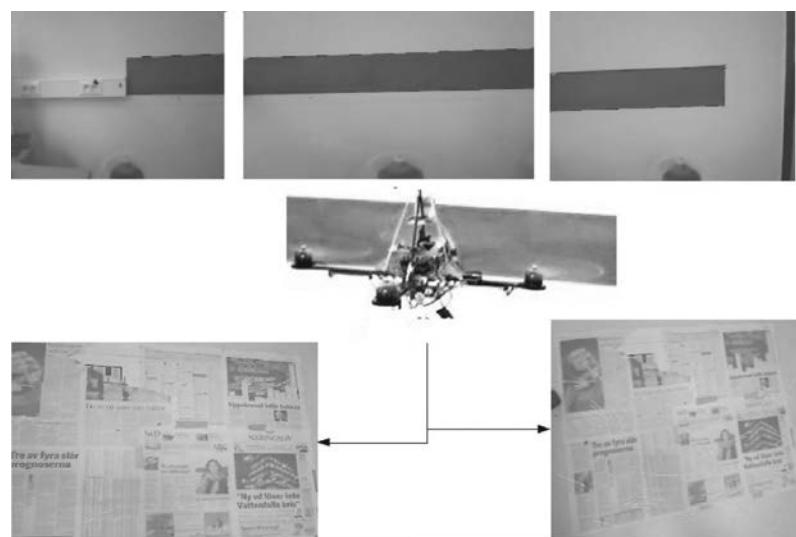
### 9.7.2. Tracking of a visual line with no marks

In order to test the proposed algorithm it was necessary to initialize the desired position values; in rotorcraft coordinates, the values are assigned in this manner:  $y_d = 130$  cm, which is the distance desired between the quadrotor and the target, and it remains constant in all the real tests. The desired altitude was set to  $z_d = 80$  cm. To displace the rotorcraft along the target line was necessary to have a variable displacement in the range [0, 230] cm. To estimate the velocity, the algorithm of Lucas–Kanade [BOU 99] was implemented in the camera mounted under the rotorcraft pointing downward. This algorithm finds a set of characteristic points in the ground and compare the displacement of this point, between two consecutive image frames.

The displacement over the  $x$ -axis was achieved by the information obtained from the frontal camera; this displacement has only two directions from left to right in the range  $[0, 230]$  that was the initial constraint and the displacement from right to left  $[230, 0]$ . To know what the direction of the displacement is, we create a boolean flag to indicate the direction of displacement; this flag changes its value once the quadrotor arrives at the range limit depending on the direction, indicating change of direction in the displacement of the quadrotor.

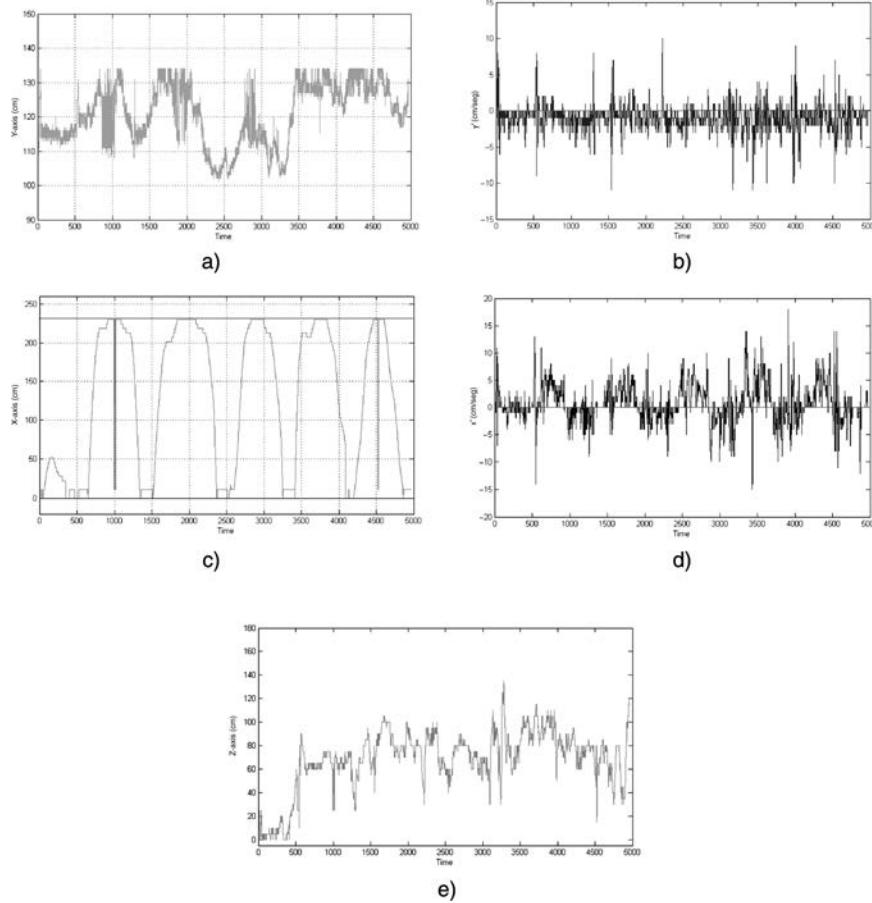


**Figure 9.18.** Image with the four corner points



**Figure 9.19.** Quadrotor helicopter following a line. Top images are used for the position algorithm, while bottom images are employed in the velocity estimation

Indeed, to find the 3D position, the image is filtered to delete the blue and green channels from the original images leaving only the red channel in a RGB image format. In this new image, a line-detection algorithm is utilized. A search algorithm is then applied to find the coordinates of four corner points of the rectangle (see Figure 9.18). These points are used to find the VPs  $V_x$  and  $V_y$ .



**Figure 9.20.** Quadrotor performance. (a)  $Y$  position of mini rotorcraft, (b)  $\dot{y}$  velocity of the mini rotorcraft, (c)  $X$  position of mini rotorcraft, (d)  $\dot{x}$  velocity of the mini rotorcraft, and (e)  $Z$  position of mini rotorcraft

Figure 9.20(a) shows the displacement of the quadrotor over its  $y$ -axis; this information is obtained from the frontal camera, and it is the value of  $T_z$  in camera coordinates, the third element in translation vector  $\mathbf{T}$  of this camera. The distance obtained from the visual algorithm is shown in centimeters; it is possible to

observe some variation in the estimation due to light changes over the displacement, nonetheless the control law guarantees stable flight.

In Figure 9.20(c), it is possible to see the quadrotor's trajectory over its  $x$ -axis, which is parallel to the line, from the initial point (0 cm) until the final point (230 cm). Once the final point has been reached, the translational displacement direction is changed by exchanging the initial and final points values. The distance covered is equal to the line length, that is, 230 cm. The lateral position of the quadrotor was obtained from the  $T_x$  value in translation vector  $\mathbf{T}$ , obtained from the frontal camera. Figure 9.20(e) shows the quadrotor altitude  $T_y$  in translation vector  $\mathbf{T}$  estimated by the frontal camera; the desired altitude was set to 80 cm from ground.

Figures 9.20(b) and 9.20(d) show the velocity estimation of the quadrotor, obtained by using an OF algorithm applied on the images coming from the downward camera. Both the figures show quadrotor displacement performing line-tracking experiments.

## 9.8. Conclusions

Two visual servoing approaches are proposed in this chapter for stabilizing a miniature quadrotor. In the first, the position was obtained using a frontal camera looking at a target placed on a wall; the vehicle velocity was estimated using a camera pointing downward running an OF algorithm. The second approach used the VP detection technique. The experimental tests showed that the quadrotor performed well during hover flight using the proposed vision-based control systems. Future work in this area includes tracking a wall for inspection purposes using the vision system as well as tracking a moving visual target. This work highlights the potential of the computer vision-based control strategies for mini UAVs.

## 9.9. Bibliography

- [ALH 10] AL-HELAL H., SPRINKLE J., “UAV search: maximizing target acquisition”, pp. 9–18, 2010.
- [ALT 05] ALTUG E., OSTROWSKI J., TAYLOR C., “Control of a quadrotor helicopter using dual camera visual feedback”, *International Journal of Robotics Research*, vol. 24, pp. 329–341, 2005.
- [BAR 04] BAR L., SOCHEN N., KIRYATI N., “Variational pairing of image segmentation and blind restoration”, *Proc. ECCV*, 2004.
- [BEA 95] BEAUCHEMIN S., BARRON J., “The computation of optical flow”, *ACM Computing Surveys*, vol. 27, pp. 433–467, 1995.

- [BOR 07] BORENSTEIN J., EVERETT H., FENG L. *et al.*, “Mobile robot positioning-sensors and mobile robot positioning-sensors and techniques”, *Journal of Robotic Systems, Special Journal of Robotic Systems, Special Issue on Mobile Robots*, vol. 14, no. 4, pp. 231–249, 2007.
- [BOU 99] BOUGUET J., Pyramidal Implementation of the Lucas Kanade Feature Tracker, Intel Corporation, Microprocessor Research Lab, 1999.
- [CAB 09] CABALLERO F., MERINO L., FERRUZ J., OLLERO A., “Unmanned aerial vehicle localization based on monocular vision and online mosaicking”, *Intelligent Robot System*, vol. 55, nos. 4–5, pp. 323–343, 2009.
- [CAN 86] CANNY J., “A computational approach to edge detection”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, no. 6, pp. 679–698, 1986.
- [CAS 04] CASTILLO P., DZUL A., LOZANO R., “Realtime stabilization and tracking of a four-rotor mini rotorcraft”, *IEEE Transactions on Control Systems Technology*, vol. 12, pp. 510–516, 2004.
- [CAS 05] CASTILLO P., LOZANO R., DZUL A., “Stabilization of a mini rotorcraft with four rotors: experimental implementation of linear and nonlinear control laws”, *IEEE Control Systems Magazine*, vol. 25, pp. 45–55, 2005.
- [CON 09] CONROY J., GREMILLION G., RANGANATHAN B., SEAN HUMBERT J., “Implementation of widefield integration of optic flow for autonomous quadrotor navigation”, *Autonomous Robots*, vol. 27, no. 3, pp. 189–198, 2009.
- [DUC 09] DUCHAINE V., BOUCHARD S., GOSSELIN M., “Computationally efficient predictive robot control”, *IEEE/ASME Transactions on Mechatronics*, vol. 12, no. 5, pp. 570–578, 2009.
- [FAU 95] FAUGERAS O., “Stratification of three-dimensional vision: projective, affine, and metric representations”, *Journal Optical Society of America*, vol. 12, no. 3, pp. 465–484, 1995.
- [GOMa] GOMEZ-BALDERAS J., GUERRERO J., SALAZAR S., Quadrotor and moving target video, <http://www.youtube.com/watch?v=3DYTncI19x0>.
- [GOMB] GOMEZ-BALDERAS J., GUERRERO J., SALAZAR S., Quadrotor hover with position control video, [http://www.youtube.com/watch?v=LkWP6-2\\_Esk](http://www.youtube.com/watch?v=LkWP6-2_Esk).
- [GOM 11a] GOMEZ-BALDERAS J., CASTILLO P., GUERRERO J., LOZANO R., “Vision based tracking for a mini-rotorcraft using vanishing points”, *International Federation of Automatic Control*, Milan, Italy, September 2011.
- [GOM 11b] GOMEZ-BALDERAS J., CASTILLO P., GUERRERO J., LOZANO R., *Vision Based Tracking for a Quadrotor Using Vanishing Points*, Springer, pp. 0921–0296, 2011.
- [GRE 04] GREEN W., OH P., BARROWS G., “Flying insect inspired vision for autonomous aerial robot maneuvers in near-earth environments”, vol. 21, pp. 2347–2352, 2004.
- [GUI 00] GUILLOU E., MENEVEAUX D., MAISEL E., BOUATOUCH K., “Using vanishing points for camera calibration and coarse 3D reconstruction from a single image”, *The Visual Computer*, vol. 16, pp. 396–410, 2000.

- [HAR 80] HARALICK R., “Using perspectives transformation in scene analysis”, *Computer Graphics Image Processing*, vol. 13, pp. 191–221, 1980.
- [HE 06] HE Z., IYER R., CHANDLER P., “Vision-based UAV flight control and obstacle avoidance”, American Control Conference, 2006.
- [KEM 06] KEMP C., Visual control of a miniature quadrotor helicopter, PhD Thesis, Churchill College, 2006.
- [KUR 04] KURDILA A., NECHYBA M., PRAZENICA R., DAHMEN W., BINEV P., DEVORE R., SHARPLEY R., “Vision-based control of micro-air-vehicles: progress and problems in estimation”, *43<sup>rd</sup> IEEE Conference on Decision and Control*, 2004.
- [MON 07] MONDRAGON I.F., CAMPOY P., CORREA J.F., MEJIAS L., “Visual model feature tracking for UAV control”, *IEEE International Symposium on Intelligent Signal Processing*, Alcala de Henares, Spain, 3–5 October 2007.
- [MUR 05] MURATETA L., DONCIEUXA S., BRIEREB Y.B. et al., “A contribution to vision-based autonomous helicopter flight in urban environments helicopter flight in urban environments”, *Robotics and Autonomous Systems*, vol. 50, pp. 195–209, April 2005.
- [QUI 05] QUIGLEY M., GOODRICH M.A., GRIFFITHS S., ELDREDGE A., BEARD R.W., “Target acquisition, localization, and surveillance using a fixed-wing mini-uav and gimbaled camera”, *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, vol. 3, pp. 2600–2605, 2005.
- [RAB] Rabbit Semiconductors, Dynamics C user manual, [www.rabbit.com](http://www.rabbit.com).
- [ROB 95] ROBERT L., ZELLER C., FAUGERAS O., HÉBERT M., Applications of non-metric vision to some visually guided robotics tasks, Report, no. 2584, 1995.
- [ROM 09] ROMERO H., SALAZAR S., LOZANO R., “Real-time stabilization of an eight-rotor UAV using optical flow”, *IEEE Transactions on Robotics*, vol. 25, pp. 809–817, 2009.
- [SAL 09] SALAZAR S., LOZANO R., ESCAREÑO J., “Stabilization and nonlinear control for a novel trirotor mini-aircraft”, *Control Engineering Practice*, vol. 17, pp. 886–894, 2009.
- [SAN 10] SANAHUJA G., CASTILLO P., SANCHEZ A., “Stabilization of n integrators in cascade with bounded input with experimental application to a VTOL laboratory system”, *International Journal of Robust Nonlinear Control*, vol. 20, no. 10, pp. 1129–1139, 10 July 2010.
- [SCH 07] SCHULZ H.-W., BUSCHMANN M., KRUGER L., et al., “Towards vision-based autonomous landing for small UAVs—first experimental results of the vision system”, *Journal of Aerospace Computing, Information, and Communication*, vol. 4, pp. 789–797, 2007.
- [STE 03] STEIN D., SCHEINERMAN E., STEIN G., SCHEINERMAN E., CHIRIKJIAN G., “Mathematical models of binary spherical-motion encoders”, *IEEE/ASME Transactions on Mechatronics*, vol. 8, pp. 234–244, 2003.
- [STO 09] STOWERS J., BAINBRIDGE-SMITH A., HAYES M., MILLS S., “Optical flow for heading estimation of a quadrotor helicopter”, *Journal International Journal of Micro Air Vehicles*, vol. 1, no. 4, December 2009.

- [TEE 92] TEEL A., “Global stabilization and global stabilization and restricted tracking for multiple integrators with bounded controls”, *Systems and Control Letters*, vol. 18, pp. 165–171, 1992.
- [WAN 90] WANG L., TSAI W., “Computing camera parameters using vanishing line information from a rectangular parallelepiped”, *Machine Vision and Applications*, vol. 3, pp. 129–141, 1990.
- [WAN 08a] WANG G., WU J., JI Z., “Single view based pose estimation from circle or parallel lines”, *Journal Pattern Recognition Letters Archive*, vol. 29, no. 7, May 2008.
- [WAN 08b] WANG L., TSAI W., “Camera calibration by vanishing lines for 3-D computer vision”, *Pattern Recognition Letters*, vol. 29, pp. 977–985, 2008.
- [WU 04] WU H., SUN D., ZHOU Z., “Micro air vehicle: configuration, analysis, fabrication, and test”, *IEEE/ASME Transactions on Mechatronics*, vol. 9, pp. 108–117, 2004.
- [YU 07] YU Z., NONAMI K., SHIN J., et al., “3D vision based landing control of a small scale autonomous 3d vision based landing control of a small scale autonomous helicopter”, *International Journal of Advanced Robotic Systems*, vol. 4, no. 1, pp. 51–56, 2007.
- [ZHA 05] ZHANG L., DEHGHANI A., SU Z., et al., “Development of a mechatronic sorting system for removing contaminants from wool”, *IEEE/ASME Transactions on Mechatronics*, vol. 10, no. 3, pp. 297–304, 2005.
- [ZUW 06] ZUWHAN K., Geometry of vanishing point and its application to external and realtime pose estimation, Research Reports, Institute of Transportation Studies, UC Berkeley, 2006.

## Chapter 10

# Toward Vision-Based Coordination of Quadrotor Platoons

This chapter focuses on the position coordination of two camera-equipped quadrotors in hover flight. In our approach, each vehicle is hovering over a fixed target on the ground. Applying a homography estimation technique, the aircrafts are capable of estimating their relative position with respect to their corresponding target. The control objective consists of coordinating the relative three-dimensional position of the quadrotors by sharing position information between them. Simulations and real-time experiments illustrate the performance of our method.

### 10.1. Introduction

The miniature rotorcraft flight formation control involves the integration of different domains, such as rotorcraft control, coordination control, computer vision among others. Control of unmanned aerial vehicles (UAV) requires knowledge of the state of the aircraft, mainly linear and angular velocity and position. Normally miniature UAVs payload is severely restricted to avoid unnecessary energy consumption and increase autonomy. Standard onboard inertial sensors for UAVs are accelerometers and gyros which measure angular velocity. Small-size GPS can be used to estimate the position and velocity of a UAV, however, the measurements are reliable only when the sensor is far from urban areas.

Computer vision becomes an important characteristic in the field of mobile robots. It is currently used in the feedback control loop as a cheap, passive, and

---

Chapter written by L.R. GARCÍA CARRILLO, J.A. GUERRERO and R. LOZANO.

information-abundant sensor, usually combined with an inertial measurement unit (IMU) to provide robust relative attitude information and allowing autonomous position and navigation. Also, note that an onboard vision system increases the sensor suit including a GPS, which provides position information relative to an inertial frame, but fails in indoor or in noisy environments. Onboard computer vision systems provide information which is obtained, for example, from the detection of landmarks, allowing the UAV to estimate its relative position with respect to such landmarks. Once the UAV knows its position, a control strategy could be implemented in order to achieve a desired goal.

Several research works have been performed aiming to control the flight of UAV's by using cameras as position sensors ([SAR 02, ROM 06, SAL 09]). A vision algorithm for visual navigation and landing of a helicopter is presented in [SAR 03]. For the purpose of estimating the location and orientation of a helicopter landing pad, the authors [YAN 98] use the projections of parallel lines. By this approach, the vision sensor estimates the position of the camera relative to the landmark, but it cannot estimate its velocity, which is an important data for controlling the position of the UAV. An approach based on optical flow techniques has been applied to the real-time stabilization of an eight-rotor UAV in [ROM 09]. Systems with more than one camera have also been studied, for example the authors of [ALT 05] use a system based on two cameras (one onboard, one offboard) in order to perform an autonomous hover, takeoff and landing of a quadrotor helicopter. Also, a two camera system is presented in [RON 10], with the drawback that none of the cameras is embedded on the UAV. In [ACH 09], a stereo vision system combined with a multisensor suit is proposed. The disadvantage of onboard stereo vision systems is that they increase the payload on the UAV. Omnidirectional visual sensor are also used in UAV's. In [DEM 06], a video sequence captured from an omnidirectional camera mounted over an aerial vehicle is analyzed in order to compute the vehicle's attitude.

Given the diversity of vision systems, in this chapter, we have decided to use a very simple combination of computer vision techniques and a simple control method applied to the position stabilization of a quadrotor UAV. The computer vision approach consists of a landmark detection and tracking algorithm. It is conceived to estimate the position and linear velocity of the UAV with respect to a landing pad on the ground, by using a calibrated onboard monocular camera.

Once a UAV is capable of estimating and controlling its position, the idea of flight formation becomes clear; however, a communication model among UAV is needed. As discussed in previous chapters graph theory has been used to model the communication between agents in a multiagent system. Recall that in a graph theoretical approach to flight formation, every node in the graph is considered as an agent or aerial vehicle which can have information exchange with all or several agents.

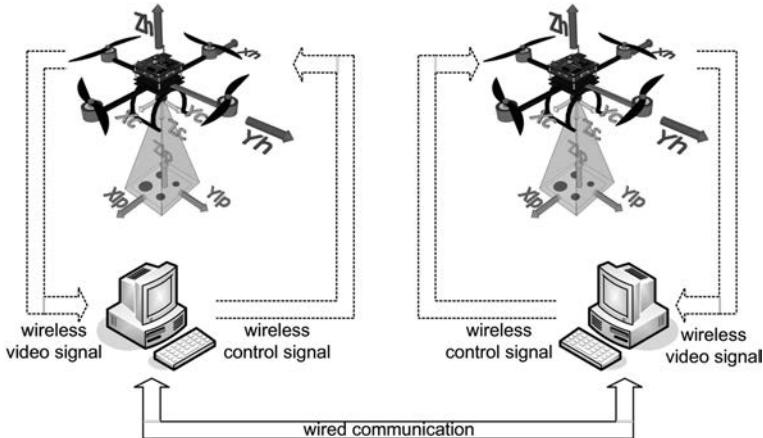
Recent advances in UAV control, computer vision, graph theory, etc., are factors that have intensified the interest of researchers in the UAV flight formation problem. Recent research improvements include flight formation control based on a forced consensus algorithms [GUE 10], cooperative and coordination control of UAVs ([MIC 11, OUN 10] among others), and lifting a rigid body using multiple aerial vehicles [MIC 11].

**REMARK 10.1.**— Some experimental results on coordination and cooperative control have been obtained using the Vicon motion capture system. This is an indoor localization system that parameterize the 3D space using a set of infrared cameras. The main advantage of this system is the accuracy (millimeters) and the processing speed (100–400 Hz). However, the main disadvantages of this system are its price which is expensive and the fact that it cannot be used in real outdoor missions due to its centralized sensing and localization nature.

## 10.2. Problem statement

### 10.2.1. Description of the process

Consider the case of two quadrotor helicopters equipped with vision and inertial sensing systems (see Figure 10.1). The vision system provides the 3D position of the vehicle with respect to an artificial landmark of known dimensions and orientation. The angular behavior of the helicopter is measured by means of the inertial sensors.



**Figure 10.1.** Vision-based position stabilization schema

A wireless communication system between the helicopters enables the exchange of position and orientation data. To achieve this, wireless RF modems will be taken into

account due to their flexibility when using different medium access control (MAC) protocols.

### 10.2.2. Objective of our approach

One of the main problems in UAV coordination control is the estimation of the vehicle's position. A promising solution to overcome such an issue consists of the implementation of an onboard vision system. The goal of this chapter is to develop a vision-based approach to achieve a coordination of multiple UAVs.

## 10.3. Dynamic model and control of a quadrotor

### 10.3.1. Dynamic model

Since the purpose of this chapter is to develop a miniature quadrotor formation flight control, let us consider the dynamical model introduced in Chapter 6:

$$\ddot{x} = -F_T \sin(\theta), \quad [10.1]$$

$$\ddot{y} = F_T \cos(\theta) \sin(\phi), \quad [10.2]$$

$$\ddot{z} = F_T \cos(\theta) \cos(\phi) - 1, \quad [10.3]$$

$$\ddot{\phi} = \tau_\phi, \quad [10.4]$$

$$\ddot{\theta} = \tau_\theta, \quad [10.5]$$

$$\ddot{\psi} = \tau_\psi, \quad [10.6]$$

where  $F_T$  is the thrust force vector in the body system,  $(x, y, z)$ ,  $(\phi, \theta, \psi)$ , and  $(\tau_\phi, \tau_\theta, \tau_\psi)$  represent the quadrotor position, orientation, and torque control, respectively. Note that the mass and gravitational acceleration have been normalized.

REMARK 10.2.– There are other ways to represent the orientation of rigid bodies, e.g. quaternion; however for the purpose of this work, Euler angles represent a simple and practical solution to be adopted.

### 10.3.2. Vehicle stabilization

In order to stabilize the altitude and the heading of the mini rotorcraft, the following control inputs are proposed:

$$F_T \triangleq \frac{-a_1 \dot{z} - a_2(z - z^d) + 1}{\cos(\phi) \cos(\theta)}, \quad [10.7]$$

$$\tau_\psi \triangleq -a_3 \dot{\psi} - a_4(\psi - \psi^d), \quad [10.8]$$

where  $a_1, a_2, a_3$ , and  $a_4$  are positive constant;  $z^d$  and  $\psi^d$  are the desired altitude and heading, respectively. Note that, the control inputs [10.7] and [10.8] stabilize the altitude and heading in closed-loop system, such that:

$$\lim_{t \rightarrow \infty} \|z - z^d\| \rightarrow 0,$$

$$\lim_{t \rightarrow \infty} \|\psi - \psi^d\| \rightarrow 0.$$

Consequently, introducing [10.7] into [10.2] the lateral dynamic model is represented by the following set of equations:

$$\begin{aligned}\ddot{y} &= \tan \phi, \\ \ddot{\phi} &= \tau_\phi.\end{aligned}$$

Similarly, the longitudinal dynamic model is represented by:

$$\ddot{x} = \frac{-\tan \theta}{\cos \phi}, \quad [10.9]$$

$$\ddot{\theta} = \tau_\theta. \quad [10.10]$$

It is assumed that pitch and roll angle will be operated in a neighborhood of the origin, i.e.  $|\phi| < \pi/10$ . Moreover, the proposed control approach provides an upper bound for the attitude subsystem, such that  $\tan \phi \approx \phi$ . Then, the lateral dynamical system can be reduced to:

$$\ddot{y} = \phi, \quad [10.11]$$

$$\ddot{\phi} = \tau_\phi. \quad [10.12]$$

The lateral and longitudinal control law can be stated as:

$$\begin{aligned}\tau_\phi &= -\sigma_4(\dot{\phi}) - \sigma_3(\dot{\phi} + \phi) - \sigma_2(\dot{\phi} + 2\phi + \dot{y}) \\ &\quad - \sigma_1(\dot{\phi} + 3\phi + 3\dot{y} + y).\end{aligned} \quad [10.13]$$

$$\begin{aligned}\tau_\theta &= -\sigma_4(\dot{\theta}) - \sigma_3(\dot{\theta} + \theta) - \sigma_2(\dot{\theta} + 2\theta - \dot{x}) \\ &\quad - \sigma_1(\dot{\theta} + 3\theta - 3\dot{x} - x).\end{aligned} \quad [10.14]$$

#### 10.4. Vision-based position estimation

The helicopter's  $(x, y, z)$  positions and  $(\dot{x}, \dot{y}, \dot{z})$  velocities are estimated with respect to a landing pad on the ground, using a monocular vision system. The developed technique enables measuring the states that are difficult to obtain from

conventional navigation systems, for example GPS, when performing in urban environments or indoors.

In the proposed approach, the 3D position estimation is based on the computation of homographies, while translational velocity is obtained based on an optical flow computation. The states estimated are applied in a full-state feedback controller, with the purpose of generating control inputs for stabilizing the quadrotor 3D position during flight. The effectiveness of the proposed method has been verified under real-time experiments. Some graphics representing the behavior of the helicopter are shown to illustrate the performance of the UAV during flight.

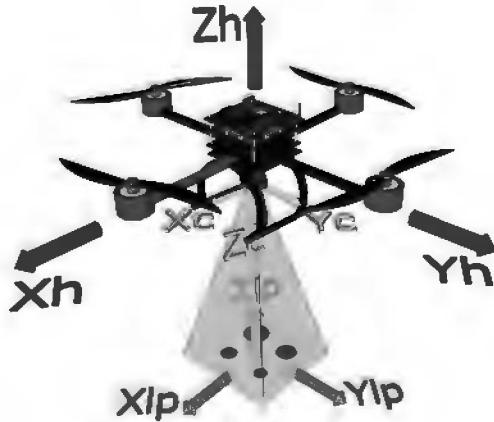
#### 10.4.1. Visual system setup

Controlling the 3D position of a UAV depends on the knowledge of the  $(x, y, z)$  vehicle coordinates and  $(\dot{x}, \dot{y}, \dot{z})$  translational velocities with respect to a well-known reference frame. Such values are required data for the controller in order to generate the control inputs to stabilize the aircraft over a desired location. To fulfill this situation a vision system has been implemented in order to provide the required position and velocities information. The complete system proposed consists of the calibrated camera onboard the UAV, a landing pad or artificial marker placed on the ground, an imaging processing algorithm running on a supervisory ground station PC, and the wireless link between the helicopter and the supervisory ground station. Figure 10.2 shows the proposed system which can be described as:

- *Quadrotor UAV*: with a body fixed frame  $(X_h, Y_h, Z_h)$ , assumed to be at its center of gravity.  $Z_h$  represents the yaw axis, and pointing upwards.  $X_h$  and  $Y_h$  are the roll and pitch axis, respectively.
- *Strapdown camera*: pointing downwards, with a reference frame  $(X_c, Y_c, Z_c)$ . When moving, the camera surveys the scene passing below the quadrotor. Since  $X_c - Y_c$  and  $X_h - Y_h$  are considered as parallel planes, then the visual information collected by the camera can be used to stabilize the vehicle.
- *Landing pad*: artificial landmark of known dimensions, formed by four circles of known coordinates, painted on high contrast background and placed underneath the rotorcraft. The coordinates frame  $(X_{lp}, Y_{lp}, Z_{lp})$  represents the inertial reference frame.

The planes formed by  $(X_h - Y_h)$  and  $(X_{lp} - Y_{lp})$  are considered to be parallel because it is assumed that the rotorcraft is in hover flight over the landing pad. Running in the supervisory ground station, an algorithm for image processing provides an estimate of the helicopter altitude, position in the  $(X, Y)$  plane, and translational velocities. As mentioned earlier, the 3D position information is deduced by a homography estimation technique applied to the image of the landing pad, while the translational velocities are estimated by means of optical flow computation. All

information sensed by the camera is related to the landing pad image. It will be shown that this information is rich enough to stabilize the 3D position of the vehicle. Let us explain first the procedure performed to compute the vehicle's positions and velocities.



**Figure 10.2.** Vision-based position stabilization schema

This section describes the imaging algorithms developed with the purpose of estimating the vehicle's states required for stabilizing its position during flight. First, a homography estimation technique for computing the 3D position is described, followed by a technique for deriving translational velocities. Finally, a method for estimating the homography when the detection of the landing pad fails is presented.

#### 10.4.2. Computing the 3D position

In order to estimate the UAV position relative to the landing pad, the extrinsic parameters of the camera are computed at every image frame. This is achieved by implementing a homography estimation technique, which provides the  $(x, y, z)$  position and  $(\psi, \theta, \phi)$  orientation of the camera with respect to the artificial landmark in the image scene. The action of the homography can be expressed as [BRA 08]:

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = s \mathbf{K} \begin{bmatrix} r_1 & r_2 & r_3 & \mathbf{T} \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \quad [10.15]$$

where  $[x \ y \ 1]^T$  represents the landing pad position in the camera image,  $s$  is a known scale factor,  $\mathbf{K} \in \mathbb{R}^{3 \times 3}$  represents the intrinsics parameters camera matrix,  $\mathbf{R} = [r_1 \ r_2 \ r_3] \in \mathbb{R}^{3 \times 3}$  are the extrinsics rotation parameters,  $\mathbf{T} \in \mathbb{R}^{3 \times 1}$  is the

extrinsics translation parameters vector, and  $[X \ Y \ Z \ 1]^T$  is the real landing pad position.

Without the loss of generality, the landing pad plane is defined so that  $Z = 0$ . This is done because, if we break up the rotation matrix into three  $3 \times 1$  columns (i.e.  $\mathbf{R} = [r_1 \ r_2 \ r_3]$ ), then one of those columns is not needed. Therefore, the homography matrix  $\mathbf{H}$  can be expressed as  $\mathbf{H} = s\mathbf{K}[r_1 \ r_2 \ \mathbf{T}]$ . The homography matrix  $\mathbf{H}$  is divided into two parts: the physical transformation (which locates the observed object plane) and the projection (the camera intrinsic matrix).

Rotation  $\mathbf{R}$  is described by three angles and translation  $\mathbf{T}$  is defined by three offsets; hence there are six unknowns for each view. The known planar object (the artificial landmark) provides eight equations, that is the mapping of a rectangle into a quadrilateral can be described by four  $(x, y)$  image points. For every instant, when the aerial vehicle is in hovering mode, it is possible to compute the homography matrix  $\mathbf{H}$  using the *a priori* knowledge of the position of the four centroids of the circles [HOR 89]. Using this estimated transformation matrix and the intrinsic camera matrix previously identified by an off-line calibration based on the method in [BOU 11], it is possible to calculate the camera extrinsic parameters, and therefore we have the vehicle's  $(x, y, z)$  position with respect to the landing pad on the ground.

Each one of the circles are detected in the image using an OpenCV function, next, they are classified according to the magnitude of its radius, allowing a correct identification of the landing pad orientation. The first circle corresponds to the upper left circle (or circle with smallest radius), continuing with the upper right circle as the second in the list. The lower left circle comes next according to its radius, and finally the lower right circle is identified as the circle whose radius magnitude is bigger. Figure 10.3 shows an image of the landing pad, as viewed from the camera, where the four circles' perimeters are highlighted with different colors according to the magnitude of its radius.

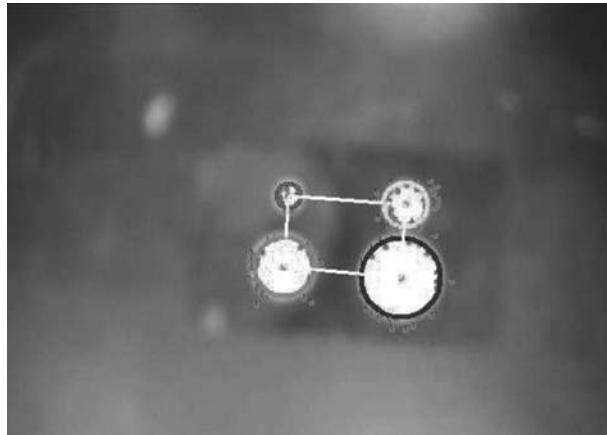
An erroneous detection of the landing pad circles must be discarded, since it will provide an erroneous position estimation. With this purpose, the parallelism of the lines mapped from the landing pad must be verified. Figure 10.2 shows that the four circles of the landing pad are positioned forming the corners of a rectangle. The line between the two upper corners and the line joining the two lower corners must satisfy a parallelism constrain. The same restriction is checked for the line joining the two left corners and the line between the two right corners. Parallelism verification is based on the slope of a line equation:

$$m_* = \frac{y_f - y_i}{x_f - x_i} \quad [10.16]$$

where  $i$  and  $f$  stand for initial and final coordinates, respectively. Thus, the slope  $m_{up}$  of the upper line must be almost equal to the slope  $m_{lo}$  of the lower line, while the slope  $m_{le}$  of the left line must be almost equal to the slope  $m_{ri}$  of the right line:

$$|m_{up} - m_{lo}| < \epsilon \quad ; \quad |m_{le} - m_{ri}| < \epsilon \quad [10.17]$$

where  $\epsilon$  stands for a constrain helping to determine the lines parallelism. Every detection of the four circles, validating the two previous verifications, ensures that a good planar homography could be estimated, resulting in a good computation of the camera extrinsic parameters. The four lines between circles, defining the sides of the rectangle, are highlighted in Figure 10.3.



**Figure 10.3.** Image processing for homography estimation: detection of the four circles and parallel lines verification

#### 10.4.3. Translational velocities

An optical flow computation procedure is applied to compute the  $(\dot{x}, \dot{y}, \dot{z})$  translational velocities of the aerial vehicle with respect to the landing pad. In order to compute optical flow, the Lucas–Kanade pyramidal algorithm [BOU 99] has been implemented in combination with a feature-detecting algorithm. This approach provides an accurate estimation of the motion field since it does not take into account the non-landing pad areas, where the motion field cannot be accurately determined.

Let us consider the camera moving with respect to a rigid scene. The velocities and rotation rates of the camera in the inertial frame are expressed by  $(V_x, V_y, V_z)$  and  $(w_x, w_y, w_z)$ , respectively. To accurately estimate the pseudo-speeds of the engine, let us define a tracking zone surrounding the landing pad, in a way that the centroid of the zone and the center of the landing pad coincide. The most representative features over

the zone are detected using an OpenCC function devoted to such task. These features, usually the circles perimeter, are selected as *features to track for*. A tracking process based on OpenCV-based optical flow computation is performed over the entire image, allowing measuring the displacements of the tracked features.

Thus, the mean of the optical flow calculated on all those points can be expressed as a function of the camera movement as follows:

$$\bar{OF}_x = \bar{V}_{OF_x} + K_x \bar{V}_{OF_z} + \bar{R}_{OF_x} \quad [10.18]$$

$$\bar{OF}_y = \bar{V}_{OF_y} + K_y \bar{V}_{OF_z} + \bar{R}_{OF_y} \quad [10.19]$$

Using the results presented in [RON 09], the rotational optical flow is compensated and the pseudo-speeds ( $\bar{V}_{OF_x}$ ,  $\bar{V}_{OF_y}$ ,  $\bar{V}_{OF_z}$ ) are deduced. Since the camera system and the helicopter share the same movements, it can be said that the deduced pseudo-velocities depend on the rotorcraft movement. Indeed, the camera is mounted onboard the quadrotor and fixed in a way that it has no degree of freedom. Thus, it can be written as:

$$\bar{V}_{OF_x} = -\frac{f\dot{x}}{z} \quad [10.20]$$

$$\bar{V}_{OF_y} = -\frac{f\dot{y}}{z} \quad [10.21]$$

$$\bar{V}_{OF_z} = \frac{\dot{z}}{z} \quad [10.22]$$

where  $(\dot{x}, \dot{y}, \dot{z})$  is the speed vector of the rotorcraft center of gravity and  $z$  is the altitude. Therefore, from these three equations the proposed optical flow vision system allows speed estimation of the rotorcraft up to a scale factor, when flying at constant altitude. These estimations can be used to control the translational velocities of the rotorcraft.

#### 10.4.4. Prediction of the landing pad position

Changes of illumination from one frame to another as well as occlusions can lead to a poor performance of image processing algorithms. Owing to this, if the landing pad is not successfully detected in the current image, the vision algorithm will fail. To overcome such a situation, optical flow measurements are applied to estimate the position of the four circles centroids in the current image as:

$$\rho_x^k = \rho_x^{k-1} + \Delta_T \bar{V}_{OF_x} \quad [10.23]$$

$$\rho_y^k = \rho_y^{k-1} + \Delta_T \bar{V}_{OF_y} \quad [10.24]$$

where  $(\rho_x^k, \rho_y^k)$  represents the circle's centroid position at time instant  $k$ , and  $\Delta_T$  is the working frequency of the algorithm. The estimated centroids positions are used to compute the homography at each frame that the detection of the landing pad fails.

### 10.5. Coordination position control of two quadrotors

One of the problems of working with multiple autonomous vehicles is collision avoidance. A coordination strategy to ensure the formation and collision avoidance at the same time is proposed here. It should be noted that a multiagent approach ensures the flock centering as well as the collision avoidance among multiagents. To develop this approach, we will start by analyzing the longitudinal kinematic model for the multiquadrotor system which is given by:

$$\dot{\mathbf{x}} = -\mathcal{L}\mathbf{x}, \quad [10.25]$$

where  $\mathcal{L}$  is the Laplacian matrix of the information exchange graph.

It is worth mentioning that dynamics [10.25] can also be written as:

$$\dot{x}_i = \bar{u}_i, \quad \forall i = 1, \dots, n;$$

with multiple-agent consensus achieved using the following forced consensus algorithm:

$$\bar{u}_i = - \sum_{j \in \mathcal{N}_i} (x_i - x_j),$$

where  $\mathcal{N}_i$  is the set of vehicles transmitting their information to the vehicle  $i$ .

Observe that, when using [10.13] and [10.14] all the states go to the origin. And since, the control objective is to force the consensus of a set of quadrotor vehicles to a desired position and heading, we propose the following change of variables:

$$x \triangleq \sum_{j \in \mathcal{N}_i} (x_j - x_i), \quad [10.26]$$

$$y \triangleq \sum_{j \in \mathcal{N}_i} (y_j - y_i), \quad [10.27]$$

$$z \triangleq \sum_{j \in \mathcal{N}_i} (z_j - z_i), \quad [10.28]$$

$$\psi \triangleq \sum_{j \in \mathcal{N}_i} (\psi_j - \psi_i), \quad [10.29]$$

where  $x_i, y_i, z_i, \psi_i, x_j, y_j, z_j$ , and  $\psi_j$  represent the 3D position and heading of the  $i$ th quadrotor and the  $j$ -th quadrotor to be coordinated.

**REMARK 10.3.–** On the one hand, a multiple mini rotorcraft consensus can be achieved by means of a single integrator consensus algorithm, then [10.26]–[10.27]

provide a simple way to solve the coordination problem. On the other hand, we may think of the neighbors position of a mini rotorcraft as the position reference and thus the stability of every mini rotorcraft is guaranteed using the nonlinear control based on separated saturations.

From the previous control analysis, we have that  $x \rightarrow 0$ ,  $y \rightarrow 0$ ,  $z \rightarrow z^d$ , and  $\psi \rightarrow \psi^d$ , and from [10.26]–[10.27], this implies that:

$$\lim_{t \rightarrow \infty} \|x_j - x_i\| = 0, \quad [10.30]$$

$$\lim_{t \rightarrow \infty} \|y_j - y_i\| = 0. \quad [10.31]$$

From [10.7], [10.8] and [10.28]–[10.29], we have:

$$\lim_{t \rightarrow \infty} \|z_j - z_i\| = z^d, \quad [10.32]$$

$$\lim_{t \rightarrow \infty} \|\psi_j - \psi_i\| = \psi^d. \quad [10.33]$$

Therefore, the control laws  $\tau_\theta$ ,  $\tau_\phi$ ,  $\tau_\psi$ , and  $F_T$  for the longitudinal, lateral, heading, and altitude subsystems of the  $i$ th mini rotorcraft become:

$$\begin{aligned} \tau_{\theta_i} &= -\sigma_4(\dot{\theta}_i) - \sigma_3(\dot{\theta}_i + \theta_i) - \sigma_2(\dot{\theta}_i + 2\theta_i - \dot{x}_i) \\ &\quad - \sigma_1\left(\dot{\theta}_i + 3\theta_i - 3\dot{x}_i - \left(\sum_{j \in \mathcal{N}_i} (x_j - x_i)\right)\right). \end{aligned} \quad [10.34]$$

$$\begin{aligned} \tau_{\phi_i} &= -\sigma_4(\dot{\phi}_i) - \sigma_3(\dot{\phi}_i + \phi_i) - \sigma_2(\dot{\phi}_i + 2\phi_i + \dot{y}_i) \\ &\quad - \sigma_1\left(\dot{\phi}_i + 3\phi_i + 3\dot{y}_i - \left(\sum_{j \in \mathcal{N}_i} (y_j - y_i)\right)\right). \end{aligned} \quad [10.35]$$

$$F_T = \frac{-a_1\dot{z} - a_2\left(\sum_{j \in \mathcal{N}_i} (z_j - z_i) - z^d\right) + 1}{\cos(\phi) \cos(\theta)}, \quad [10.36]$$

$$\tau_\psi = -a_3\dot{\psi} - a_4\left(\sum_{j \in \mathcal{N}_i} (\psi_j - \psi_i) - \psi^d\right). \quad [10.37]$$

Note that [11.1]–[11.4] enable us to stabilize the coordination for a group of  $n$  mini rotorcraft.

## 10.6. Architecture of the experimental platforms

### 10.6.1. *Quadrotor system*

The quadrotor experimental platform considered in the present study is shown in Figure 10.4. It was developed at the University of Technology of Compiègne, France. Some characteristics of this aerial vehicle are summarized in Table 10.1.



**Figure 10.4.** The four-rotor aircraft experimental platform

Parameter	Value
Diameter between rotors	40 cm
Weight	800 g
Autonomy	15 min
Power	Li-Po battery
Motor	Brushless

**Table 10.1.** Characteristics of the rotorcraft

The helicopter's onboard electronics are composed of two interconnected cards: the first board corresponds to a control unit, while the second board deals with the motors electronic speed controllers. The control unit card performs the essential tasks of sensing, communicating, and stabilizing the UAV attitude during fly. The properties of this board can be summarized as follows:

– *Processor*: A Texas Instruments® TMS320F2812 DSP module is used to process the data of the different sensing devices, and to compute the control algorithm, which sends the control inputs in the form of four pulse-width modulation (PWM) signals to the motor drivers.

– *Inertial sensors*: A MIDG II INS/GPS from Microbotics Inc.® is used to measure the angular position of the rotorcraft. We also use three additional gyros to measure angular velocity at a higher rate.

– *Atmospheric pressure sensor*: A Freescale® MPXH6115A pressure sensor is used with an appropriate amplifier circuit, to measure the altitude of the engine on an appropriate sensing range.

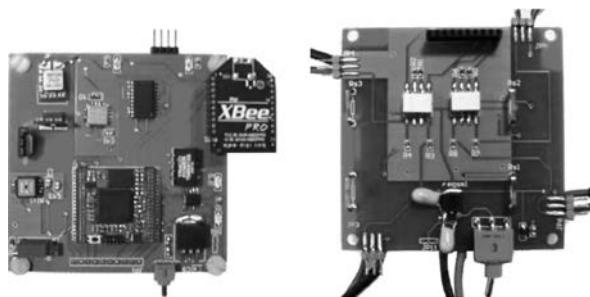
– *Battery voltage measurement circuit*: This electronic circuit is intended to provide the actual tension level of the supply battery. This information is used for several goals: to perform a safety landing and turn-off before an unwanted discharge of tension (avoiding accidents). Also, information regarding the supply voltage level is used in a preprocessing stage of the incoming measurements from the atmospheric pressure sensor.

– *Wireless link*: A XBee ZB ZigBee PRO® Radio Modem is used to link the ground station and the aerial vehicle. This communication link can be used to introduce external control inputs, send the sensors information to the ground station, etc.

The properties of the second board are as follows:

– *Signal conditioner*: In this stage, each control input of the four motors is decoupled from the rest of the electronic systems. The PWM signals are also filtered and conditioned.

The control unit board and the signal conditioner board are shown in Figure 10.5.



**Figure 10.5.** The electronics on board (left side: control board; right side: signal conditioner board)

#### 10.6.2. Ground station

The ground station for each of the platforms consists of a desktop PC, a flight simulator Cyborg-X® joystick, a XBee ZB ZigBee PRO® Radio Modem, and a Diversity video receiver system. Using this station, the user can send information to the aerial vehicle. Different flying modes can be chosen: manual control, altitude stabilization using the pressure sensor, vision-based position hold, and reactive navigation. The quadrotor includes some safety features, such as an emergency stop switch and a signal condition that verifies that the thrust command on the ground station is at zero before starting the motors. The ground station receives and saves all the information needed to debug and analyze the flight experiments and results. The quadrotor system and the ground station PC can be seen in Figure 10.6.

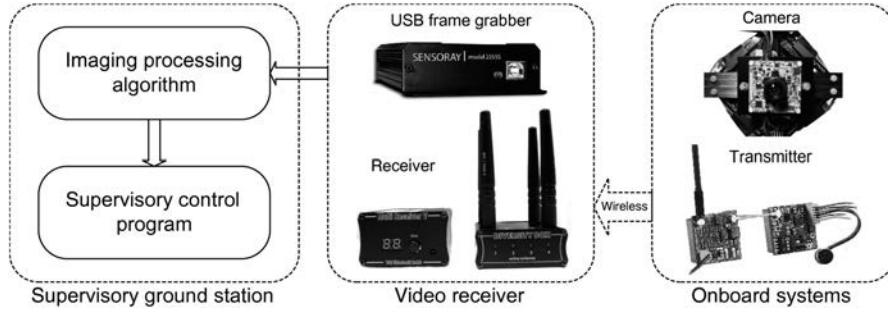


**Figure 10.6.** Quadrotor system and ground station PC

### 10.6.3. Monocular imaging system implementation

The quadrotor's monocular imaging system was developed with the purpose of enabling the vehicle to perform vision-based tasks. The imaging sensor is a *CTDM-5351* camera from SONY, with a resolution of  $640 \times 480$  pixels, installed in the lower part of the vehicle, placed pointing downwards. This camera has been chosen, since it offers a considerably high-quality image within well and poorly illuminated areas. The analog output of the camera is connected directly to a wireless *Micro PLL Transmitter* and a *Micro Booster* of 200 mW. By combining these two components, the transmission power can be improved up to 20 times, ensuring a good quality video during the UAV's missions. With the purpose of reducing possible disturbances in the transmitted images, the video signal is recovered by means of a four-antenna *Diversity System Receiver*. Incoming signals are received by each of the four antennas, so that the receiver can evaluate which antenna is providing the most suitable signal. Once detected, the system switches to this antenna automatically. The receiver outputs the video signal through a composite connector to a *Sensoray 2255* USB Frame Grabber, which is specially designed for fast video acquisition. The overall system, consisting of transmitter, receiver, and frame grabber, was chosen with the purpose to reduce possible time delays, and to ensure a high-quality real-time video.

The frame grabber is connected to the supervisory ground station through a USB port. A C-coded imaging application based on OpenCV functions performs image processing for estimating the vehicle's states. OpenCV [BRA 08] is a functions library containing a series of low-overhead, high-performance operations, that can be used to perform fast computing algorithms on images. An overall scheme of the monocular system components is shown in Figure 10.7.



**Figure 10.7.** Schema of the monocular vision system

The information extracted by the imaging application is placed on a fixed memory segment that is shared with the supervisory control application. With this method, the supervisory control application receives the required data for performing vision-based tasks.

### 10.7. Experimental results

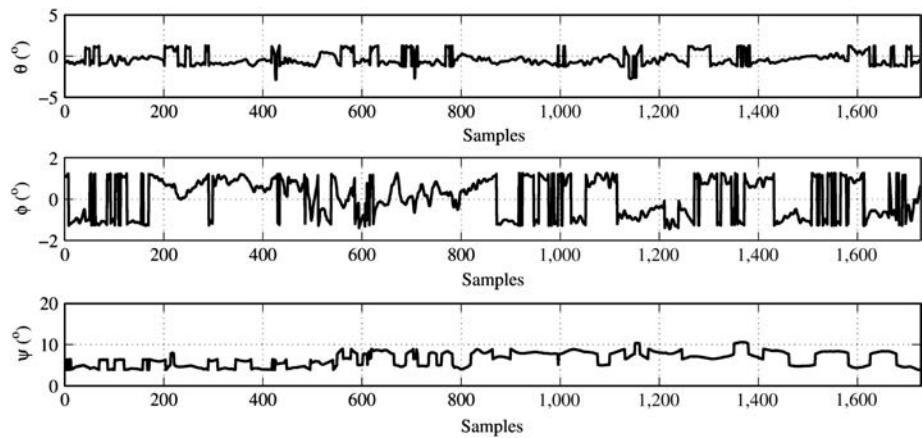
This section presents some preliminary results consisting of the autonomous stabilization of a single quadrotor during flight. The main idea is to use visual information with the purpose of maintaining the 3D position of the aerial vehicle with respect to the landing pad placed on ground. With these results, we demonstrate that, using the vision-based methodology previously presented, it is possible to obtain the signals required in the control strategy for coordinating the position of two quadrotors equipped with visual feedback.

For the present experiments, the vision system provides the helicopter's 3D position and translational velocity, while the embedded inertial electronic system provides the attitude data. To verify the performance of the vision-based position control method, we conducted a set of experiments which are described next.

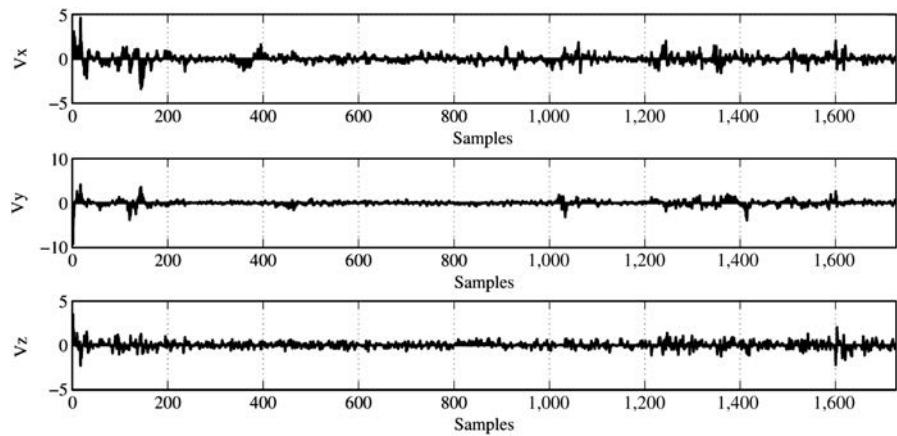
The helicopter's position reference is selected when the vehicle is placed exactly on top of the landing pad, such position is used as the X-Y desired position during the rest of the experiment. The desired altitude is fixed at 150 cm, while the desired  $\psi$  yaw angle is fixed at 5°. The parameter chosen in the control strategy are:  $k_{pz} = 0.68$ ,  $k_{vz} = 1.6$ ,  $k_{p\psi} = 38$ ,  $k_{p\phi} = 1,350$ ,  $k_{0y} = 1$ ,  $k_{1y} = 2$ ,  $k_{2y} = 38$ ,  $k_{3y} = 1,400$ ,  $k_{0x} = 1$ ,  $k_{1x} = 2$ ,  $k_{2x} = 38$ , and  $k_{3x} = 1,350$ . Note that, during the experiments, the attitude stabilization control is always running at a higher frequency, which ensures that the Euler angles are always near to zero (hover flight condition).

Figures 10.8, 10.9, and 10.10 show the results obtained from the experiment performed. It is important to mention that, for these graphics, each 10 samples

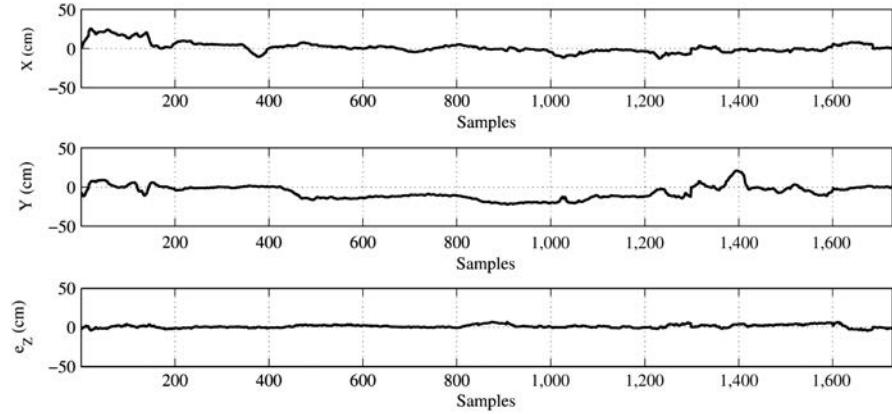
correspond to 1 sec. Note that the pitch and roll angles remain in the interval  $(-1.5, 1.5)$  degrees. Therefore, it can be concluded that the position control adds only small changes in the attitude of the rotorcraft for bringing the position to the desired one. This is an important property because the position controller runs at a lower rate compared to the attitude controller, then a smooth position control is necessary to ensure the stability of the vehicle. A picture of the quadrotor performing the autonomous position hold experiment is depicted in Figure 10.11. It can be seen that the UAV maintains its 3D position with respect to the landmark.



**Figure 10.8.** Euler angles, experimental results



**Figure 10.9.** Velocities, experimental results



**Figure 10.10.** Position errors, experimental results



**Figure 10.11.** The quadrotor helicopter stabilized over the landmark in a desired position

### 10.8. Conclusions and future work

This chapter focuses on the position coordination of two camera-equipped quadrotors in hover flight. In the present approach, each vehicle is hovering over a fixed target on the ground. Applying a homography estimation technique, the aircrafts are capable of estimating their relative position with respect to their corresponding target. The main objective consists of coordinating the relative 3D position of the quadrotors, by sharing position information between them.

The experiment of autonomous position stabilization was performed indoors using only one quadrotor, showing that the vision-based strategy allows stabilizing the aerial vehicle at a selected X-Y-Z position above the landing pad. The attitude of the vehicle was not significantly perturbed by the control input used to correct the UAV X-Y-Z position. The vehicle's velocity also remained very close to zero. A video of the experiments can be seen at <http://www.youtube.com/watch?v=SQLSXruTnj0>. Note that actual experiments include a virtual quadrotor and a real quadrotor coordination.

Future work includes two or more experimental quadrotors with coordination, platoon vision-based trajectory tracking, and indoor cooperative navigation. Platform improvements include using embedded wireless communication technologies such that every quadrotor could exchange its position directly with its neighbors.

## 10.9. Bibliography

- [ACH 09] ACHTELIK M., BACHRACH A., HE R., PRENTICE S., ROY N., “Autonomous navigation and exploration of a quadrotor helicopter in GPS-denied indoor environments”, *First Symposium on Indoor Flight Issues*, Mayagüez, Puerto Rico, July 2009.
- [ALT 05] ALTUG E., OSTROWSKI J., TAYLOR C., “Control of a quadrotor helicopter using dual camera visual feedback”, *International Journal of Robotics Research*, vol. 24, no. 5, pp. 329–341, 2005.
- [BEA 01] BEARD R., LAWTON J., HADAEGH F., “A coordination architecture for spacecraft formation control”, *IEEE Transactions on Control Systems Technology*, vol. 9, pp. 777–790, 2001.
- [BOU 04] BOUABDALLAH S., NOTH A., SIEGWART R., “PID vs LQ control techniques applied to an indoor micro quadrotor”, *Proceedings of the 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2004. (IROS 2004)*, vol. 3, pp. 2451–2456, 28 September–2 October 2004.
- [BOU 99] BOUGUET J., Pyramidal implementation of the Lucas Kanade Feature Tracker, Technical Report, Intel Corporation, Microprocessor Research Labs, 1999.
- [BOU 11] BOUGUET J., Camera Calibration Toolbox for Matlab, June 2011 [http://www.vision.caltech.edu/bouguetj/calib\\_doc/index.html](http://www.vision.caltech.edu/bouguetj/calib_doc/index.html).
- [BRA 08] BRADSKI G., KAEHLER A., *Learning OpenCV*, O'Reilly Media, 2008.
- [CAS 05] CASTILLO P., LOZANO R., DZUL A., “Stabilization of a mini rotorcraft with four rotors”, *IEEE Control Systems*, vol. 25, no. 6, pp. 45–55, 2005.
- [DEM 06] DEMONCEAUX C., VASSEUR P., PEGARD C., “Omnidirectional vision on UAV for attitude computation”, *Proceedings of the IEEE International Conference on Robotics and Automation, ICRA 2006*, Orlando, Florida, USA, pp. 2842–2847, May 2006.
- [ERG 07] ERGINER B., ALTUG E., “Modeling and PD control of a quadrotor VTOL vehicle”, *Intelligent Vehicles Symposium, 2007 IEEE*, pp. 894–899, June 2007.

- [GUE 10] GUERRERO J., FANTONI I., SALAZAR S., LOZANO R., “Flight formation of multiple mini rotorcraft via coordination control”, *IEEE International Conference on Robotics and Automation (ICRA), 2010*, Anchorage, Alaska, USA, pp. 620–625, May 2010.
- [HOK 07] HOKAYEM P., STIPANOVIC D., SPONG M., “Reliable control of multiagent formations”, *American Control Conference, 2007, ACC '07*, pp. 1882–1887, July 2007.
- [HOR 89] HORAUD R., CONIO B., LEBOULLEUX O., LACOLLE B., “An analytic solution for the perspective 4-point problem”, *Proceedings of the CVPR '89, IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, San Diego, California, USA, pp. 500–507, June 1989.
- [ISI 03] ISIDORI A., MARCONI L., SERRANI A., “Robust nonlinear motion control of a helicopter”, *IEEE Transactions on Automatic Control*, vol. 48, no. 3, pp. 413–426, 2003.
- [LAC 03] LA CIVITA M., PAPAGEORGIOU G., MESSNER W., KANADE T., “Design and flight testing of a gain-scheduled H infin; loop shaping controller for wide-envelope flight of a robotic helicopter”, *Proceedings of the 2003 American Control Conference*, vol. 5, pp. 4195–4200, June 2003.
- [LAR 06] LARA D., SANCHEZ A., LOZANO R., CASTILLO P., “Real-time embedded control system for VTOL aircrafts: Application to stabilize a quad-rotor helicopter”, *Computer Aided Control System Design, 2006 IEEE International Conference on Control Applications, 2006 IEEE International Symposium on Intelligent Control, 2006 IEEE*, pp. 2553–2558, October 2006.
- [LEE 03] LEE D., LI P., “Formation and maneuver control of multiple spacecraft”, *Proceedings of the 2003 American Control Conference*, vol. 1, pp. 278–283, June 2003.
- [LEE 07] LEE L., SPONG M., “Stable flocking of multiple inertial agents on balanced graphs”, *IEEE Transactions on Automatic Control*, vol. 52, no. 8, pp. 1469–1475, August 2007.
- [LOZ 07] LOZANO R., *Objets Volants Miniatures: Modélisation et Commande Embarquée*, Hermès-Lavoisier, 2007.
- [MIC 11] MICHAEL N., FINK J., KUMAR V., “Cooperative manipulation and transportation with aerial robots”, *Autonomous Robots*, vol. 30, pp. 73–86, 2011.
- [OLF 06] OLFATI-SABER R., “Flocking for multiagent dynamic systems: algorithms and theory”, *IEEE Transactions on Automatic Control*, vol. 51, no. 3, pp. 401–420, March 2006.
- [OUN 10] OUNG R., BOURGAULT F., DONOVAN M., D’ANDREA R., “The distributed flight array”, *2010 IEEE International Conference on Robotics and Automation (ICRA)*, Anchorage, Alaska, USA, pp. 601–607, May 2010.
- [REN 07] REN W., “Consensus seeking in multi-vehicle systems with a time-varying reference state”, *American Control Conference, 2007, ACC '07*, pp. 717–722, July 2007.
- [ROM 06] ROMERO H., BENOSMAN R., LOZANO R., “Stabilization and location of a four rotor helicopter applying vision”, *American Control Conference*, Minneapolis, MN, USA, pp. 3930–3935, June 2006.
- [ROM 09] ROMERO H., SALAZAR S., LOZANO R., “Real-time stabilization of an eight-rotor UAV using optical flow”, *IEEE Transactions on Robotics*, vol. 25, no. 4, pp. 809–817, August 2009.

- [RON 09] RONDON E., FANTONI-COICHOT I., SANCHEZ A., SANAHUJA G., “Optical flow-based controller for reactive and relative navigation dedicated to a four rotor rotocraft”, *IEEE/RSJ International Conference on Intelligent Robots and Systems*, St. Louis, USA, pp. 684–689, October 2009.
- [RON 10] RONDON E., SALAZAR S., ESCAREÑO J., LOZANO R., “Vision-based position control of a two-rotor VTOL mini UAV”, *Journal of Intelligent & Robotic Systems*, vol. 57, pp. 49–64, 2010.
- [SAL 09] SALAZAR S., ROMERO H., LOZANO R., CASTILLO P., “Modeling and real time stabilization of an aircraft having eight rotors”, VALAVANIS K.P., OH P., PIEGL L.A. (eds), *Unmanned Aircraft Systems*, Springer, The Netherlands, pp. 455–470, 2009.
- [SAR 02] SARIPALLI S., MONTGOMERY J., SUKHATME G., “Vision-based autonomous landing of an unmanned aerial vehicle”, *Proceedings of the ICRA '02, IEEE International Conference on Robotics and Automation*, 2002, Washington, DC, USA, vol. 3, pp. 2799–2804, 2002.
- [SAR 03] SARIPALLI S., MONTGOMERY J., SUKHATME G., “Visually-guided landing of an unmanned aerial vehicle”, *IEEE Transactions on Robotics and Automation*, vol. 19, pp. 371–381, 2003.
- [TAN 03] TANNER H., JADBABAIE A., PAPPAS G., “Stable flocking of mobile agents, part I: fixed topology”, *Proceedings of the 42nd IEEE Conference on Decision and Control*, 2003, vol. 2, pp. 2010–2015, December 2003.
- [YAN 98] YANG Z., TSAI W., “Using parallel line information for vision-based landmark location estimation and an application to automatic helicopter landing”, *Robotics and Computer-Integrated Manufacturing*, vol. 14, no. 4, pp. 297–306, 1998.



## Chapter 11

# Optimal Guidance for Rotorcraft Platoon Formation Flying in Wind Fields

In this chapter, a time-optimal guidance for a platoon of rotorcraft flying in formation through a region of strong wind fields is presented. The main goal is to program the heading for the virtual center of mass in such way that the flight time between two-way points is minimized. The heading program is obtained by using a Zermelo navigation approach which provides a kinematic model for the heading subsystem. The heading kinematic model is later used to compute the cartesian coordinate of the optimal trajectory. The collective behavior of a rotorcraft platoon is modeled using a multiagent approach. A nonlinear control based on separated saturations ensures the convergence of the platoon to the desired formation. Then, a trajectory-tracking strategy based on a forced consensus strategy is utilized to guide the platoon along the optimal path. The proposed algorithm is illustrated in simulations.

### 11.1. Introduction

The problem of miniature rotorcraft flying in formation involves the integration of different domains such as rotorcraft control, coordination control, guidance control, and wireless communications among others. The work reported in the literature is by now quite vast and addresses different approaches for miniature rotorcraft stabilization including linear control [BOU 04], [LAR 06], [ERG 07], robust control [LAR 06], [LAC 03], [ISI 03], nonlinear control [LOZ 07], [CAS 05], [GUE 10a]

---

Chapter written by J.A. GUERRERO, Y. BESTAOUI and R. LOZANO.

among others. In [LAR 06], the authors propose a robust linear proportional derivative (PD) controller considering parametric interval uncertainty. There, the authors present a robust stability analysis and compute the robustness margin of the system with respect to the parameters uncertainty. In [CAS 05], a nonlinear control based on nested saturations is presented. In this approach, the dynamics are decoupled into lateral and longitudinal dynamical subsystems. Thus, nested saturations control was used to stabilize each subsystem.

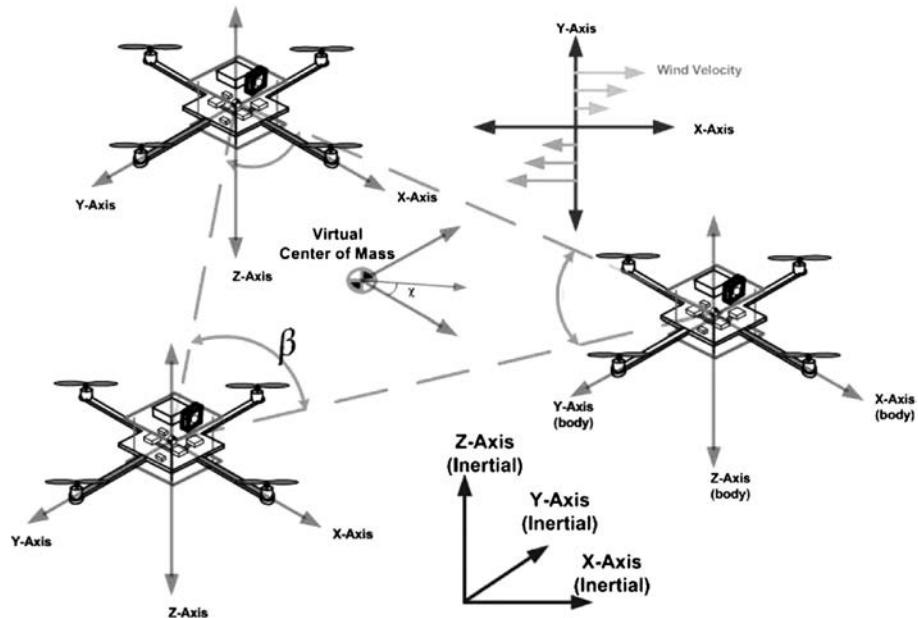
In order to model the communication between aerial vehicles, a theoretical graph approach is the most common approach. Every aerial vehicle in the platoon is modeled as a node on the graph which can have information exchange with its nearest neighbors. In [BEA 01], [TAN 03a], [TAN 03b], and [OLF 06], the authors use algebraic graph theory in order to model the information exchange between vehicles. By using this technique, several control strategies have been developed, e.g. see [OLF 06], [REN 07], [LEE 03], and [HOK 07] among others. In [OLF 06], the authors present several algorithms for consensus and obstacle avoidance for multiple agent systems. Ren [REN 07] presents a consensus algorithm for trajectory tracking of a time-varying reference for a single-integrator multiagent system. References [LEE 03] and [LEE 06] present a passive decomposition approach for consensus and formation control.

Recent advances in unmanned aerial vehicle (UAV) control, graph theory, and the technological revolution of the past decades have spurred the interest in the UAV flight formation problem. In [GUE 10a], a flight formation control based on a forced consensus algorithm is presented. Experimental results on cooperative and coordination control of UAVs have been obtained using the Vicon motion capture system (see [MIC 09] and [OUN 10] among others). Michael *et al.* [MIC 09] present a cooperative control strategy to lift a rigid body using three quadrotors.

On the one hand, experimental flight tests using quadrotor helicopters have shown that wind is an important phenomenon to take into account for guidance purposes. On the other hand, the research interests in quadrotor control are focused on the UAV navigation in both indoors and outdoors. The problem of guidance in winds was first introduced by Zermelo [ZER 30]. The work of Zermelo gave rise to different publications including [ARR 49] and [BRY 75] among others. In [JAR 01], a neighboring optimal control has been developed to find near-optimal trajectories starting from nominal waypoints. The authors assume that the winds vary in smooth continuous way.

The aim of this chapter is to study the problem of optimal guidance in general wind fields for a platoon of miniature rotorcraft flying in formation, as shown in Figure 11.1, using a Zermelo-based approach. A nonlinear control based on separated

saturations and a single-integrator coordination control strategy stabilize the rotorcraft formation. The coordination algorithm assumes that there are  $N$ -aerial vehicles which have some kind of information exchange between them. In this approach, every miniature rotorcraft is considered as an agent in the multiagent system. For simplicity, a decoupled dynamic coordination is used. Thus, the lateral, longitudinal, heading, and altitude dynamical subsystems of each miniature rotorcraft are considered as agents to be coordinated to follow a desired reference. To do this, combined with a nonlinear control, a force consensus control will be used to synchronize the behavior of a miniature rotorcraft platoon.



**Figure 11.1.** Multiple mini rotorcraft flying in formation

In order to achieve consensus among the members of the platoon, we propose a single-integrator coordination which implies position coordination only. An important advantage of the approach adopted here is that a member of the platoon will not be affected by perturbations in the altitude of its neighbors.

This chapter is organized as follows: section 11.2 introduces the aerial vehicle dynamical model, a formation control based on separated saturations and a trajectory-tracking scheme using the center of mass of the rotorcraft formation. In section 11.3, a Zermelo-based trajectory planning in wind fields is introduced. Simulation results

are discussed in section 11.6. Finally, the conclusions and future work are discussed in section 11.7.

## 11.2. Preliminaries

### 11.2.1. Dynamic model

Since the purpose of this work is to develop a time-optimal trajectory planning scheme for a rotorcraft platoon flying in formation, let us consider the  $i$ th rotorcraft dynamical model introduced in [GUE 10a]:

$$\begin{aligned}\ddot{x}_i &= -F_{T,i} \sin(\theta_i), \\ \ddot{y}_i &= F_{T,i} \cos(\theta_i) \sin(\phi_i), \\ \ddot{z}_i &= F_{T,i} \cos(\theta_i) \cos(\phi_i) - 1, \\ \ddot{\phi}_i &= \tau_{\phi,i}, \\ \ddot{\theta}_i &= \tau_{\theta,i}, \\ \ddot{\psi}_i &= \tau_{\psi,i},\end{aligned}$$

where  $F_{T,i}$  is the thrust force in the body system for the  $i$ th vehicle,  $\mathbf{P}_i = (x_i, y_i, z_i)$ ,  $\Theta_i(\phi_i, \theta_i, \psi_i)$ , and  $\tau_i(\tau_{\phi,i}, \tau_{\theta,i}, \tau_{\psi,i})$  represent the aerial vehicle position, orientation, and torque control, respectively. Note that the mass and gravitational acceleration have been normalized.

On one hand, note that this is a simplified dynamic model of the classical quadrotor helicopter. A detailed dynamic model can be found in [LOZ 07]. On the other hand, we remark that there are different ways to represent the orientation of rigid bodies, e.g. quaternion; however, for the purpose of this work, Euler angles represent a simple and practical solution to be adopted.

### 11.2.2. Vehicle control

As discussed in the previous section, different approaches for flight formation control of a group of miniature rotorcraft have been proposed [GUE 10a, GUE 10b]. For the purposes of this work, the separated saturations control scheme will be utilized

$$\tau_{\theta_i} = -\sigma_4(\dot{\theta}_i) - \sigma_3(\theta_i) - \sigma_2(\dot{x}_i) - \sigma_1 \left( \sum_{j \in \mathcal{N}_i} (x_j - x_i) - x^d \right), \quad [11.1]$$

$$\tau_{\phi_i} = -\sigma_4(\dot{\phi}_i) - \sigma_3(\phi_i) + \sigma_2(\dot{y}_i) + \sigma_1 \left( \sum_{j \in \mathcal{N}_i} (y_j - y_i) - y^d \right), \quad [11.2]$$

$$F_{T,i} = \frac{-k_{\dot{z}_i}\dot{z} - k_{z_i} \left( \sum_{j \in \mathcal{N}_i} (z_j - z_i) - z^d \right) + 1}{\cos(\phi) \cos(\theta)}, \quad [11.3]$$

$$\tau_{\psi,i} = -k_{\dot{\psi}_i}\dot{\psi} - k_{\psi_i} \left( \sum_{j \in \mathcal{N}_i} (\psi_j - \psi_i) - \psi^d \right). \quad [11.4]$$

where  $\sigma(\cdot)$  represents a saturation function,  $k_{z_i}, k_{\dot{z}_i}, k_{\psi_i}$ , and  $k_{\dot{\psi}_i}$  are positive constants,  $z^d$  and  $\psi^d$  are the desired altitude and heading, and  $\mathcal{N}_i$  is the group of neighbors of the vehicle  $i$ .

Note that [11.1] and [11.4] allow us to synchronize a group of  $N$  miniature rotorcraft to a fixed geometric formation relative to the leader.

### 11.3. Path planning

For the purpose of path generation, it is usually sufficient to treat only the translational motion [BES 11]. The translational equations of an aerial vehicle through the atmosphere are directly derived from Newton's law. In this derivation, the assumptions are that the Earth is non-rotating and flat and that the vehicle weight is constant.

#### 11.3.1. Center of mass of the platoon

The position of the center of mass of the platoon is described in the local coordinate system **NED** with unit vectors **n,e,d** pointing North, East, and Down, respectively. The flight path coordinate system relates the velocity vector of the center of mass with respect to the Earth's geographic system. Two angles relate the velocity coordinates to the geographic system. The heading angle  $\chi$  is measured from North to the projection of  $V$  (the center-of-mass velocity relative to the wind) in the local tangent plane and the flight path angle  $\gamma$  takes vertically up to  $V$ . The wind relative velocity vector is defined by the airspeed  $V$ , the flight path angle  $\gamma$ , and the heading  $\chi$ . The variables  $x, y, z$  are the center-of-mass inertial coordinates. The  $x$  and  $y$  directions are chosen such that the  $xy$  plane is horizontal, the  $x$ -direction is aligned with the principal axis of symmetry, and the  $z$ -direction is descending vertically. The equations of motion are expressed in a velocity coordinate frame attached to the aerial robot, considering the velocity of the horizontal wind  $\mathbf{W} = (W_x \ W_y \ W_z)^T$  (components of the wind velocity in the inertial frame).

#### 11.3.2. Zermelo navigation problem: case 2D

Zermelo's problem was originally formulated to find the quickest nautical path for a ship at sea in the presence of currents, from a given departure point in  $\mathbb{R}^2$  to a given

destination point. It can also be applied to the particular case of an aerial vehicle with a zero flight path angle and the wind velocity represented by  $\mathbf{W} = (W_x, W_y)$ .

#### 11.3.2.1. Navigation equation

Time-optimal trajectory generation can be formulated as follows:

$$\min \int_0^T dt,$$

subject to

$$\dot{x} = u_1(t) + W_x,$$

$$\dot{y} = u_2(t) + W_y,$$

with the constraint  $u_1^2(t) + u_2^2(t) \leq V_{\max}^2$ . If the terminal point is reachable at any time, then it is reachable in the minimal time. However, if the wind is too strong, there may be points that are not reachable at all.

The Hamiltonian is classically given by:

$$H = 1 + \lambda_1(u_1(t) + W_x) + \lambda_2(u_2(t) + W_y),$$

where the Lagrange multipliers are represented by  $\lambda_1, \lambda_2$ . The application of the necessary condition of optimality is given by

$$\frac{d\lambda_1}{dt} = -\frac{\partial H}{\partial x} = -\lambda_1 \frac{\partial W_x}{\partial x} - \lambda_2 \frac{\partial W_y}{\partial x},$$

$$\frac{d\lambda_2}{dt} = -\frac{\partial H}{\partial y} = -\lambda_1 \frac{\partial W_x}{\partial y} - \lambda_2 \frac{\partial W_y}{\partial y}.$$

Each extremal control  $u*(t)$  must satisfy  $\|u*(t)\| = V_{\max}$  for almost all  $t$ . The maximality condition yields

$$u*(t) = V_{\max} \frac{\lambda(t)}{\|\lambda\|}$$

for almost all  $t$ , as  $\lambda(t)$  cannot be identically zero.

Zermelo's navigation formula consists of a differential equation for  $u*(t)$  expressed in terms of only the drift vector and its derivatives. The derivation can be explained as follows. Let the angle  $\mu(t)$  be given by  $u_1(t) = V_{\max} \cos \chi(t)$  and  $u_2(t) = V_{\max} \sin \chi(t)$  then

$$\cos \chi(t) = \frac{\lambda_1}{\|\lambda\|}, \quad \sin \chi(t) = \frac{\lambda_2}{\|\lambda\|}$$

Differentiating these relations, the following equalities can be obtained:

$$\begin{aligned}\cos \chi \frac{d\|\lambda\|}{dt} - \|\lambda\| \sin \chi \frac{d\chi}{dt} &= \frac{d\lambda_1}{dt} = -\lambda_1 \frac{\partial W_x}{\partial x} - \lambda_2 \frac{\partial W_y}{\partial x}, \\ \sin \chi \frac{d\|\lambda\|}{dt} + \|\lambda\| \cos \chi \frac{d\chi}{dt} &= \frac{d\lambda_2}{dt} = -\lambda_1 \frac{\partial W_x}{\partial y} - \lambda_2 \frac{\partial W_y}{\partial y}.\end{aligned}$$

Finally, the Zermelo navigation equation is given by:

$$\frac{d\chi}{dt} = -\cos^2 \chi \frac{\partial W_x}{\partial y} + \sin \chi \cos \chi \left( \frac{\partial W_x}{\partial x} - \frac{\partial W_y}{\partial y} \right) + \sin^2 \chi \frac{\partial W_y}{\partial x}. \quad [11.5]$$

#### 11.3.2.2. One particular solution

When the problem is to find minimum-time paths through a 2D region of position-dependent velocity vector [JAR 01]:

$$\begin{aligned}\dot{x} &= V \cos \chi + W_x(x, y), \\ \dot{y} &= V \sin \chi + W_y(x, y),\end{aligned} \quad [11.6]$$

the heading angle is the control available for achieving the minimum time objective.

Let  $V_w$  be the wind velocity, a constant. If  $W_x(x, y) = \mp V_w y$  and  $W_y(x, y) = 0$ , where  $V_w$  is the wind velocity, a constant, it has been proved in [JAR 01] that

$$y = \frac{V}{V_w} \left( \frac{1}{\sin \chi} - \frac{1}{\sin \chi_f} \right) \quad [11.7]$$

$$x = \frac{V}{2V_w} \left( \operatorname{asinh}(\tan \chi_f) - \operatorname{asinh}(\tan \chi) \right. \quad [11.8]$$

$$\left. + \tan \chi \left( \frac{1}{\sin \chi} - \frac{1}{\sin \chi_f} \right) \right) \quad [11.9]$$

$$- \frac{1}{\cos \chi_f} (\tan \chi_f - \tan \chi). \quad [11.10]$$

where  $V_w$  is a positive constant, i.e. the wind velocity magnitude. The evolution of the heading angle  $\chi$  in [11.5] is reduced to

$$\dot{\chi} = -\cos^2(\chi) \frac{\partial W_x}{\partial y}, \quad [11.11]$$

$$\dot{\chi} = \pm V_w \cos^2(\chi). \quad [11.12]$$

The time to go is given by

$$T = \frac{1}{V_w} (\tan \chi_f - \tan \chi). \quad [11.13]$$

### 11.3.3. Zermelo navigation problem: case 3D

Now, we consider the wind time optimal trajectory planning problem for an aerial vehicle in a 3D space. A rotorcraft must travel through a region of strong winds. The magnitude and the direction of the winds are known to be functions of position, i.e.  $W_x = W_x(x, y, z)$ ,  $W_y = W_y(x, y, z)$ , and  $W_z = W_z(x, y, z)$ , where  $(x, y, z)$  are the rectangular coordinates and  $(W_x, W_y, W_z)$  are the velocity components of the wind. The aircraft velocity relative to the air mass is  $V$ , a constant. We wish to find the minimum-time path from point  $A$  to point  $B$ .

The kinematic model of the aircraft is given by

$$\dot{x} = V \cos \chi \cos \gamma + W_x, \quad [11.14]$$

$$\dot{y} = V \sin \chi \cos \gamma + W_y, \quad [11.15]$$

$$\dot{z} = V \sin \gamma + W_z \quad [11.16]$$

where  $\chi$  is the heading angle of the aircraft relative to the inertial frame and  $\gamma$  is the flight path angle.

The Hamiltonian of the system is given by

$$\begin{aligned} H = & \lambda_x(V \cos \chi \cos \gamma + W_x) + \lambda_y(V \sin \chi \cos \gamma + W_y) \\ & + \lambda_z(V \sin \gamma + W_z) + 1. \end{aligned} \quad [11.17]$$

The Euler–Lagrange equations are given by

$$\dot{\lambda}_x = -\frac{\partial H}{\partial x} = -\lambda_x \frac{\partial W_x}{\partial x} - \lambda_y \frac{\partial W_y}{\partial x} - \lambda_z \frac{\partial W_z}{\partial x}, \quad [11.18]$$

$$\dot{\lambda}_y = -\frac{\partial H}{\partial y} = -\lambda_x \frac{\partial W_x}{\partial y} - \lambda_y \frac{\partial W_y}{\partial y} - \lambda_z \frac{\partial W_z}{\partial y}, \quad [11.19]$$

$$\dot{\lambda}_z = -\frac{\partial H}{\partial z} = -\lambda_x \frac{\partial W_x}{\partial z} - \lambda_y \frac{\partial W_y}{\partial z} - \lambda_z \frac{\partial W_z}{\partial z}, \quad [11.20]$$

$$0 = \frac{\partial H}{\partial \chi}, \quad [11.21]$$

$$0 = \frac{\partial H}{\partial \gamma}. \quad [11.22]$$

By taking the partial derivative of the Hamiltonian with respect to the heading angle (equation [11.21]), we have

$$\begin{aligned} 0 = & -\lambda_x V \sin \chi \cos \gamma + \lambda_y V \cos \chi \cos \gamma, \\ \lambda_x \sin \chi = & \lambda_y \cos \chi, \end{aligned} \quad [11.23]$$

$$\tan \chi = \frac{\lambda_y}{\lambda_x}. \quad [11.24]$$

In order to obtain an expression for  $\lambda_x$  and  $\lambda_y$ , we use [11.23] and [11.17]

$$-\lambda_x \sin \chi + \lambda_y \cos \chi = 0, \quad [11.25]$$

$$\lambda_x = \lambda_y \frac{\cos \chi}{\sin \chi}. \quad [11.26]$$

By taking the partial derivative of the Hamiltonian with respect to the flight path angle (equation [11.22]), we have

$$\begin{aligned} 0 &= -\lambda_x V \cos \chi \sin \gamma - \lambda_y V \sin \chi \sin \gamma + \lambda_z V \cos \gamma, \\ 0 &= \sin \gamma (-\lambda_x \cos \chi - \lambda_y \sin \chi) + \lambda_z \cos \gamma. \end{aligned} \quad [11.27]$$

Introducing [11.26] into the previous equation, we get

$$\begin{aligned} 0 &= \sin \gamma (-\lambda_y \frac{\cos \chi}{\sin \chi} \cos \chi - \lambda_y \sin \chi) + \lambda_z \cos \gamma, \\ 0 &= -\lambda_y \frac{\sin \gamma}{\sin \chi} + \lambda_z \cos \gamma, \\ \lambda_y \frac{\sin \gamma}{\sin \chi} &= \lambda_z \cos \gamma, \\ \lambda_z &= \lambda_y \frac{\sin \gamma}{\sin \chi \cos \gamma}. \end{aligned} \quad [11.28]$$

Introducing [11.26] and [11.28] relations into [11.17], and taking into account that  $H = 0$  due to the fact that  $H$  is not a function of time, which implies that  $\dot{H} = \text{const.}$ ; since we are minimizing time, this constant must be zero:

$$\begin{aligned} \left( \frac{\lambda_y \cos \chi}{\sin \chi} \right) (V \cos \chi \cos \gamma + W_x) + \lambda_y (V \sin \chi \cos \gamma + W_y) \\ + \left( \lambda_y \frac{\sin \gamma}{\sin \chi \cos \gamma} \right) (V \sin \gamma + W_z) + 1 = 0, \\ \lambda_y = \frac{-\sin \chi \cos \gamma}{V + W_x \cos \chi \cos \gamma + W_y \sin \chi \cos \gamma + W_z \sin \gamma}. \end{aligned} \quad [11.29]$$

Then,

$$\begin{aligned} \lambda_x &= \frac{-\sin \chi \cos \gamma}{V + W_x \cos \chi \cos \gamma + W_y \sin \chi \cos \gamma + W_z \sin \gamma} \left( \frac{\cos \chi}{\sin \chi} \right), \\ \lambda_x &= \frac{-\cos \chi \cos \gamma}{V + W_x \cos \chi \cos \gamma + W_y \sin \chi \cos \gamma + W_z \sin \gamma}, \end{aligned}$$

and

$$\lambda_z = \lambda_y \frac{\sin \gamma}{\sin \chi \cos \gamma} \quad [11.30]$$

$$\lambda_z = \lambda_x \frac{\sin \gamma}{\cos \chi \cos \gamma} \quad [11.31]$$

$$\lambda_z = \frac{-\sin \gamma}{V + W_x \cos \chi \cos \gamma + W_y \sin \chi \cos \gamma + W_z \sin \gamma}.$$

To simplify the computations in the following, we define  $\Lambda = V + W_x \cos \chi \cos \gamma + W_y \sin \chi \cos \gamma + W_z \sin \gamma$ , then

$$\lambda_x = \frac{-\cos \chi \cos \gamma}{\Lambda}, \quad [11.32]$$

$$\lambda_y = \frac{-\sin \chi \cos \gamma}{\Lambda}, \quad [11.33]$$

$$\lambda_z = \frac{-\sin \gamma}{\Lambda}. \quad [11.34]$$

Introducing [11.32]–[11.34] into [11.18]–[11.20], we have

$$\dot{\lambda}_x = \frac{\cos \chi \cos \gamma}{\Lambda} \frac{\partial W_x}{\partial x} + \frac{\sin \chi \cos \gamma}{\Lambda} \frac{\partial W_y}{\partial x} + \frac{\sin \gamma}{\Lambda} \frac{\partial W_z}{\partial x}, \quad [11.35]$$

$$\dot{\lambda}_y = \frac{\cos \chi \cos \gamma}{\Lambda} \frac{\partial W_x}{\partial y} + \frac{\sin \chi \cos \gamma}{\Lambda} \frac{\partial W_y}{\partial y} + \frac{\sin \gamma}{\Lambda} \frac{\partial W_z}{\partial y}, \quad [11.36]$$

$$\dot{\lambda}_z = \frac{\cos \chi \cos \gamma}{\Lambda} \frac{\partial W_x}{\partial z} + \frac{\sin \chi \cos \gamma}{\Lambda} \frac{\partial W_y}{\partial z} + \frac{\sin \gamma}{\Lambda} \frac{\partial W_z}{\partial z} \quad [11.37]$$

Now, we can use [11.32]–[11.34] to obtain an expression that describes the evolution of the heading. Recall that  $H = 0$  and  $\dot{H} = 0$ , then differentiating [11.25] with respect to time, we obtain

$$-\dot{\lambda}_x \sin \chi - \lambda_x \cos \chi \dot{\chi} + \dot{\lambda}_y \cos \chi - \lambda_y \sin \chi \dot{\chi} = 0,$$

$$\dot{\chi} = \frac{-\dot{\lambda}_x \sin \chi + \dot{\lambda}_y \cos \chi}{\lambda_x \cos \chi + \lambda_y \sin \chi}.$$

Substituting  $\lambda_x$ ,  $\lambda_y$  and  $\dot{\lambda}_x$ ,  $\dot{\lambda}_y$ , we have

$$\begin{aligned} \dot{\chi} &= \sin^2 \chi \frac{\partial W_y}{\partial x} + \sin \chi \cos \chi \left( \frac{\partial W_x}{\partial x} - \frac{\partial W_y}{\partial y} \right) \\ &\quad + \sin \gamma \sec \gamma \left( \sin \chi \frac{\partial W_z}{\partial x} - \cos \chi \frac{\partial W_z}{\partial y} \right) - \cos^2 \chi \frac{\partial W_x}{\partial y}. \end{aligned} \quad [11.38]$$

Now, we can use [11.32]–[11.34] to obtain an expression that describes the evolution of the heading. Recalling that  $H = 0$  and  $\dot{H} = 0$ , and then differentiating [11.27] with respect to time, we obtain

$$-\dot{\lambda}_x \sin \gamma \cos \chi - \lambda_x (-\sin \gamma \sin \chi \dot{\chi} + \cos \chi \cos \gamma \dot{\gamma}) \quad [11.39]$$

$$-\dot{\lambda}_y \sin \gamma \sin \chi - \lambda_y (\sin \gamma \cos \chi \dot{\chi} + \sin \chi \cos \gamma \dot{\gamma}) \quad [11.40]$$

$$+\dot{\lambda}_z \cos \gamma + \lambda_z (-\sin \gamma \dot{\gamma}) = 0. \quad [11.41]$$

Substituting [11.32]–[11.34] into previous equation, we have

$$\dot{\gamma} = \dot{\lambda}_x \sin \gamma \cos \chi + \dot{\lambda}_y \sin \gamma \sin \chi - \dot{\lambda}_z \cos \gamma. \quad [11.42]$$

Substituting  $\dot{\lambda}_x, \dot{\lambda}_y, \dot{\lambda}_z$ , we have

$$\begin{aligned} \dot{\gamma} = & \cos^2 \chi \cos \gamma \sin \gamma \frac{\partial W_x}{\partial x} + \sin \chi \cos \gamma \sin \gamma \cos \chi \frac{\partial W_y}{\partial x} + \sin^2 \gamma \cos \chi \frac{\partial W_z}{\partial x} \\ & [11.43] \end{aligned}$$

$$\begin{aligned} & + \cos \chi \cos \gamma \sin \gamma \sin \chi \frac{\partial W_x}{\partial y} + \sin^2 \chi \cos \gamma \sin \gamma \frac{\partial W_y}{\partial y} + \sin^2 \gamma \sin \chi \frac{\partial W_z}{\partial y} \\ & [11.44] \end{aligned}$$

$$-\cos \chi \cos^2 \gamma \frac{\partial W_x}{\partial z} + \sin \chi \cos^2 \gamma \frac{\partial W_y}{\partial z} + \sin \gamma \cos \gamma \frac{\partial W_z}{\partial z}. \quad [11.45]$$

### 11.3.3.1. Constant wind

From [11.38] note that, if the  $W_x, W_y$ , and  $W_z$  are constants, then it implies that  $\chi = \text{const.}$ , i.e. the minimum-time path are straight lines.

### 11.3.3.2. Linear variation of wind velocity

If we have  $W_x = \pm V_w y$  and  $W_y = W_z = 0$ , then [11.38] is reduced to

$$\begin{aligned} \dot{\chi} &= -\cos^2 \chi \frac{\partial W_x}{\partial y} \\ \dot{\chi} &= -\cos^2 \chi (\pm V_w) \\ \dot{\chi} &= \mp V_w \cos^2 \chi \end{aligned} \quad [11.46]$$

and

$$\begin{aligned} \dot{\gamma} &= \sin \gamma \sin \chi \cos \gamma \cos \chi \frac{\partial W_x}{\partial y} \\ \dot{\gamma} &= \sin \gamma \sin \chi \cos \gamma \cos \chi (\pm V_w) \\ \dot{\gamma} &= \pm V_w \sin \gamma \sin \chi \cos \gamma \cos \chi. \end{aligned} \quad [11.47]$$

### 11.4. Quadrotor formation control scheme

Considering planar movement, the translational kinematics of the center of mass of the multiagent system taking into account the wind effect can thus be expressed by the following equations:

$$\dot{x}_{CM} = V \cos \chi + W_x, \quad [11.48]$$

$$\dot{y}_{CM} = V \sin \chi + W_y, \quad [11.49]$$

$$\dot{\chi} = -V_w \cos^2(\chi), \quad [11.50]$$

where  $\chi$  is the heading angle of the aircraft relative to the inertial frame.

The goal is to find a minimum-time path for a platoon of rotorcrafts flying in formation through a region of variable winds. The typical Zermelo navigation problem considers a 2D trajectory planning. In this section, we show that a 3D Zermelo navigation approach can be used to solve the formation flying through regions of variable winds.

### 11.5. Quadrotor trajectory-tracking control

Using the center of mass of a multivehicle system for trajectory-tracking purposes is a common technique that ensures a good global performance. We will consider the case of trajectory tracking of a multiple rotorcraft system. For simplicity, it is assumed that the leader of the group is always vehicle 1. Define,

$$x_{CM} = \frac{1}{N} \sum_{i=1}^N x_i,$$

where  $N$  is the number of quadrotor in the formation. Let  $x_{CM}^d$  be the desired value for  $x_{CM}$ . Thus, the control for the center of mass,  $u_{1_x}$ , can be stated as

$$u_{1_x} = N k_{CM} \sigma(x_{CM}^d - x_{CM}), \quad [11.51]$$

where  $k_{CM}$  is a positive gain. Note that  $x_{CM}$  may not be directly measurable for the leader (vehicle 1). We assume that the topology of information exchange is controllable and observable from the input and output of the leader. The state can therefore be observed from the input and output of the leader.

Then, the trajectory-tracking control for the leader of the group is given by

$$\tau_{\theta_1} = -\sigma_4(\dot{\theta}_i) - \sigma_3(\theta_i) - \sigma_2(\dot{x}_i) - \sigma_1 \left( \sum_{j \in \mathcal{N}_i} (x_j - x_i) - x_i^d - u_{1_x} \right), \quad [11.52]$$

$$\tau_{\phi_1} = -\sigma_4(\dot{\phi}_i) - \sigma_3(\phi_i) - \sigma_2(\dot{y}_i) - \sigma_1 \left( \sum_{j \in \mathcal{N}_i} (y_j - y_i) - y_i^d - u_{1_y} \right). \quad [11.53]$$

Note that wind time-optimal paths usually have the form of time-varying references. When the reference for a leader-based multirobot system is varying in time there is a small bias in agents' coordination. A simple solution would be to use the double-integrator multivehicle system discussed in Chapter 3

$$\dot{\xi} = -\mathcal{L}\xi + b\bar{u} \quad [11.54]$$

where  $\mathcal{L}$  is the Laplacian matrix of information exchange, and  $\xi$  and control law are defined as

$$\xi_i \triangleq \dot{x}_i + \kappa x_i \quad [11.55]$$

$$\tilde{u}_l \triangleq \tilde{u}_{CM} \quad [11.56]$$

$$\tilde{u}_i \triangleq \dot{\xi}_j \text{ for some } j \in \mathcal{N}_i \quad [11.57]$$

where  $u_l$  is the input given to the leader,  $u_i$  is the input given to the  $i$ th vehicle, and  $\tilde{u}_{CM} = k_{CM}\sigma_{M_{CM}}(\xi_{CM} - \xi_{CM}^d)$ .

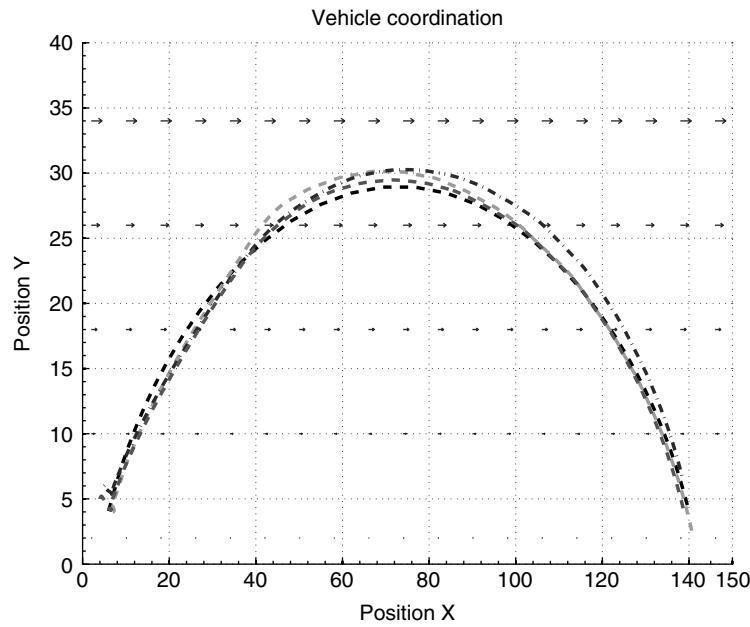
Note that the selection of the control  $\tilde{u}_i$  depends on the structure of the information-exchange topology.

## 11.6. Simulation results

To validate the proposed algorithm, extensive simulations were run in Matlab/Simulink®. A simplified nonlinear dynamic model of the quadrotor has been used in simulations. The wind optimal trajectory is obtained by using a pseudospectral collocation method on the resulting heading program and dynamic model for the center of mass. The Cartesian coordinates of the optimal path are then introduced in a multiple-quadrotor system with cyclic topology of information exchange as the desired  $x^d$  and  $y^d$  positions along time. The initial positions for the quadrotor were set to  $\mathbf{P}_1^0 = [6 \ 4 \ 0]$  m,  $\mathbf{P}_2^0 = [5 \ 6 \ 0]$  m, and  $\mathbf{P}_3^0 = [4 \ 5 \ 0]$  m, and the initial and final waypoints were set to  $\mathbf{P}_s = [6 \ 4 \ 1]$  m, and  $\mathbf{P}_f = [140 \ 4 \ 1]$  m. We have considered a region of strong winds with linearly varying behavior  $\mathbf{W} = [-V_w W y \ 0 \ 0]$ . The wind velocity  $V_w$  has been set to 0.022 m/s. In this work, we consider the following cases.

### 11.6.1. Reference given to leader vehicle

First, we consider the case in which only the leader has access to the reference. Figure 11.2 shows the trajectory tracking of a group of three quadrotor flying in formation. The reference has been given only to the leader. It can be observed that the platoon achieves formation as it follows the optimal trajectory. As the platoon gets closer to stronger winds the formation losses rigidity due to the influence of linearly varying wind.



**Figure 11.2.** Planar view of a quadrotor platoon following a time-optimal trajectory in a region of strong winds. Platoon evolving in linearly varying winds

Note that giving the reference only to the leader implies a delay in the propagation of the information along the communication network. As a consequence of this, the vehicles subject to stronger winds than the others, cannot compensate their position on time.

Figures 11.3 and 11.4 show the coordination and tracking of the decoupled lateral and longitudinal subsystems.

Table 11.1 lists the values of nonlinear control and parameters used in simulations.

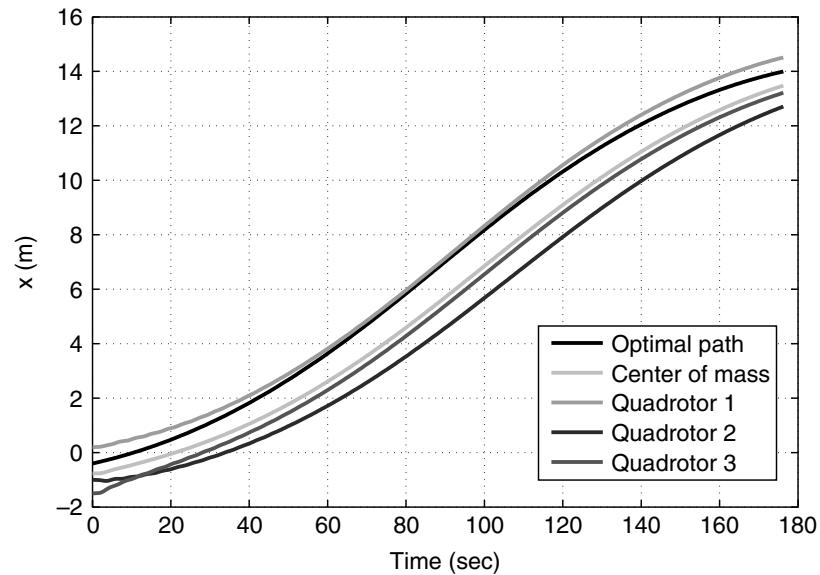


Figure 11.3. Longitudinal coordination

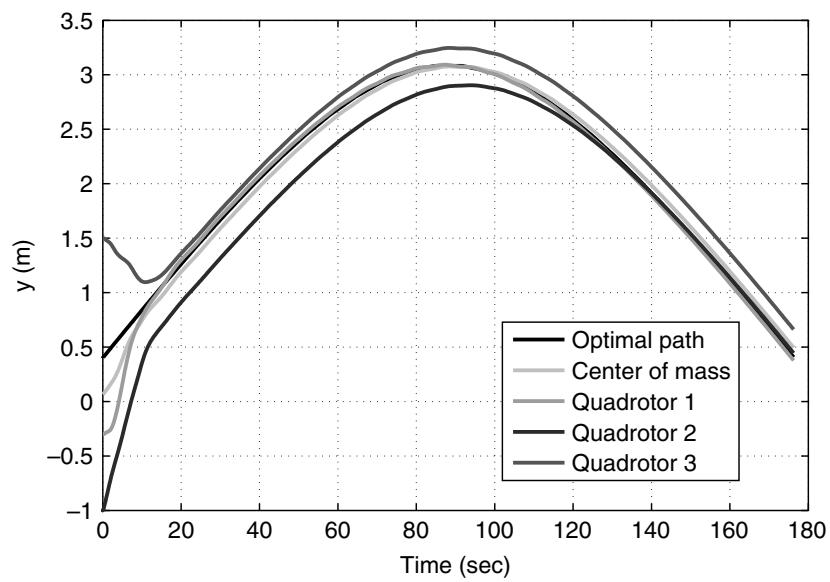
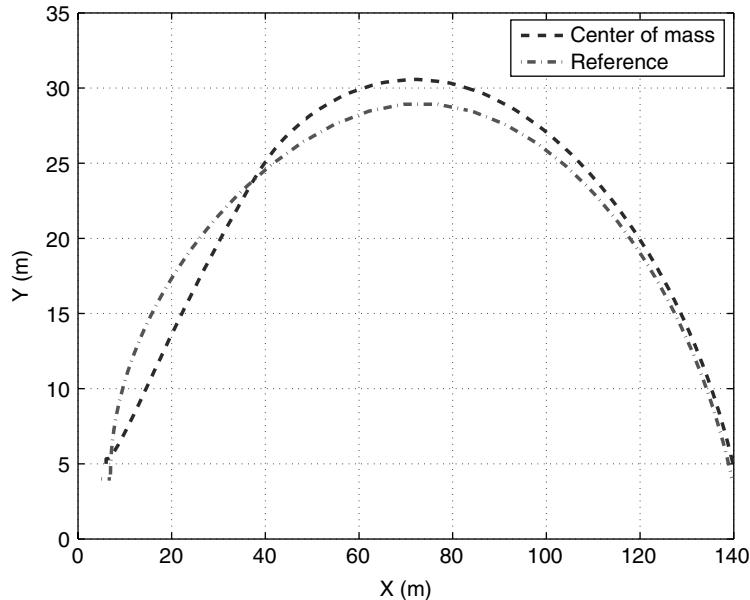


Figure 11.4. Lateral coordination



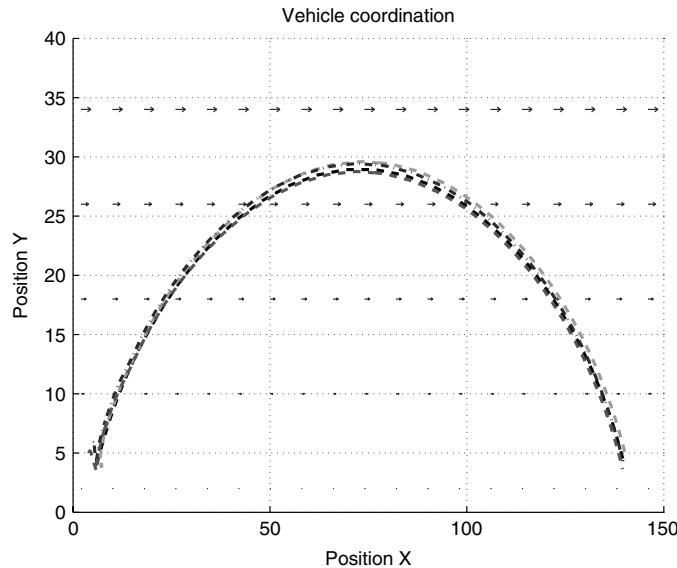
**Figure 11.5.** Planar view of a quadrotor platoon following a time-optimal trajectory in a region of strong winds. Platoon evolving in linearly varying winds

Parameter	Lateral	Longitudinal
Position	0.5	0.5
Velocity	2	2
Altitude	4	4
Angular rate	4	4
Center of mass	1.5	1.5
Vel. center of mass	14	5

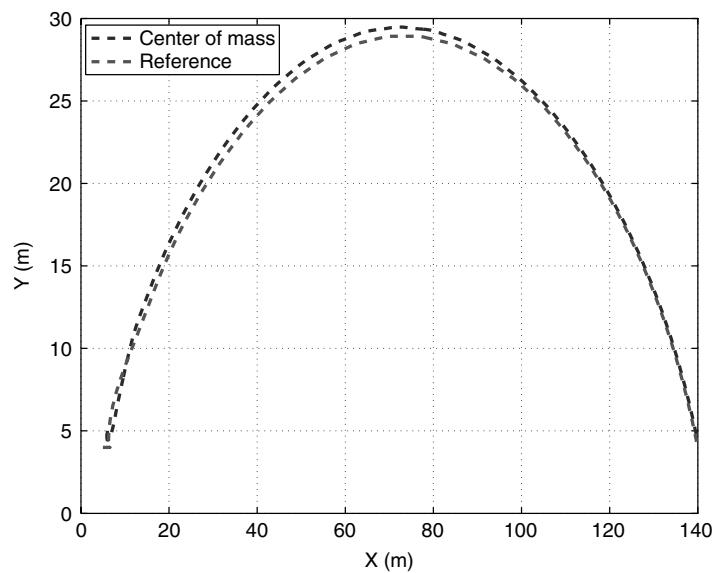
**Table 11.1.** Control parameters

### 11.6.2. Reference given to all vehicles

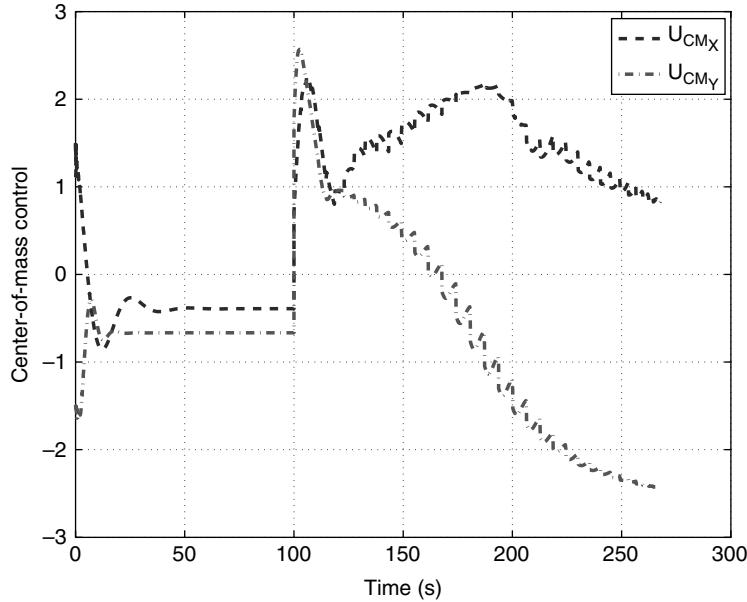
Figure 11.6 shows the trajectory tracking of a group of three quadrotor flying in formation with reference which has been given to every vehicle in the formation. It can be observed that the platoon achieves formation as it tracks the optimal trajectory. As the platoon gets closer to stronger winds, the formation maintains its structural stiffness. By giving the reference to every quadrotor in the formation, the vehicles subject to stronger winds compensate their position error with respect to the group and the optimal trajectory.



**Figure 11.6.** Planar view of a quadrotor platoon following a time-optimal trajectory in a region of strong winds. Platoon evolving in linearly varying winds



**Figure 11.7.** Planar view of the center of mass following the time-optimal trajectory in a region of strong winds



**Figure 11.8.** Center-of-mass control inputs

### 11.7. Conclusions and future work

A nonlinear control based on separated saturations and a forced consensus control to track a time-varying reference for flight formation of miniature rotorcraft was developed. The  $x$ -position and the  $y$ -position of each miniature rotorcraft were considered as dynamical agents with full information access. A Zermelo navigation approach has been adopted to determine the optimal path in a region of strong winds. Trajectory tracking for the group of miniature rotorcraft was achieved by using the virtual center of mass of the agents' formation. Extensive simulations were run in order to show the performance of the developed control scheme. Future work in this area includes experimental tests on miniature rotorcraft using real-time embedded control systems.

### 11.8. Bibliography

- [ARR 49] ARROW K., "On the use of winds in flight planning", *Journal of Meteorology*, vol. 6, pp. 150–159, 1949.
- [BEA 01] BEARD R., LAWTON J., HADAEGH F., "A coordination architecture for spacecraft formation control", *IEEE Transactions on Control Systems Technology*, vol. 9, no. 6, pp. 777–790, 2001.

- [BES 11] BESTAOUI Y., KAHALE E., “Time optimal trajectories for an autonomous airship”, *IEEE Workshop on Robot Motion Control (ROMOCO 2011)*, 2011.
- [BOU 04] BOUABDALLAH S., NOTH A., SIEGWART R., “PID vs LQ control Techniques Applied to an Indoor Micro Quadrotor”, in *Proceedings of the IEEE/RSJ International Conference on the Intelligent Robots and Systems*, Sendai, Japan, pp. 2451–2456, 2004.
- [BRY 75] BRYSON A., HO Y., *Applied Optimal Control: Optimization, Estimation and Control*, Taylor and Francis, 1975.
- [CAS 05] CASTILLO P., LOZANO R., DZUL A., “Stabilization of a mini-rotorcraft having four rotors”, *IEEE Control Systems Magazine*, vol. 25, no. 6, pp. 45–55, 2005.
- [ERG 07] ERGINER B., ALTUG E., “Modeling and PD control of a quadrotor VTOL vehicle”, *IEEE Intelligent Vehicles Symposium*, pp. 894–899, Istanbul, 2007.
- [GUE 10a] GUERRERO J., FANTONI I., SALAZAR S., LOZANO R., “Flight formation of multiple mini rotorcraft via coordination control”, *IEEE International Conference on Robotics and Automation*, Anchorage, pp. 620–625, 2010.
- [GUE 10b] GUERRERO J., LOZANO R., “Flight formation of multiple mini rotorcraft based on nested saturations”, *IEEE International Conference on Intelligent Robots and Systems (IROS 2010)*, Taipei, pp. 634–639, 2010.
- [HOK 07] HOKAYEM P., STIPANOVIC D., SPONG M., “Reliable control of multi-agent formations”, *IEEE American Control Conference*, New York, pp. 1882–1887, 2007.
- [ISI 03] ISIDORI A., MARCONI L., SERRANI A., “Robust nonlinear motion control of a helicopter”, *IEEE Transactions on Automatic Control*, vol. 48, no. 3, pp. 413–426, 2003.
- [JAR 01] JARDIN M., BRYSON A., “Neighboring optimal aircraft guidance in winds”, *Journal of Guidance, Control, and Dynamics*, vol. 24, no. 4, 2001.
- [LAC 03] LA CIVITA M., PAPAGEORGIOU G., MESSNER W., KANADE T., “Design and Flight testing of a gain-scheduled  $H_1\infty$  loop shaping controller for wide-envelope flight of a robotic helicopter”, *IEEE American Control Conference*, Denver, pp. 4195–4200, 2003.
- [LAR 06] LARA D., SANCHEZ A., LOZANO R., CASTILLO P., “Real time embedded control system for VTOL aircrafts: application to stabilize a quad-rotor helicopter”, *IEEE Conference on Control Applications*, Munich, pp. 2553–2558, 2006.
- [LEE 03] LEE D., LI P., “Formation and maneuver control of multiple spacecraft”, *IEEE American Control Conference*, Denver, pp. 278–283, 2003.
- [LEE 06] LEE D., SPONG M., “Flocking of multiple inertial agents on balanced graph”, *IEEE American Control Conference*, Minneapolis, pp. 1469–1475, 2006.
- [LOZ 07] LOZANO R., *Objets Volants Miniatures: Modélisation et Commande Embarquée*, Hermès-Lavoisier, 2007.
- [MIC 09] MICHAEL N., FINK J., KUMAR V., “Cooperative manipulation and transportation with aerial robots”, *Robotics: Science and Systems*, 2009.
- [OLF 06] OLFATI-SABER R., “Flocking for multi-agent dynamic systems: algorithms and theory”, *IEEE Transactions on Automatic Control*, vol. 51, no. 3, pp. 401–420, 2006.

- [OUN 10] OUNG R., BOURGAULT F., DONOVAN M., D'ANDREA R., “The distributed flight array”, *IEEE International Conference on Robotics and Automation*, Anchorage, pp. 601–607, 2010.
- [REN 07] REN W., “Consensus seeking in multi-vehicle systems with a time varying reference state”, *IEEE American Control Conference*, New York, pp. 717–722, 2007.
- [TAN 03a] TANNER H., JADBABAIE A., PAPPAS G., “Stable flocking of mobile agents, part I: fixed topology”, *IEEE Conference on Decision and Control*, Maui, pp. 2010–2015, 2003.
- [TAN 03b] TANNER H., JADBABAIE A., PAPPAS G., “Stable flocking of mobile agents, part II: dynamic topology”, *IEEE Conference on Decision and Control*, Maui, pp. 2016–2021, 2003.
- [ZER 30] ZERMELO E., “Über die Navigation in der Luft als Problem der Variationsrechnung”, *Jahresbericht der Deutschen Mathematiker-Vereinigung*, vol. 39, 1930.

## Chapter 12

# Impact of Wireless Medium Access Protocol on the Quadrotor Formation Control

In this chapter, the impact of the medium access protocols on the average consensus problem over wireless networks for a group of quadrotors is established. We study the case of a group of quadrotors communicating over a wireless network considering both directed and undirected graphs of information flow. It turns out that the media access control (MAC) protocols have a direct impact in both convergence time and average consensus solution, i.e. the solution of the average consensus is no longer the average of the initial conditions. It will be shown that the solution for the average consensus problem over a wireless network depends directly on the MAC algorithm. Simulations are provided to demonstrate the theoretical results.

### 12.1. Introduction

In recent years, much attention has been directed toward multiple robot systems with global or local communications, such as swarming [GAZ 03b, GAZ 03a], consensus [REN 07, POR 07, GUE 10b], flocking [OLF 06, TAN 03a, TAN 03b, LEE 07], formations [FAX 04, LAF 05, DUN 06], etc. Different approaches have been proposed in the literature for coordination of multiple autonomous robot systems such as leader/follower [CHE 06, KRI 06], virtual structure [LEO 01, BEA 01], and behavioral control [ARR 06, BAL 98]. In the particular case of a multiagent (multiple aerial vehicle) consensus problem, much of the literature is focused on modeling the information flow network using either fixed or switching topologies. This means that

---

Chapter written by J.A. GUERRERO, Y. CHALLAL and P. CASTILLO.

the communication network does not suffer from a time delay and from packet drop problems. The main difference between fixed and switching topologies is that the switching topology case takes into account that every aerial vehicle has a limited range of interaction with its neighbors, i.e. the mobility of the aerial vehicles affects the information exchange topology.

The distributed nature of a multiple robot control system over wireless networks represents an interesting research problem. Data loss, data corruption, and time delay over lossy network are key factors that may lead to performance degradation and even cause instability. Recent work on networked control over noisy communication channels includes [SAH 04] and [TAK 04]. In most of the cases, the packet drop phenomena are modeled as a random process without any specification of its probabilistic distribution [LIN 09, ZHA 10]. In [HU 07], the authors consider the packet drop process as a Bernoulli process and develop stability conditions under these conditions. Another way to model the packet loss phenomena has been described in [LIN 03], where a Markov chain has been used to model the packet dropout process.

This chapter addresses the issue of packet loss as well as packet delay in a multiagent aerial system considering a wireless network. For this end, we use the widely adopted network simulator (NS2) that provides a rich simulation environment modeling the different network communication layers: the physical layer (modulation, frequencies, signal and radio propagation models, wired and wireless channels, etc.), link layer (different MAC algorithms: TDMA, CSMA/CD, CSMA/CA, ALOHA, etc.), routing layer (routing protocols over wired and wireless networks as well as mobile ad hoc networks), transport layer (TCP, UDP, RTP, etc.), and the application layer with a rich sample of applications for typical traffic generation scenarios. It will be assumed, for simplicity, that every agent in the multivehicle system broadcasts its information to its neighbors considering a fixed topology of information exchange, i.e. the mobility of the vehicles does not affect the information flow network. It is assumed that the neighbors of the  $i$ th quadrotor are always within the  $i$ th quadrotor broadcasting range. Different network MAC protocols have been designed for wireless networks. Likewise, in this chapter, we analyze the impact of the following algorithms on the multiple quadrotor average consensus problem: carrier sense multiple access with collision avoidance (CSMA/CA) and time-division multiple access (TDMA).

This chapter is organized as follows: a brief description of the multiquadrotor dynamics and control over perfect communication links are introduced in section 12.2. In section 12.3, the multiquadrotor average consensus over wireless network analysis is presented. A stability analysis to improve the convergence of the consensus is proposed in section 12.4. Simulation results are illustrated in section 12.5. Finally, conclusion and future work are discussed in section 12.6.

## 12.2. Multiquadrotor consensus

### 12.2.1. Quadrotor dynamic model and control

Let us consider a group of  $N$ -quadrotor helicopters with the following dynamical model [LOZ 10, GUE 10a]:

$$\ddot{x}_i = -\mathbf{F}_i \sin(\theta_i) \quad [12.1]$$

$$\ddot{y}_i = \mathbf{F}_i \cos(\theta_i) \sin(\phi_i) \quad [12.2]$$

$$\ddot{z}_i = \mathbf{F}_i \cos(\theta_i) \cos(\phi_i) - 1 \quad [12.3]$$

$$\ddot{\phi}_i = \tau_{\phi,i} \quad [12.4]$$

$$\ddot{\theta}_i = \tau_{\theta,i} \quad [12.5]$$

$$\ddot{\psi}_i = \tau_{\psi,i} \quad [12.6]$$

where  $\mathbf{F}_i$  is the thrust force vector,  $(x_i, y_i, z_i)$  and  $(\phi_i, \theta_i, \psi_i)$  represent, respectively, the position and orientation of the  $i$ th quadrotor.

It was proved in [LOZ 10] that the following control strategies stabilize the previous system:

$$\mathbf{F}_i = \frac{-k_{1,i}\dot{z} - k_{2,i}(z_i - z_i^d) + 1}{\cos(\phi_i) \cos(\theta_i)} \quad [12.7]$$

$$\tau_{\psi,i} = -k_{3,i}\dot{\psi}_i - k_{4,i}(\psi_i - \psi_i^d) \quad [12.8]$$

$$\begin{aligned} \tau_{\theta,i} = & -\sigma_4(\dot{\theta}_i + \sigma_3(\dot{\theta}_i + \theta_i + \sigma_2(\dot{\theta}_i + 2\theta_i \\ & - \dot{x}_i + \sigma_1(\dot{\theta}_i + 3\theta_i - 3\dot{x}_i - x_i)))) \end{aligned} \quad [12.9]$$

$$\begin{aligned} \tau_{\phi,i} = & -\sigma_4(\dot{\phi}_i + \sigma_3(\dot{\phi}_i + \phi_i + \sigma_2(\dot{\phi}_i + 2\phi_i + \dot{y}_i \\ & + \sigma_1(\dot{\phi}_i + 3\phi_i + 3\dot{y}_i + y)))) \end{aligned} \quad [12.10]$$

where  $k_{1,i}$ ,  $k_{2,i}$ ,  $k_{3,i}$ , and  $k_{4,i}$  are positive constants;  $z^d$  and  $\psi^d$  are the desired altitude and heading for the  $i$ th quadrotor, respectively. Observe that the nonlinear control laws [12.7]–[12.10] guarantee the stabilization of the  $i$ th quadrotor in closed-loop system, such that

$$\lim_{t \rightarrow \infty} z_i = z^d$$

$$\lim_{t \rightarrow \infty} \psi_i = \psi^d$$

$$\lim_{t \rightarrow \infty} x_i = 0$$

$$\lim_{t \rightarrow \infty} y_i = 0$$

### 12.2.2. From individual to collective behavior

In order to model the interactions among helicopters, a graph-based theoretical approach has been considered. Lee and Spong [LEE 07] have established that the kinematic model of the  $x$ -position for a group of quadrotors can also be written as:

$$\dot{x}_i = \bar{u}_i \quad \forall i = 1, \dots, n;$$

with leader-based multiquadrotor consensus achieved using the following algorithm:

$$\bar{u}_i = - \sum_{j \in \mathcal{N}_i} (x_i - x_j) \quad [12.11]$$

where  $\mathcal{N}_i$  is the set of vehicles transmitting their information to the quadrotor  $i$ . Note that [12.11] ensures the consensus agreement in the sense of  $\lim_{t \rightarrow \infty} |x_i - x_j| = 0$ .

Hence, the position consensus among quadrotors yields

$$\dot{\mathbf{x}} = -\mathcal{L}\mathbf{x} \quad [12.12]$$

where  $\mathcal{L}$  is the Laplacian matrix of the information exchange graph, for more details see [GUE 10b] and [GUE 10a].

Therefore, the controllers [12.7]–[12.10] can be improved to the case of multiquadrotor consensus with the form

$$\mathbf{F}_i = \frac{-a_{1,i}\dot{z} - a_{2,i}(\sum_{j \in \mathcal{N}_i} (z_i - z_j) - z^d) + 1}{\cos(\phi_i) \cos(\theta_i)} \quad [12.13]$$

$$\tau_{\psi,i} = -a_{3,i}\dot{\psi}_i - a_{4,i} \left( \sum_{j \in \mathcal{N}_i} (\psi_i - \psi_j) - \psi^d \right) \quad [12.14]$$

$$\begin{aligned} \tau_{\theta,i} = & -\sigma_4 \left( \dot{\theta}_i + \sigma_3 \left( \dot{\theta}_i + \theta_i + \sigma_2 \left( \dot{\theta}_i + 2\theta_i - \dot{x}_i \right. \right. \right. \\ & \left. \left. \left. + \sigma_1 \left( \dot{\theta}_i + 3\theta_i - 3\dot{x}_i - \left( \sum_{j \in \mathcal{N}_i} (x_i - x_j) \right) \right) \right) \right) \right) \end{aligned} \quad [12.15]$$

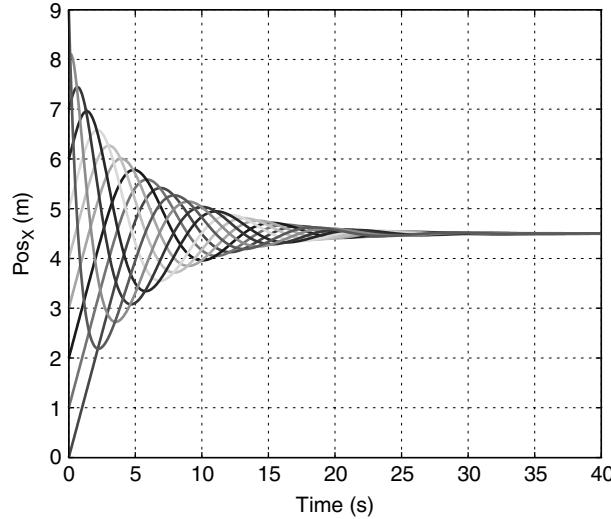
$$\begin{aligned} \tau_{\phi,i} = & -\sigma_4 \left( \dot{\phi}_i + \sigma_3 \left( \dot{\phi}_i + \phi_i + \sigma_2 \left( \dot{\phi}_i + 2\phi_i + \dot{y}_i \right. \right. \right. \\ & \left. \left. \left. + \sigma_1 \left( \dot{\phi}_i + 3\phi_i + 3\dot{y}_i - \left( \sum_{j \in \mathcal{N}_i} (y_i - y_j) \right) \right) \right) \right) \right) \end{aligned} \quad [12.16]$$

which implies that

$$\begin{aligned}\lim_{t \rightarrow \infty} |(z_j - z_i)| &= z^d \\ \lim_{t \rightarrow \infty} |(\psi_j - \psi_i)| &= \psi_i^d \\ \lim_{t \rightarrow \infty} |(x_j - x_i)| &= 0 \\ \lim_{t \rightarrow \infty} |(y_j - y_i)| &= 0\end{aligned}$$

In order to illustrate the performance of the previous control strategy, simulations were carried out, over the  $x$ -axis, considering a 10-quadrotor platoon over a perfect communication network (no delay, no packet loss) with cyclic topology. The initial conditions are  $x_i(0) = i$ ,  $\forall i = 0, 1, \dots, 9$ .

Figure 12.1 shows the consensus response. Observe in this figure that the convergence time is small and the solution of the average consensus over the  $x$ -axis is the average of the initial conditions, i.e. the platoon achieves consensus to 4.5. Thus, the previous nonlinear control laws guarantee the position synchronization of the quadrotor platoon, see [GUE 10a].



**Figure 12.1.** Multiple quadrotor consensus using perfect communication

In literature, several simulations using the previous (similar or different) controllers have been carried out in order to prove the stability of the consensus or of the flight formation trajectory, see [FAX 04], [LAF 05], [DUN 06], and [CHE 06].

Most of the assumptions in these works are considering wireless and perfect communication between the vehicles. In addition, the main results in flight formation are, in general, illustrated only in simulations.

As it is known, in a real multivehicle flight formation system, each aircraft collects information from its sensors and then exchange's its information, employing wireless communication, with other autonomous vehicles in the network. Packet delay and packet loss in wireless become major issues of study that must be taken into account when stabilizing multiaerial vehicles. The goal of this work is to prove, in simulations, the impact of the wireless network communication in a multivehicle consensus.

### **12.3. Multiagent consensus over wireless networks**

Let us consider the case of an  $N$ -quadrotor formation flying over a wireless communication channel. From the automatic control point of view, the key factors when using wireless channels are: end-to-end time delay, packet dropout rate, network connectivity, and noise. Let us assume that the mobility of the agents does not affect the network connectivity and neglect the noise from sensors. Then, we focus our attention on the packet dropout rate and the end-to-end time delay problems. It has been shown in the communications' literature [LIA 01], [LIU 04] that both packet dropout rate and the end-to-end time delay are determined by the MAC protocols. Our study considers two common technologies for wireless communications: CSMA/CA used in the IEEE 802.11 and TDMA used in GSM.

#### **12.3.1. CSMA/CA**

This is a distributed random access algorithm used in many standards such as WiFi IEEE 802.11. This scheme uses a carrier sense multiple access with collision avoidance (CSMA/CA) mechanism for resolving the problem of access to the communication medium. This implies that when a node detects a collision, it stops transmitting and waits for a random time before retransmitting. More precisely, the protocol CSMA/CA works as follows [ATI 07]:

- 1) a carrier sensing scheme is used;
- 2) a data station that intends to transmit sends a jam signal;
- 3) after waiting a sufficient time for all stations to receive the jam signal, the data station transmits a frame;
- 4) while transmitting, if the data station detects a jam signal from another station, it stops transmitting for a random time and then tries again.

We distinguish two variants: with and without channel reservation. In CSMA/CA with channel reservation, the transmitter first transmits a Request To Send (RTS) for channel reservation and waits for to receive a Clear To Send (CTS) from the destination before starting to send the data frame.

CSMA/CA is not suitable for real-time communications, since it does not guarantee an upper bound for the delay before sending a data frame.

### **12.3.2. TDMA**

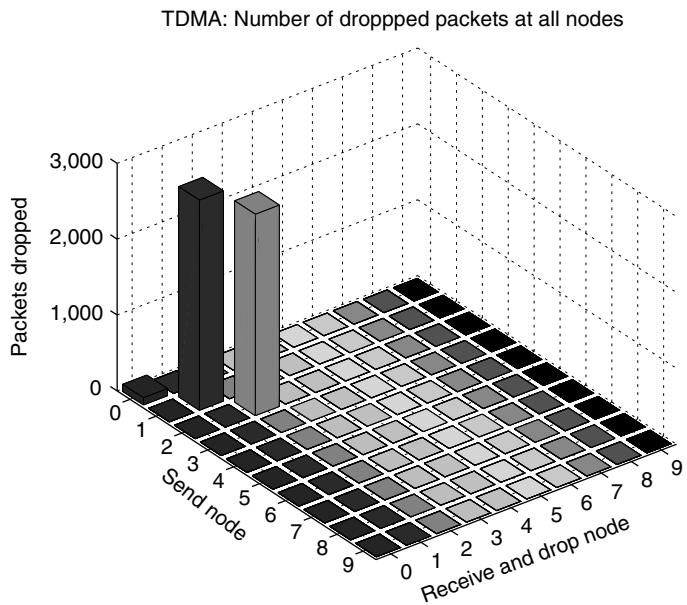
This is a time-slotted scheme used in many standards such as GSM. According to [ATI 07], TDMA is a collision-free multiple access technique whereby users share a transmission medium by being assigned and using (one at a time) time slots assigned previously.

TDMA is more suitable for real-time communications, since it guarantees an upper bound delay before transmitting a data frame.

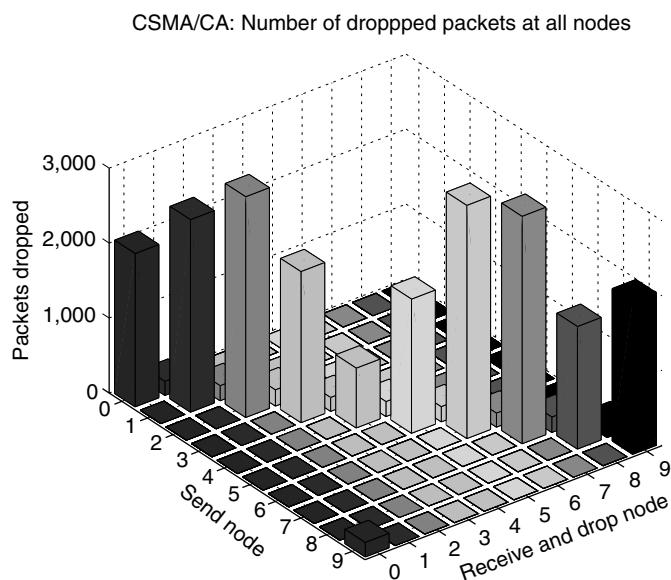
### **12.3.3. Network analysis**

To evaluate the performance of CSMA/CA and TDMA protocols on the quadrotor consensus problem, extensive simulations have been run using the network simulator NS2. As it is shown in Figures 12.2 and 12.3, it is clear that the packet drops for TDMA protocol are almost null. We attribute the packet drops shown in Figure 12.2 to the synchronization phase during the simulation initialization. Unlike TDMA, CSMA/CA shows a higher rate of packet drops due to its random broadcasting nature (see Figure 12.3).

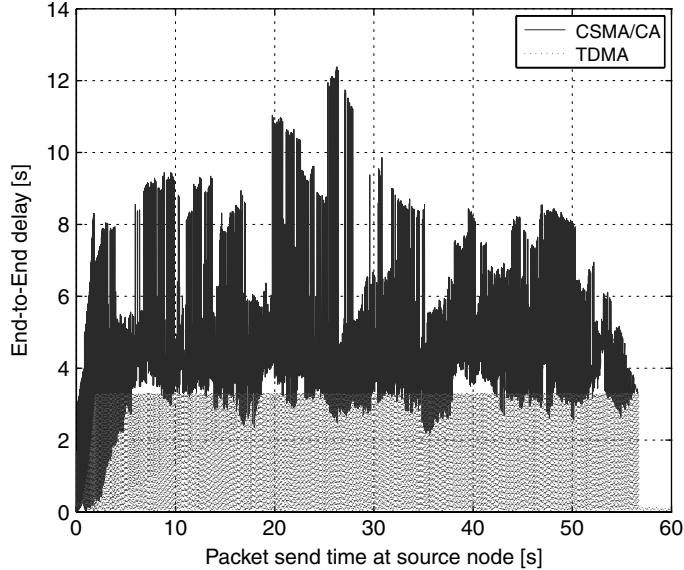
Now, let us analyze the average end-to-end delay which gives an insight into the expected time delay on multiquadrotor systems. It is known that in real-time applications, after a large time delay, data may become useless. Therefore, it is important to analyze the performance of MAC protocols such that the multiquadrotor system will undergo the minimal time delay. On the one hand, since TDMA is a time-division multiple access technique to access the transmission medium, it is almost intuitive that the time delay should show an almost constant rate. This can be confirmed by observing Figure 12.4. On the other hand, taking into account the packet drop rate at each quadrotor, it is expected that CSMA/CA would show a variable time delay and the maximum time delay is much higher than TDMA. The evolution of the average end-to-end time delay over time for the CSMA/CA protocol is shown in Figure 12.4. It is worth mentioning that the end-to-end time delay has been obtained at agent trace level, i.e. from application to application layer.



**Figure 12.2.** Packet drop using TDMA; simulation time = 60 s



**Figure 12.3.** Packet drop using CSMA/CA; simulation time = 60 s



**Figure 12.4.** End-to-end delay comparison between CSMA/CA and TDMA; simulation time = 60 s

#### 12.4. Quadrotor consensus over wireless networks

In this section, we can introduce a network-aware average consensus control for a multiquadrotor system over a wireless network. By taking into account the phenomena discussed above, the packet dropout process will be considered as a source of time delays in the wireless network. As shown in [LIA 01], the end-to-end time delay  $\tau$  is given by the difference  $\tau_{dst} - \tau_{src}$ , which, in turn, depends on the preprocessing time  $\tau_{pre}$ , wait time  $\tau_{wait}$ , transmission time  $\tau_{tx}$ , and the post-processing time  $\tau_{post}$ . Since the nearest-neighbor approach assumes that each vehicle communicates only with immediate neighbors that are in its radio range, and assuming that a packet loss implies the retransmission of the data that have been lost, we consider the packet dropout process, as part of the transmission time delay which has been defined as the frame time  $\tau_{frame}$  and the propagation time  $\tau_{prop}$ ; for more details, see [LIA 01].

Then, we propose to use the following multiquadrotor consensus control as in [CHO 06]:

$$\bar{u}_i(t) = - \sum_{j \in \mathcal{N}_i} (x_i - x_j(t - \tau_{ji})) \quad [12.17]$$

Using the control law [12.17] instead of [12.11], the quadrotor consensus control is given as:

$$\mathbf{F}_i = \frac{-k_{1,i}\dot{z}_i - k_{2,i}(\sum_{j \in \mathcal{N}_i}(z_j(t - \tau_{ji}) - z_i) - z^d) + 1}{\cos(\phi_i)\cos(\theta_i)} \quad [12.18]$$

$$\tau_{\psi,i} = -k_{3,i}\dot{\psi}_i - k_{4,i}\left(\sum_{j \in \mathcal{N}_i}(\psi_j(t - \tau_{ji}) - \psi_i) - \psi^d\right) \quad [12.19]$$

$$\begin{aligned} \tau_{\theta,i} = & -\sigma_4\left(\dot{\theta}_i + \sigma_3\left(\dot{\theta}_i + \theta_i + \sigma_2\left(\dot{\theta}_i + 2\theta_i - \dot{x}_i\right.\right.\right. \\ & \left.\left.\left. + \sigma_1\left(\dot{\theta}_i + 3\theta_i - 3\dot{x}_i - \sum_{j \in \mathcal{N}_i}(x_j(t - \tau_{ji}) - x_i)\right)\right)\right)\right) \end{aligned} \quad [12.20]$$

$$\begin{aligned} \tau_{\phi,i} = & -\sigma_4\left(\dot{\phi}_i + \sigma_3\left(\dot{\phi}_i + \phi_i + \sigma_2\left(\dot{\phi}_i + 2\phi_i + \dot{y}_i\right.\right.\right. \\ & \left.\left.\left. + \sigma_1\left(\dot{\phi}_i + 3\phi_i + 3\dot{y}_i - \sum_{j \in \mathcal{N}_i}(y_j(t - \tau_{ji}) - y_i)\right)\right)\right)\right) \end{aligned} \quad [12.21]$$

Let us consider the Lyapunov function as in [CHO 06]

$$V = \sum_{i=1}^N \sum_{j \in N} \int_{t-\tau_{ji}}^t x_j^T(\chi)x_j(\chi)d\chi + 2(V_1 + \dots + V_N) \quad [12.22]$$

where  $V_i$  is the storage function for each quadrotor vehicle in the platoon. Differentiating [12.22], we have

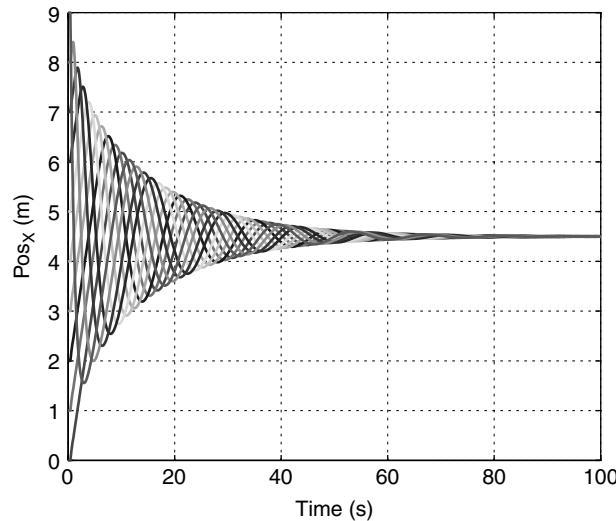
$$\dot{V} = -2 \sum_{i=1}^N S_i(x_i) - \sum_{i=1}^N \sum_{j \in N_i} (x_j(t - \tau_{ji}) - x_i)^T (x_j(t - \tau_{ji}) - x_i) \quad [12.23]$$

From [12.23], we have that

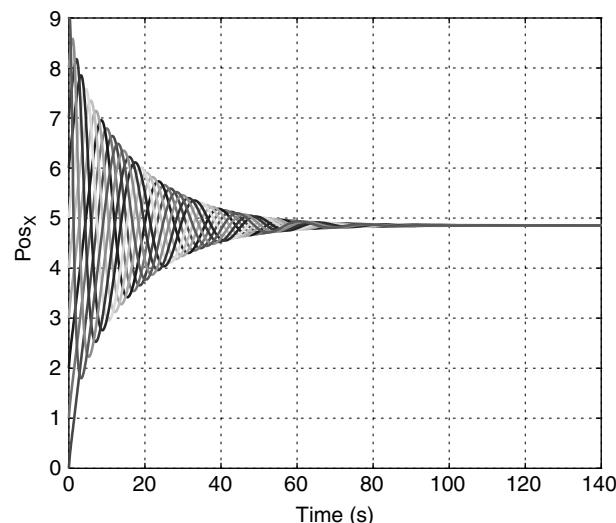
$$\lim_{t \rightarrow \infty} |(x_j(t - \tau_{ji}) - x_i)| = 0 \quad [12.24]$$

From [12.24], it is clear that the solution of the average consensus to the average of the initial conditions depends on the network medium access algorithm, which, in turn, determines the values of the  $\tau_{ji}$  for each quadrotor. Figure 12.5 shows the performance of the average consensus considering  $\tau_{ji} = \tau_{kj}$ . It can be observed that the average consensus is the same as the one for the perfect communication case. The difference is that the convergence time for the equal time delay case is larger than for the perfect communication case. Figure 12.6 shows the performance of the average consensus considering  $\tau_{ji} \neq \tau_{kj}$ . It can be observed that the solution of average consensus is

different from the previous two cases: perfect communication and equal time delays. Based on the law of large numbers, the average of the results from a large number of simulations should be closed to the expected value. Figure 12.6 shows the consensus over time of the average of 1,000 simulations.



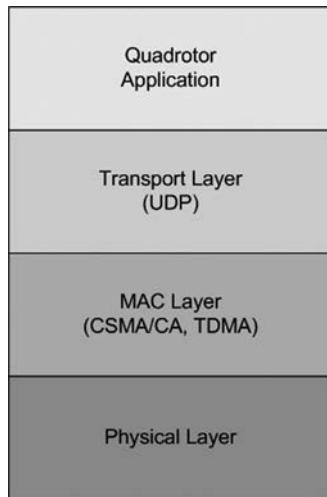
**Figure 12.5.** Multiple quadrotor consensus with  $\tau_{ji} = \tau_{kj}$



**Figure 12.6.** Multiple quadrotor with  $\tau_{ji} \neq \tau_{kj}$

### 12.5. Simulation results

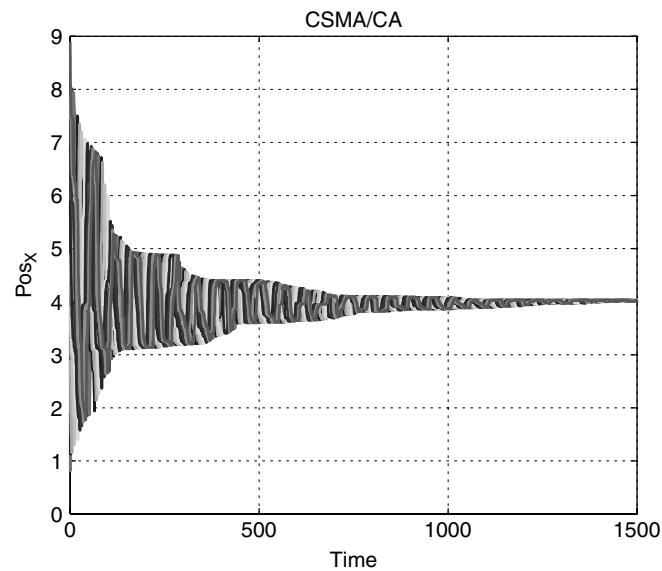
Now, let us take a look at the average consensus performance using a wireless network. The network scenario consisted of 10 quadrotors sharing information over a cyclic topology. A quadrotor application was developed, a UDP transport protocol was modified to exchange information with the quadrotor application and the MAC layer, as shown in Figure 12.7. The following MAC algorithms: CSMA/CA and TDMA were analyzed using the network simulator NS2. Figures 12.8 and 12.9 show the performance of the MAC protocols CSMA/CA and TDMA, respectively.



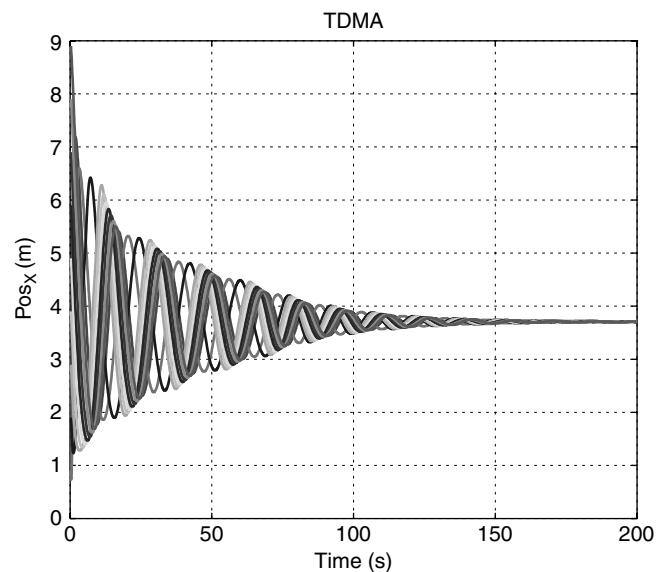
**Figure 12.7.** *Multiple quadrotor wireless network stack*

From Figures 12.8 and 12.9, it is possible to observe that the solution for the average consensus as well as its convergence time is affected by the network MAC algorithm. It can be concluded that the average consensus using a CSMA/CA protocol presents a larger convergence time due to both packet drops and time delay in the end-to-end transmission among quadrotors. Also, due to the fact that the access to the transmission medium is assigned randomly, it is evident that some quadrotors will transmit their positions to their neighbors before their counterparts. This implies that the difference between any two  $\tau_{ji}$  can be large enough such that some quadrotors will evolve faster than the others. Then, the quadrotor with the smallest  $\tau_{ji}$  will update its position at the highest rate than the others. Figure 12.10 shows the performance of the average consensus for a small platoon of four quadrotors using different MAC algorithms such as: CSMA/CA, TDMA, constant time delay ( $\tau_{ji} > 0$ ), and perfect communication. The simulation results are summarized in Table 12.1.

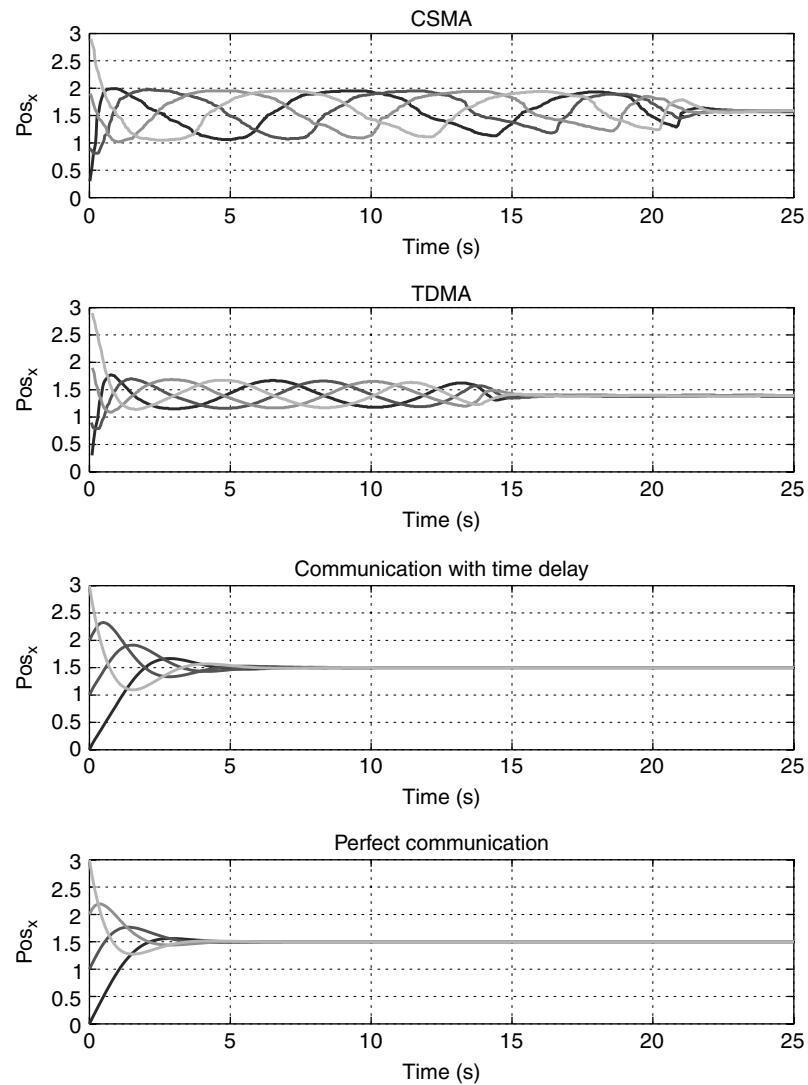
Figure 12.11 shows the performance of the average consensus for the heading subsystem of a two-quadrotor system using CSMA/CA.



**Figure 12.8.** Multiagent consensus using IEEE 802.11

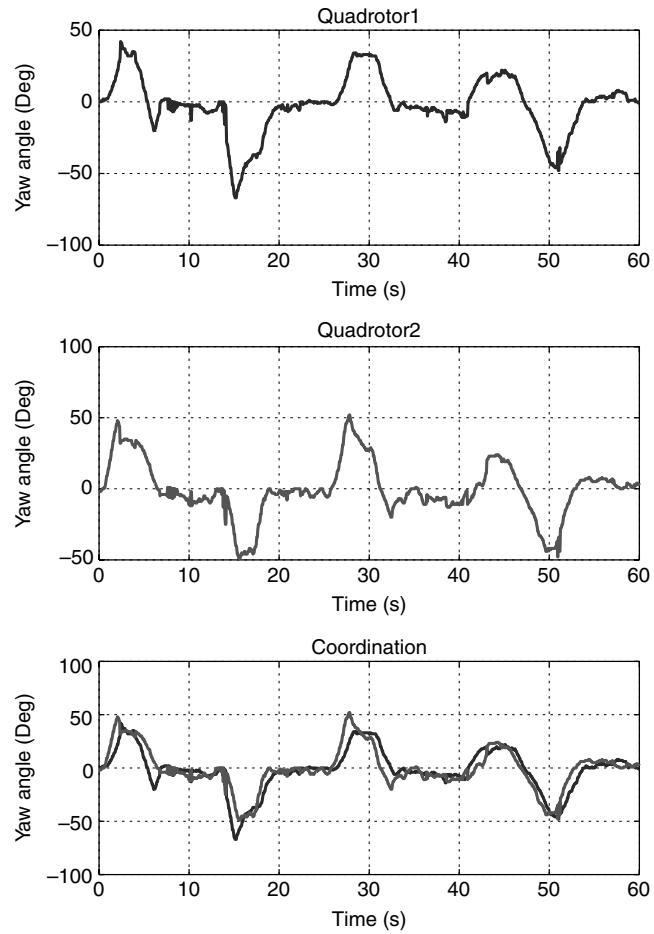


**Figure 12.9.** Multiagent consensus using TDMA



**Figure 12.10.** Comparison of convergence time for different MAC algorithms

MAC algorithm	Max E2E time delay	Convergence time
CSMA/CA	~ 12	~ 1200
TDMA	~ 3.2	~ 200
Perfect communication	0	~ 40

**Table 12.1.** Performance comparison**Figure 12.11.** Average consensus for the yaw subsystem. An external reference was given to the system in order to show the delay over the wireless network

## 12.6. Conclusions and future work

A nonlinear control based on nested saturations and a single integrator consensus control considering time delay for flight formation of mini rotorcraft was developed. We remark that TDMA is more suitable for real-time communications than CSMA/CA, since it guarantees an upper bound delay before transmitting a data frame. CSMA/CA is less suitable for real-time communications, since it does not guarantee an upper bound for the delay before sending a data frame. Extensive simulations were run in order to show the performance of the developed control scheme. Future work in this area includes experimental tests on mini rotorcraft using real-time embedded control systems, and work to develop an optimized MAC algorithm for mobile robot systems.

## 12.7. Bibliography

- [ATI 07] ALLIANCE FOR TELECOMMUNICATIONS INDUSTRY SOLUTIONS, *Telecom Glossary*, 2007.
- [ARR 06] ARRICELLO F., CHIAVERINI S., FOSSEN T., “Formation control of marine vessels using the null-space-based behavioral control”, *LNCIS 336 Group Coordination and Cooperative Control*, Springer-Verlag, pp. 1–19, 2006.
- [BAL 98] BALCH T., ARKIN R., “Behavior-based formation control for multirobot teams”, *IEEE Transactions on Robotics and Automation*, vol. 14, no. 6, pp. 926–939, 1998.
- [BEA 01] BEARD R., LAWTON J., HADAEGH F., “A coordination architecture for spacecraft formation control”, *IEEE Transactions on Control Systems Technology*, vol. 9, no. 6, pp. 777–790, 2001.
- [CHE 06] CHEN X., SERRANI A., “ISS-based robust leader/follower trailing control”, *LNCIS 336 Group Coordination and Cooperative Control*, Springer-Verlag, pp. 75–91, 2006.
- [CHO 06] CHOPRA N., Output synchronization of mechanical systems, PhD Thesis, UIUC, 2006.
- [DUN 06] DUNBAR W., MURRAY R., “Distributed receding horizon control for multi-vehicle formation”, *Automatica*, vol. 42, no. 4, pp. 549–558, 2006.
- [FAX 04] FAX J., MURRAY R., “Information flow and cooperative control on vehicle formations”, *IEEE Transactions on Automatic Control*, vol. 49, no. 9, pp. 1465–1476, 2004.
- [GAZ 03a] GAZI V., “Stability of an asynchronous swarm with time-dependent communication links”, *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 38, no. 1, pp. 267–274, 2003.
- [GAZ 03b] GAZI V., PASSINO K., “Stability analysis of swarms”, *IEEE Transactions on Automatic Control*, vol. 48, no. 4, pp. 692–697, 2003.
- [GUE 10a] GUERRERO J., FANTONI I., SALAZAR S., LOZANO R., “Flight formation of multiple mini rotorcraft via coordination control”, *IEEE International Conference on Robotics and Automation*, Anchorage, pp. 634–639, 2010.

- [GUE 10b] GUERRERO J., ROMERO G., LOZANO R., “Robust consensus tracking of leader-based multi-agent systems”, *IEEE American Control Conference*, Baltimore, pp. 6299–6305, 2010.
- [HU 07] HU S., YAN W.-Y., “Stability robustness of networked control systems with respect to packet loss”, *Automatica*, vol. 43, no. 7, pp. 1243–1248, 2007.
- [KRI 06] KRISTIANSEN R., LORIA A., CHAILLET A., NICKLASSON P., “Output feedback control of relative translation in a leader–follower spacecraft formation”, *LNCIS 336 Group Coordination and Cooperative Control*, Springer-Verlag, pp. 131–150, 2006.
- [LAF 05] LAFFERRIERE G., WILLIAMS A., CAUGHMAN J., VEERMAN J., “Decentralized control of vehicle formations”, *Systems and Control Letters*, vol. 54, no. 9, pp. 899–910, 2005.
- [LEE 07] LEE D., SPONG M., “Stable flocking of multiple inertial agents on balanced graphs”, *IEEE Transactions on Automatic Control*, vol. 52, no. 4, pp. 1469–1475, 2007.
- [LEO 01] LEONARD N., FIORELLI E., “Virtual leaders, artificial potentials and coordinated control of groups”, *IEEE Conference on Decision and Control*, Orlando, pp. 2968–2973, 2001.
- [LIA 01] LIAN F.-L., MOYNE J., TILBURY D., “Performance evaluation of control networks: ethernet, controlnet and devicenet”, *IEEE Control Systems Magazine*, vol. 21, no. 1, pp. 66–83, 2001.
- [LIN 03] LING Q., LEMMON M., “Soft real-time scheduling of networked control systems with dropouts governed by a Markov chain”, *IEEE American Control Conference*, pp. 4845–4850, 2003.
- [LIN 09] LIN W., ZHIXIN L., “Robust consensus of multi-agent systems with noise”, *Sci. China Ser. F-Inf. Sci.*, vol. 52, no. 5, pp. 824–834, 2009.
- [LIU 04] LIU X., GOLDSMITH A., “Wireless medium access control in networked control systems”, *IEEE American Control Conference*, pp. 3605–3610, 2004.
- [LOZ 10] LOZANO R., *Unmanned Aerial Vehicles*, ISTE Ltd., London and John Wiley and Sons, New York, 2010.
- [OLF 06] OLFATI-SABER R., “Flocking for multi-agent dynamic systems: algorithms and theory”, *IEEE Transactions on Automatic Control*, vol. 51, no. 3, pp. 401–420, 2006.
- [POR 07] PORFIRI M., ROBERSON D., STILWELL D., “Tracking and formation control of multiple autonomous agents: a two-level consensus approach”, *Automatica*, vol. 43, no. 48, pp. 1318–1328, 2007.
- [REN 07] REN W., “Consensus seeking in multi-vehicule systems with a time varying reference state”, *IEEE American Control Conference*, New York, pp. 717–722, 2007.
- [SAH 04] SAHAI A., “The necessity and sufficiency of anytime capacity for control over noisy communication link”, *IEEE Conference on Decision and Control*, pp. 1896–1901, 2004.
- [TAK 04] TAKIKONDA S., MITTER S., “Control over noisy channels”, *IEEE Transactions on Automatic Control*, vol. 49, no. 7, pp. 1196–1201, 2004.

- [TAN 03a] TANNER H., JADBABAIE A., PAPPAS G., “Stable flocking of mobile agents, part I: fixed topology”, *IEEE Conference on Decision and Control*, Maui, pp. 2010–2015, 2003.
- [TAN 03b] TANNER H., JADBABAIE A., PAPPAS G., “Stable flocking of mobile agents, part II: dynamic topology”, *IEEE Conference on Decision and Control*, Maui, pp. 2016–2021, 2003.
- [ZHA 10] ZHANG W., WANG Z., GOU Y., “Robust consensus for uncertain multi-agent systems on directed communication topologies”, *IEEE Conference on Decision and Control*, New Jersey, pp. 6317–6322, 2010.

## Chapter 13

# MAC Protocol for Wireless Communications

### 13.1. Introduction

Wireless networks are also deployed for various purposes such as environmental monitoring, industrial application, transportation, home systems automation, control of distributed embedded systems (such as unmanned aerial vehicles, or UAVs), and even medical services. These networks are interconnected systems of devices that are capable of communication, computation, data storage, and adaptation in a distributed fashion [MOL 11].

In the design of wireless networks, one critical issue is the medium access control (MAC) protocol. A great concern and problem in wireless networks is the collision, and this is caused by two terminals sending data at the same time over the same transmission medium. Therefore, a MAC protocol plays an important role in enabling normal network operation and achieving good network performance [MEN 09].

MAC has been studied in wireless networks. A variety of MAC protocols have been proposed to address different network scenarios. From different perspectives, MAC protocols can be classified into different categories, for example, centralized and distributed, single channel based and multiple channel based, contention based and contention free, and so on.

MAC is critical for enabling successful network operation in all shared medium networks [COV 05]. The primary task for a MAC protocol is to arbitrate access to a shared medium or channel in order to avoid collision and at the same time to fairly

---

Chapter written by A. MENDEZ, M. PANDURO, O. ELIZARRARAS and D. COVARRUBIAS.

and efficiently share the bandwidth resources among multiple terminals. According to the underlying control mechanism for collision avoidance (CA), MAC protocols can be typically classified into two broad categories: contention-based and contention-free [ZHE 09].

The protocol should be able to deal with the following conditions [PRA 98]:

- *Hidden-terminal problem*: This problem occurs when two senders are not in the vicinity of each other (so cannot sense each other's signals) but both of them are in the range of the common receiver. So carrier sensing fails in this case.

- *Exposed-terminal problem*: Exposed terminals are complementary to hidden terminals. An exposed terminal is one that is within the range of the sender but out of range of the destination.

- *Co-channel interference*: This problem refers to the interference from two different terminals on the same frequency. It is one of the major limitations in cellular networks.

- *Capture*: This occurs when the received power at the receiver from two senders is significantly different. The sender with the higher power captures the receiver, which will never be able to sense the second signal. This leads to significant fairness problems.

- The problem of fading (multipath and shadowing).

We should mention that many of the properties listed above are conflicting, so we must come to a compromise in the design of the properties of multiple access protocol.

Moreover, the tendency of the MAC protocols for wireless networks is using adaptive systems to adjust the transmission parameters and the objective is to maximize the throughput in the use of the channel. In addition, in the MAC protocols the low throughput in the region of low traffic is because there is no more information to send and is not due to errors of the multiuser interference [MEN 10]. We can say that the system performance is limited by the access technique used in wireless networks. In order to improve the performance (throughput) of a wireless network, we use a MAC protocol based on code-division multiple access (CDMA) with multirate scheme. In this scheme, we control the transmission rate using transmission threshold, and handling the spreading factors for a MAC scheme based on carrier sense multiple access (CSMA)–CDMA wireless networks.

The aim of this chapter is to provide details of the protocols that define and control access to the wireless channel, i.e. wireless MAC protocols. We start with the analysis of the protocols: slotted ALOHA (S-ALOHA), CSMA, and inhibit sense multiple access (ISMA). In addition, the results evaluation of these protocols are presented in section 13.2. Section 13.3 describes the operation of a proposed protocol based on CSMA–CDMA. Section 13.4 presents the performance evaluation and its corresponding results. Finally, section 13.5 provides conclusions for this chapter.

### 13.2. Protocols of medium access control

#### 13.2.1. Slotted ALOHA

In this process, we consider a random access channel (RACH) divided in time slots. The terminals use this RACH to transmit service requests to the base station, soliciting resources (as number of required slots). The base station will acknowledge those terminals with service requests granted.

The terminal carries out a request through an RACH, through which the terminal notifies to the base station the communication requirements, i.e. the type of service and the number of packets to transmit. With this information, the base station plans the assignment of resources of the system to different requests. This planning is carried out dynamically on a slot-by-slot basis. After the request, the terminals establish a procedure to achieve a contention-free transmission. It is worth emphasizing that given the present random characteristics in a wireless communication scenario, it is necessary to count on a MAC technique that can efficiently cope with the possible interference among terminals which use, simultaneously, the RACH channel directed toward the same base station, i.e. the MAC technique should help prevent and solve these problems, as well as optimize the request of the channel in order to have high throughput.

##### 13.2.1.1. Modeling of S-ALOHA

As mentioned before, the first interaction between the terminal and the base station is given in the access request stage (RACH channel), where a collision of request packets will occur every time if two or more terminals try to transmit or retransmit [CAR 75]. In this situation, the system sets the terminals involved in the collision in a backlog state. We model the state of the system as the number of backlogged terminals and idle terminals in each slot, and we also control the transmission probability.

The initial stage of the analysis and modeling of the RACH channel is made on a slot-by-slot basis, where the slots of the system are numbered sequentially,  $k = 0, 1, 2, \dots$ . Moreover,  $\eta_s(k)$  denote the number of backlogged terminals at the beginning of the  $k$ th slot, the random variable  $\eta_s(k)$  is referred to as the state of the system. The number of terminals in backlog at the beginning of the  $(k + 1)$ th slot depends on the number of terminals in backlog at the beginning of the  $k$ th slot, and the number of terminals going from one state to another within the slot. Owing to the fact that this is independent of the activities in any previous slot, the process could be represented by a Markov chain modeled by a birth-death process.

According to the analysis done in [ROM 90], the process of retransmission and transmission of each terminal for a finite number of terminals,  $M$ , is an independent geometric process, in which the probability that  $i$  out of the  $j$  backlogged terminals program a retransmission in a single slot represents a binomial distribution, with

probability that a terminal retransmits a packet  $\nu$  and probability that a terminal generates a new packet  $\varphi$ . By this means, the steady-state transition probabilities  $p_{ij} = \lim_{x \rightarrow \infty} \Pr(\eta_s(k) = j \mid \eta_s(k-1) = i)$  are obtained and the transition matrix  $P$  is formed [BOL 06]. The steady-state probability vector  $\pi$  whose elements are  $\pi_j$  is the solution to the finite set of linear equations  $\pi = \pi \cdot P$ , and  $\sum_{i=0}^M \pi_i = 1$  [ROM 90, BOL 06].

In order to evaluate the throughput of the system, it is considered that all intention of transmission agrees with the beginning of each slot and that the activity in any given slot is independent of the activity in any previous slot. Taking this into consideration, the fraction of time that a channel transports useful information or throughput,  $S$ , is equal to the average fraction of slots in any successful transmission.

To have successful transmission, there should be a single transmission in the slot. This indicates that all of the terminals in backlog are in silence and that only a new terminal transmits, or only one terminal in backlog transmits while there is no new packet generated. So the probability of success when  $i$  terminals are in backlog state is given by:

$$P_{succ}(i) = (1-\nu)^i (M-i) \varphi (1-\varphi)^{M-i-1} + i\nu (1-\nu)^{i-1} (1-\varphi)^{M-i}, \quad [13.1]$$

and the throughput  $S$  is expressed as:

$$S = E[P_{succ}(i)] = \sum_{i=0}^M P_{succ}(i) \cdot \pi_i. \quad [13.2]$$

Up to now, the S-ALOHA modeling, carried out as request channel in an environment of wireless communication systems, has been considered with a noiseless channel in which all the packets can reach the receiver with the same power level as presented in [CAR 75]. Under these conditions, when two or more packets reach the receiver simultaneously, they will collide and all of the packets involved will be destroyed. On the other hand, in a real radio channel the transmitted packets not only experience the presence of noise, but also the fading even if there is no collision. The fact that the packet reaches the receiver with different power levels makes possible the capture effect. In other words, the signal with greater intensity can be captured by the receiver. Hence, the probability of mutual destruction of the colliding packets is reduced, resulting in an increase in the efficiency of the system [COV 99].

So, we define the capture probability,  $P_{capture}(i)$ ,  $i > 0$ , as the probability that one of the  $i$  packets which collided will be received successfully. This indicates that  $i$  terminals, either in idle state or backlogged, transmit in the same slot, and the base station receives one successful packet. The capture probability is greatly related to the concept of sensitivity of the receiver ( $S_{sensitivity}$ ) or in our case of the base station. According to Hernando and Perez-Fontan [HER 99], such sensitivity is defined as

the minimal voltage level in the input of the receiver required in order to obtain a value of power ratio usually between 12 and 20 dB. To be successful in the capture of a packet, the maximum signal power should be greater than the sensitivity of the receiver, besides that the threshold between the power of the signal and the power of the possible interference should be greater than a certain margin known as capture ratio ( $\gamma$ ) [ZHO 98]. According to the previous information and given the presence of  $L$  ( $L \geq 1$ ) interfering packets (each one with a power of  $\omega_{ui}$ ,  $i = 1, 2, 3, \dots, L$ ), the probability of capture is obtained by comparing  $\omega_c$ , the power of a useful packet, with the total power of possible interference according to

$$P_{capture}(L) = \Pr\left(\frac{\omega_c}{\sum_{i=1}^L \omega_{ui}} > \gamma, \quad \omega_c > S_{sensitivity}\right). \quad [13.3]$$

Considering the modeling of the throughput of S-ALOHA carried out previously, we now add the capture effect. Under this new scheme, the state transition probability of going from  $i$  (backlogged/idle terminals transmit in a slot) to  $j$  (backlogged terminals) can be determined according to four cases of state transition [COV 05]. A transition from state  $i$  to state  $j < i - 1$  is impossible because the number of backlogged terminals cannot be decreased by more than one since the receiver only decodes one packet per time slot. The transition from state  $i$  to state  $i - 1$  indicates a reduction in the backlog which is possible when  $i > 0$  and no terminal in idle state transmits and only one terminal in backlog is captured successfully. Another case is when the number of backlogged terminals move to a state  $j$ , where  $j \geq i$ , this situation could happen for two different reasons. The first results when  $j - i + 1$  terminals in idle state transmit and one terminal is captured successfully, and the second results from a situation in which  $j - 1$  terminals in idle state transmit and no packet is captured successfully. The last case is when the final number of terminals in backlog is equal to  $M$ , and this will happen when  $M - i$  idle terminals transmit and no packet is captured successfully. The expressions obtained to form the matrix of transition probabilities, according to the previous cases, are the following:

$$p_{ij} = \begin{cases} 0, & j < i - 1, \\ (i - \varphi)^{M-i} \sum_{c=1}^i \binom{i}{c} \nu^c (1 - \nu)^{i-c} P_{capture}(c), & j = i - 1, \\ 1_{\{M>j\}} \binom{M-i}{j-i+1} \varphi^{j-i+1} (1 - \varphi)^{M-j-1} \\ \cdot \sum_{k=0}^i \binom{i}{k} \nu^k (1 - \nu)^{i-k} P_{capture}(k+j-i+1) \\ + \binom{M-i}{j-i} \varphi^{j-i} (1 - \varphi)^{N-j} \sum_{k=0}^i \binom{i}{k} \nu^k \\ \cdot (1 - \nu)^{i-k} (1 - P_{capture}(k+j-i)), & M \geq j \geq i. \end{cases}$$

In the case of capture effect, we will follow the same procedure described previously, in which the vector of steady-state probability  $\pi$  can be calculated by solving a group of nonlinear equations.

This makes the probability of success,  $P_{succ}(i)$  in [13.1], be rewritten to consider the capture effect as:

$$P_{succ}(i) = \sum_{r=0}^M \binom{M-i}{M-i-r} \varphi^r (1-\varphi)^{M-i-r} \sum_{c=0}^i \binom{i}{c} \nu^c (1-\nu)^{i-c} P_{capture}(c+r). \quad [13.4]$$

To calculate the capture probabilities,  $P_{capture}$ , we apply the analysis done in [COV 05]. By considering the capture effect, the throughput ( $S$ ) is given by [13.2].

### 13.2.2. Carrier sense multiple access

We analyzed in section 13.2.1 the behavior of the throughput without/with capture effect to S-ALOHA. In this section, the analysis of CSMA protocol is presented.

#### 13.2.2.1. Modeling of CSMA

ALOHA does not listen to the channel before transmission. On the other hand, CSMA protocol is based on the concept that each terminal on the network is able to sense the channel before transmitting the data packet. Sensing the channel means being able to monitor the status of channel whether it is idle or busy. If the channel is idle/free, then the terminal can transmit the data. But if the channel is sensed busy, the station will wait and keep on sensing the carrier till it becomes free. This method decreases the probability of collision. The CSMA access modes are [KLE 75]:

- non-persistent (np): In this type of CSMA, a station senses the channel first. If the channel is free then it starts transmission immediately. But if channel is busy then the station does not continuously sense the channel, rather it waits for a random amount of time and then repeats the algorithm.

- $p$ -persistent: When the sender is ready to send data, it checks continually if the medium is busy. If the medium becomes idle, the sender transmits a frame with a probability  $p$ . If the station did not choose to transmit (the probability of this event is  $1 - p$ ), the sender waits until the next available time slot and transmits again with the same probability  $p$ . This process repeats until the frame is sent or some other sender stops transmitting. In the latter case, the sender monitors the channel, and when idle transmits with a probability  $p$ , and so on.

- 1-persistent: When the sender (station) is ready to transmit data, it checks if the physical medium is busy. If so, it senses the medium continually until it becomes idle, and then it transmits a piece of data (a frame). In case of a collision, the sender waits for a random period of time and attempts to transmit again.

We will only analyze non-persistent CSMA protocol, because the random process used by this protocol makes it easy to obtain the mathematical model, and with this the relative simplicity of simulation itself.

By considering the analysis done in [KLE 75] and [NOG 06], and according to the renewal theory the average channel utilization (throughput,  $S$ ) is given by:

$$S = \frac{E[U]}{E[I] + E[B]}, \quad [13.5]$$

where the average useful time in which no collisions occur and packets are successfully transmitted ( $E[U]$ ) is given by:

$$E[U] = e^{-aG}, \quad [13.6]$$

where  $G$  is the offered traffic and  $a$  is the ratio of propagation delay to packet transmission.

The average duration of an idle period is given by:

$$E[I] = \frac{1}{G}. \quad [13.7]$$

The expectation time length of the busy period is given by:

$$E[B] = 1 + 2a - \frac{1}{G}(1 - e^{-aG}). \quad [13.8]$$

Therefore, using [13.5] and substituting the expressions for  $E[U]$ ,  $E[I]$ , and  $E[B]$ , we get the throughput,  $S$ :

$$S = \frac{Ge^{-aG}}{G(1 + 2a) + e^{-aG}}. \quad [13.9]$$

As mentioned previously, the development of packet communication protocols has been based mainly on the assumption that when two or more packets collide (overlap in time) all packets are lost. However, a realistic radio receiver is able to be captured by the strongest of the overlapping packets, and thus, receives this packet correctly. This is called the capture effect. The probability that one packet captures the common receiver in the presence of several other (weather) contenders depends on the power differences with which the packets are received. Below is presented the modeling of the capture effect applied to non-persistent (np)-CSMA [NOG 06, HAR 02].

Considering  $t$  the arrival time of the first packet and any other packet arriving between  $t$  and  $t + a$  will find the channel as unused, will transmit, and hence, will cause a conflict. Then, a successful transmission is present when:

- 1) there are no packets arriving at vulnerable period;
- 2) there are some colliding packets arriving but the original packet is captured.

Thus, the average useful time is given as:

$$E[U] = e^{-aG} + \sum_{n=1}^{\infty} (aG)^{-n} \frac{e^{-aG}}{(n+1)!} P_{capture}, \quad [13.10]$$

$$E[U] = e^{-aG} + \frac{P_{capture}}{aG} (1 - e^{-aG} - aGe^{-aG}). \quad [13.11]$$

Therefore, considering capture effect the throughput is:

$$S = \frac{E[U]}{E[I] + E[B]} = \frac{e^{-aG} + \frac{P_{capture}}{aG} (1 - e^{-aG} - aGe^{-aG})}{1 + 2a + \frac{e^{-aG}}{G}}. \quad [13.12]$$

CSMA reduces the probability of collisions by allowing terminals to detect the carrier of the other user's transmissions. But CSMA cannot avoid collision if two terminals are within the range of each other or if they are separated by some physical obstacle. Two such terminals are considered hidden from each other. ISMA is one solution that has been offered to solve the hidden access-terminal problem and this protocol is analyzed in the following section.

### 13.2.3. Inhibit sense multiple access

In ISMA, the base station sends a busy signal to all access terminals when the base station is receiving packets from access terminals. On the other hand, the base station sends an idle signal when the base station is not receiving any packets. When each access terminal receives the idle signal, each access terminal must decide whether to transmit packets to the base station or not. When each access terminal receives the busy signal, the packet transmission of each access terminal is inhibited.

#### 13.2.3.1. Modeling of ISMA

In this section, we analyze only slotted non-persistent ISMA as undertaken in [PRA 97] and [RIV 06]. When the idle period is only one slot long, it means there is at least one arrival in the first slot of the idle period. For the period to be two slots long means that there is no arrival in the first slot, but there is at least one arrival in its second slot. Continuing the reasoning and considering the Poisson scheduling process, we have

$$E[I] = \frac{d}{1 - e^{-dG}}, \quad [13.13]$$

where  $d$  is the propagation delay and  $G$  is the offered traffic.

A collision might occur if two or more packets arrive within the same slot and are scheduled for transmission in the next slot. A busy period will contain  $t$  transmission periods if there is at least one arrival in the slot of each of the first  $t - 1$  transmission periods, and no arrival in the last slot of the  $t$ th transmission period. Thus, the busy period is:

$$E[B] = \frac{1 + d}{e^{-dG}}. \quad [13.14]$$

The expected useful time is found as:

$$E[U] = \frac{E[B]}{1 + d} P_{succ} \quad [13.15]$$

where  $P_{succ}$  is the probability of a successful transmission period. We have:

$$P_{succ} = \frac{\Pr[\text{single arrival within a slot}]}{\Pr[\text{more arrival within a slot}]} = \frac{dGe^{-dG}}{1 - e^{-dG}}. \quad [13.16]$$

So, the throughput is given as:

$$S = \frac{E[U]}{E[I] + E[B]} = \frac{dGe^{-dG}}{1 + d - e^{-dG}}. \quad [13.17]$$

By considering the capture effect for ISMA [RIV 06, ZDU 97], we have

$$E[B] = 1 + \alpha, \quad [13.18]$$

and

$$E[I] = \frac{\alpha e^{-\alpha G}}{1 - e^{-\alpha G}}, \quad [13.19]$$

where  $\alpha$  is the inhibit time which is the average time between the beginning of a packet transmission and the detection of a busy status by the remaining terminals.

Taking into account capture, there is non-zero probability that a packet is received successfully in the presence of interfering packets. Let  $q_l$  be the probability that any packet is received successfully given that there are  $l$  colliding packets. Then  $E[U]$  is the sum over all possible numbers of colliding packets during a busy period:

$$E[U] = \frac{\sum_{l=0}^{\infty} \frac{(\alpha G)^{l+1}}{(l+1)!}}{1 - e^{-\alpha G}} e^{-\alpha G} q_l. \quad [13.20]$$

Therefore, considering capture effect the throughput is:

$$S = \frac{\sum_{l=0}^{\infty} \frac{(\alpha G)^{l+1}}{(l+1)!}}{1 + \alpha - e^{-\alpha G}} e^{-\alpha G} q_l. \quad [13.21]$$

To evaluate [13.21],  $q_l$  must be determined by considering  $(l + 1)$  packets simultaneously on the forward channel. The capture criterion is that a successful packet reception occurs if there exists a received packet power  $\omega_j$ , such that

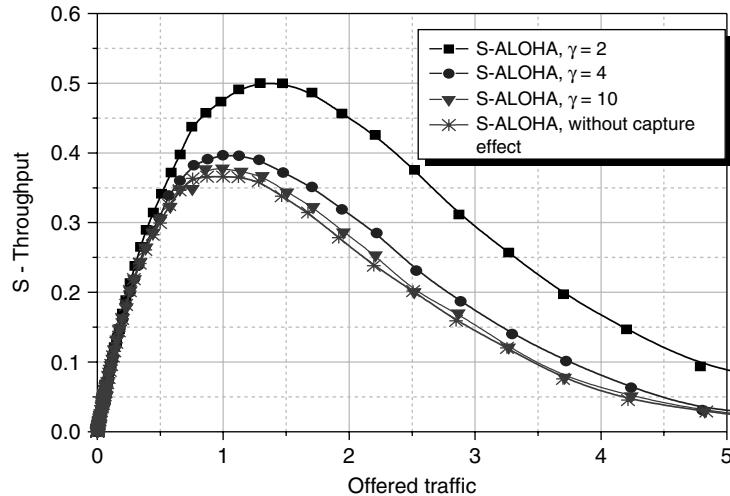
$$\omega_j > \gamma \sum_{i=1, i \neq j}^{l+1} \omega_i, \quad [13.22]$$

where  $\gamma$  is the capture ratio.

The results of the performance evaluation of the protocols S-ALOHA, CSMA, and ISMA are presented in the following section.

#### 13.2.4. Results of performance evaluation

In order to achieve the simulation process by a program in C language, we have generated packets with generation probabilities of  $10^{-3}$  to 1. For each probability of generation, 10,000 packets have been transmitted, where each packet is contained in a slot, and the probability of transmitting a new packet is 1, independently of the present value of retransmission probability. We also assume that a terminal cannot generate a new packet until the present packet has been transmitted. In addition, we have considered 100 terminals and minimum sensitivity of  $-116$  dBm.



**Figure 13.1.** Behavior of the throughput for S-ALOHA

In Figure 13.1, we show the typical response of S-ALOHA with its low efficiency (0.36), and a strong instability as traffic increases on the channel radio. In addition, in

in this figure we can observe the influence of the capture ratio ( $\gamma$ ) in the behavior of the throughput. For the case of almost perfect capture,  $\gamma = 2$ , the throughput presents the best response, reaching a value of 0.50, that is, a 38% increase with respect to the theoretical S-ALOHA. As the capture ratio increases, the value of the throughput tends to decrease. When  $\gamma = 10$ , the throughput tends toward the S-ALOHA values with an ideal channel (error-free channel). The result can be improved by adding the radio channel effect (Rayleigh fading and shadowing) because it serves to increase the capture probabilities compared to the case where no radio channel is considered, this is because with the radio channel the probability that a terminal survives a collision with other terminals contending is high.

Figure 13.2 shows the CSMA throughput with and without capture effect. We observe that in both the cases there is an instability zone. This problem is due to the propagation delay and offered traffic on the network. In addition, throughput is inversely proportional to the traffic and propagation delay, i.e. if the traffic and/or propagation delay increases, the throughput decreases.

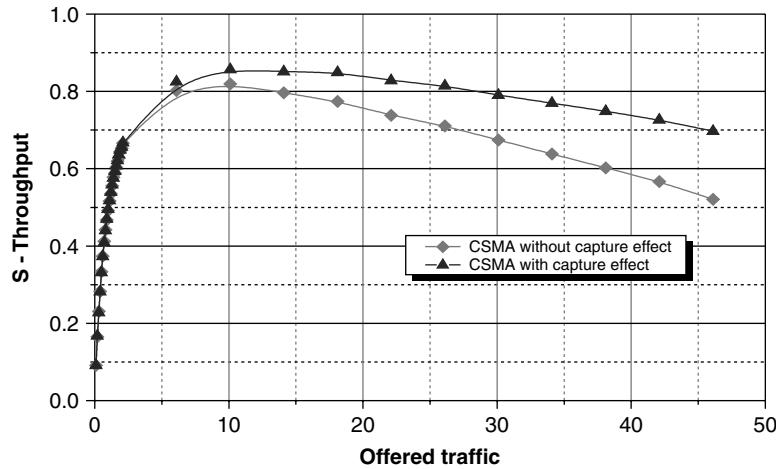


Figure 13.2. Behavior of the throughput for CSMA

Figure 13.3 represents the throughput of the ISMA protocol. In this figure, we see that the throughput increases when the capture effect is used. In addition, the terminals transmit before the end of the slot and reduce collisions, and this results in a high throughput. Furthermore, ISMA gives a higher throughput than slotted ALOHA, provided the transmission delay  $d$  is small and the offered traffic ( $G$ ) is large. If we use larger values of  $d$  then the ISMA performance decreases very rapidly.

Up until now we have been able to analyze the MAC protocols: S-ALOHA, CSMA, and ISMA. These protocols can be used in the access

request stage. So, we use CSMA protocol for the access request because it is an alternative in the IEEE 802.11 protocol and the hidden terminal problem is solved using the RTS (request-to-send)/CTS (clear-to-send) mechanism [JAS 07]. Moreover, in transmission stage we apply the CDMA protocol. This protocol (CSMA-CDMA) is explained in following section.

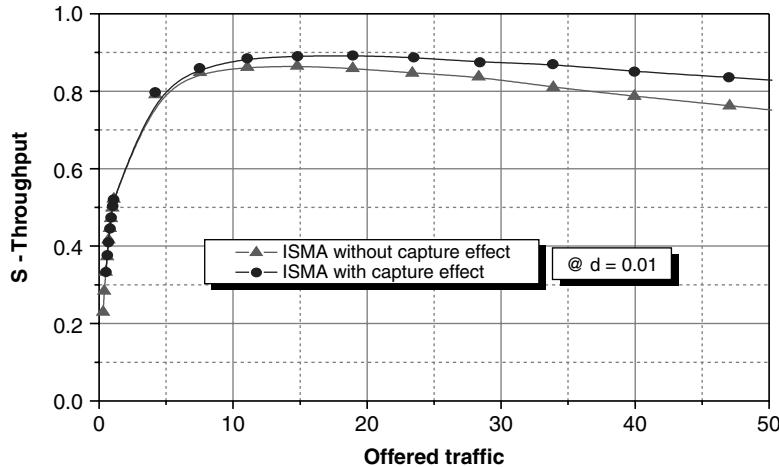


Figure 13.3. Behavior of the throughput for ISMA

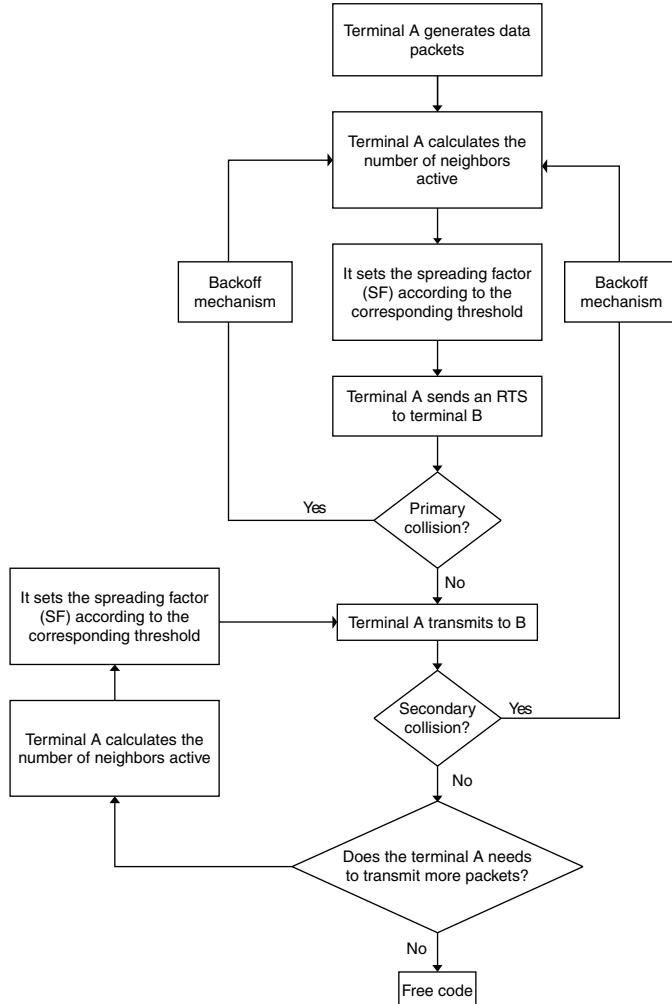
### 13.3. Proposed MAC protocol

In this section, we use a MAC protocol using CSMA/CA with CDMA for wireless networks. The basic access mechanism for CSMA/CA fails to address the well-known hidden-terminal and exposed-terminal problems [JAY 04]. The solution for these problems is to use handshaking process. In this process, whenever a packet is to be transmitted, the transmitting terminal first sorts out the RTS packet. If receiving terminal hears the RTS, it responds with a short CTS packet. After this exchange, the transmitting terminal sends its packet. When the packet is received successfully, the receiving terminal transmits an acknowledgement (ACK) packet. The RTS/CTS packets include the information of how long it takes to transmit the next data packet. Thus, other terminals close to the transmitting terminal and the hidden terminal close to the receiving terminal will not start transmission during this period. In the protocol described previously, CSMA/CA, the terminals transmit with the same rate, and it presents the problem of congestion in the network. Therefore, we use an alternative to be able to transmit with different transmission rates for obtaining adaptability in the traffic of the network. These improvements in the performance (throughput) in the wireless network are obtained using CDMA.

CDMA is a multiple access protocol based on a spread-spectrum communication [IPA 05]. CDMA does achieve its access multiple by assigning each terminal a different code. This code is used to transform a terminal's signal in wideband signal (spread-spectrum signal). If a receiver receives multiple wideband signals, it will use the code assigned to a particular terminal to transform the wideband signal received from that terminal back to the original signal. During this process, the desired signal power is compressed into the original signal bandwidth, while the wideband signals of the other terminals remain wideband signals and appear as noise when compared to the desired signal. As long as the number of interfering terminals is not too large, the signal-to-noise ratio will be large enough to extract the desired signal without error. If the number of terminals rises above a certain limit, the interference becomes too large for the desired signal to be extracted and contention occurs, thus making the protocol interference limited. The operation of the alternative to transmit with multirate for CSMA–CDMA protocol is presented below.

A flowchart of the operation of CSMA–CDMA protocol with multirate is illustrated in Figure 13.4. When the terminal needs to transmit information packets, it estimates the transmission rate to be used and the number of active neighboring terminals. For this, the terminal decides the use of transmission rate takes into account threshold values. For instance, for the threshold  $x_1$  all active terminals transmit with the highest rate ( $r_{max}$ ). If the number of active terminals ( $n$ ) is  $x_1 < n < x_2$  then all terminals transmit with a rate  $r_{max}/2$ . Moreover, when the number of active terminals is  $x_2 < n < x_3$ , then active terminals transmit with a rate of  $r_{max}/3$ . All active terminals transmit with lowest rate ( $r_{min}$ ) when  $n > x_m$ . In this algorithm when the terminal  $A$  generates data packets to transmit, it estimates the number of active neighboring terminals and takes into account the threshold value. In addition, terminal  $A$  sets its spreading factor (processing gain). Subsequently, the terminal  $A$  sends an RTS message to terminal  $B$  by using a common code. If a primary collision occurs then the terminal uses a backoff mechanism to retransmit and again it estimates the number of active neighboring terminals to adjust its spreading factor. When terminal  $A$  receives the CTS message successfully then it transmits the data information to terminal  $B$ . Terminal  $A$  applies the backoff mechanism in case of a collision due to multiuser interference (secondary collision), and it estimates the number of active neighboring terminals. If after transmitting, terminal  $A$  has no more information to transmit, it frees its code and switches to an idle state. In case terminal  $A$  has additional information to transmit, it estimates again the number of active neighbors terminals and adjusts its spreading factor according to the threshold value.

In CDMA the bandwidth is constant,  $B = SF \cdot r$ , where  $SF$  is the spreading factor and  $r$  is the transmission rate. Therefore, if we increase the transmission rate by a factor of  $\zeta$  then the spreading factor decreases in the same proportion and reduces the protection against multiuser interference.



**Figure 13.4.** Flowchart of CSMA–CDMA protocol with multirate

For threshold values, we use the probability of receiving correctly a packet,  $P_c(n)$ ,

$$P_c(n) = (1 - P_b(n))^{\zeta Z}, \quad [13.23]$$

the bit error rate,  $P_b(n)$ , can be described by:

$$P_b(n) = Q(\sqrt{2 \cdot SNR}), \quad [13.24]$$

where  $Q(y) = 1/\sqrt{2\pi} \int_y^\infty e^{-u^2/2} du$ , and  $SNR = 1/(2(n-1)/3 \cdot SF) + (n_o/E_b)$ , the energy per bit is  $E_b$ ,  $SF$  is the spreading factor or processing gain, packet length

of  $Z$  bits,  $n$  denotes number of active terminals, and  $n_o$  is the density of the Additive White Gaussian Noise (AWGN) power spectrum.

To apply the proposed MAC protocol to UAVs, we need to take the communication between UAVs into account through wireless links using directional antennas.

As previously mentioned, the proposed MAC protocol uses the RTS/CTS mechanism. Moreover, we should mention that each UAV is equipped with a global positioning system (GPS) and is used to provide its position (latitude, longitude, and altitude), and this information is used in the RTS/CTS mechanism. Therefore, the proposed MAC protocol used in UAVs is as follows:

- The UAV A sends an RTS message to UAV B by using a common code and in this message transmits the position of the UAV using the omnidirectional antenna. In addition, this information can be received for the neighboring UAVs and update their information tables (position).
- The UAV B responds with a CTS and this message carries the information of its position for UAV A and the neighboring UAVs.
- Furthermore, the UAV A calculates the distance of each neighbor and the UAV B. For the information transmission (data packets) the directional antenna is used. The UAV A determines the transmission rate according to the threshold value. If there is an error in data transmission (interference), the UAV enters a state of retransmission.
- If the UAV B receives the data successfully then it sends an ACK using the directional antenna sending the information of the table (position).

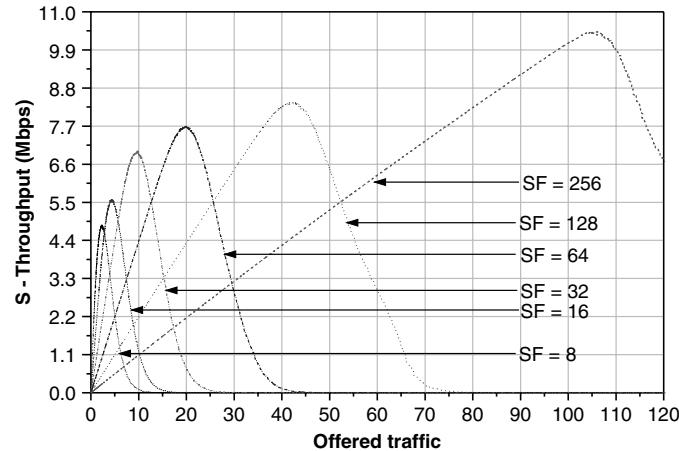
The advantages of the MAC protocol applied to UAVs are: Calculate the signal-to-interference ratio and manage dynamically the transmission rate; in addition, the necessary power level for transmission. Moreover, UAVs can be located by GPS and calculating the distance between its neighbors. Furthermore, communication channels are specified by CDMA codes.

The results of the proposed protocol (CSMA–CDMA) are presented in the following section.

### 13.4. Experimental setup and results

The performance of the proposed MAC protocol will now be evaluated. The process of simulation was made through the design of a program in C language, on a personal computer, which considers the properties of the CDMA protocol. In the simulation, 80 simulation cycles were conducted. During each cycle, 10,000 frames with frame size of 10 ms are generated. In addition, system performance is only limited by interference, therefore thermal noise is negligible.

We present in Figure 13.5 the throughput behavior for CSMA–CDMA with fixed rates using six spreading factors (8, 16, 32, 64, 128, and 256). We might think that transmitting at higher rates would result in increased throughput, but, as was mentioned previously, by decreasing the spreading factor the result is a reduction in protection against multiuser interference, and therefore, a lower throughput in the system (see Figure 13.5). Furthermore, the number of active terminals diminishes when spreading factor is reduced. After analyzing the behavior of CSMA–CDMA with fixed rates, now analyze the proposed protocol CSMA–CDMA with multirate considering transmission thresholds. In this case, it is determined how many terminals are active and according to the transmission threshold all terminals adjust their transmission rate and the result is shown in Figure 13.6.



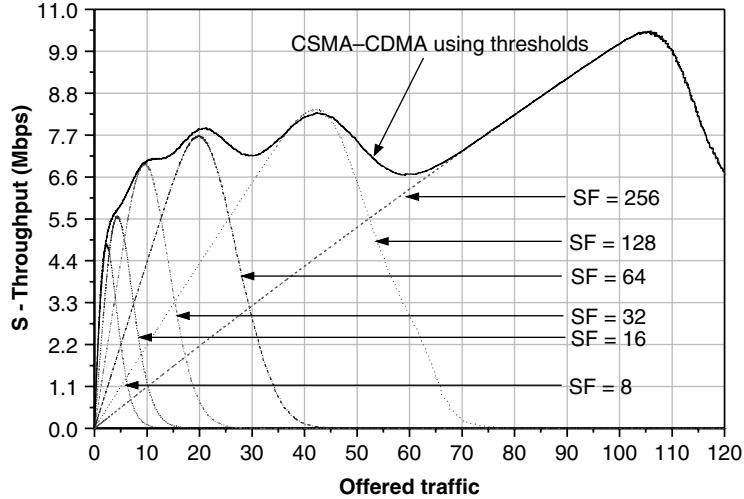
**Figure 13.5.** CSMA–CDMA protocol with fixed rates

The behavior observed in Figure 13.6 is due to the use of transmission threshold. At first the terminals start transmitting with the highest transmission rate, because there are few terminals (low traffic) that want to transmit and when the number of active terminals increase, then all active terminals decrease their transmission rate, because it exceeded the threshold for the higher transmission rate. If the number of active terminals continues to increase then the transmission rate will reduce again, so this procedure is adapted to traffic conditions in the communication channel. Therefore, the proposed MAC protocol improves network performance at fixed rates.

### 13.5. Conclusions

In this chapter, we have modeled and simulated the MAC protocols: S-ALOHA, CSMA, and ISMA. To increase the efficiency of the system, we have used the capture

effect. Therefore, S-ALOHA suffers heavily from collisions among packets. CSMA offers higher capacity but their performance is degraded by hidden terminals problem. On the other hand, ISMA reduces the two problems of collisions and hidden terminals.



**Figure 13.6.** CSMA-CDMA protocol using thresholds

By using CSMA at the channel request stage and CDMA to transmit, we developed a MAC protocol that is adaptive to traffic conditions in the communication channel. Therefore, we have proposed a MAC protocol based on CSMA-CDMA for wireless networks and, through a simulation process, have shown that our proposed MAC protocol using thresholds improves the performance compared with fixed rates.

### 13.6. Acknowledgments

This work was supported by the Autonomous University of Tamaulipas Mexico, under grant UAT10-ING-0217.

### 13.7. Bibliography

- [BOL 06] BOLCH G., GREINER S., DE MEER H., TRIVEDI K., *Queueing Networks and Markov Chains: Modeling and Performance Evaluation with Computer Science Applications*, 2<sup>nd</sup> edition, Wiley-Interscience, New York, 2006.
- [CAR 75] CARLEIAL A., HELLMAN M., “Bistable behavior of ALOHA-type systems”, *IEEE Transactions on Communications*, vol. 23, no. 4, pp. 401–410, 1975.

- [COV 99] COVARRUBIAS D., Procedures and techniques of dynamic assignment and stabilizing of applicable MAC to mobile systems of third generation, PhD Thesis, UPC, Spain, 1999 (in Spanish).
- [COV 05] COVARRUBIAS D., MENDEZ A., VARGAS C., “Performance analysis of S-Aloha as a random access channel on mobile communications”, *Journal of Applied Research and Technology*, vol. 3, no. 1, pp. 3–12, 2005.
- [HAR 02] HARADA H., PRASAD R., *Simulation and Software Radio for Mobile Communications*, Artech House, Boston, MA, 2002.
- [HER 99] HERNANDO J., PEREZ-FONTAN F., *Introduction to Mobile Communications Engineering*, Artech House, Boston, 1999.
- [IPA 05] IPATOV V., *Spread Spectrum and CDMA: Principles and Applications*, 1<sup>st</sup> edition, John Wiley and Sons Inc, West Sussex, Boston, MA, 2005.
- [JAS 07] JASANI H., ALARAJE N., “Evaluating the performance of IEEE 802.11 network using RTS/CTS mechanism”, *Proceedings of the IEEE International Conference on Electro/Information Technology*, Chicago, IL, pp. 616–621, May 2007.
- [JAY 04] JAYASURIYA A., PERREAU S., DADEJ A., GORDON S., “Hidden vs exposed terminal problem in ad hoc networks”, *Proceedings of the Australian Telecommunication Networks and Applications Conference*, Sydney, Australia, pp. 52–59, December 2004.
- [KLE 75] KLEINROCK L., TOBAGI F.A., “Packet switching in radio channels: Part I-carrier sense multiple-access modes and their throughput-delay characteristics”, *IEEE Transactions on Communications*, vol. 23, no. 12, pp. 1400–1416, 1975.
- [MEN 09] MENDEZ A., COVARRUBIAS D., VARGAS C., *Performance Analysis for Medium Access Control Protocols*, 1<sup>st</sup> edition, Innovacion Editorial Lagares SA de CV, Naucalpan, Estado de Mexico, 2009 (in Spanish).
- [MEN 10] MENDEZ A., AQUINO R., COVARRUBIAS D., PANDURO M., “Performance evaluation of the RQMA-CDMA scheme for multimedia traffic with QoS guarantees”, *AEU International Journal of Electronics and Communications*, vol. 64, no. 10, pp. 916–923, 2010.
- [MOL 11] MOLISCH A., *Wireless Communications*, 2<sup>nd</sup> edition, John Wiley and Sons Inc., West Sussex, 2011.
- [NOG 06] BRISEÑO NOGUEDAD I., Performance analysis of CSMA protocol applied to mobile communications, Master Thesis, Electronics Department, UAT, Reynosa, Tamaulipas, Mexico, 2006 (in Spanish).
- [PRA 97] PRASAD N., NIJHOF J., “Indoor wireless communications using slotted non-persistent ISMA, 1-persistent ISMA and non-persistent ISMA/CD”, *Proceedings of the IEEE 47<sup>th</sup> Vehicular Technology Conference*, Phoenix, AZ, pp. 1513–1517, May 1997.
- [PRA 98] PRASAD R., *Universal Wireless Personal Communications*, 1<sup>st</sup> edition, Artech House, Boston, MA, 1998.
- [RIV 06] RIVERA-CHAVEZ O., Performance evaluation of ISMA protocol for mobile communications, Master Thesis, Electronics Department, UAT, Reynosa, Tamaulipas, Mexico, 2006 (in Spanish).

- [ROM 90] ROM R., SIDI M., *Multiple Access Protocol-Performance and Analysis*, Springer-Verlag, New York, 1990.
- [ZDU 97] ZDUNEK K., UCCI D., LOCICERO J., “Packet radio performance of inhibit sense multiple access with capture”, *IEEE Transactions on Communications*, vol. 45, no. 2, pp. 164–167, 1997.
- [ZHE 09] ZHENG J., JAMALIPOUR A., *Wireless Sensor Networks: A Networking Perspective*, John Wiley and Sons Inc., Hoboken, NJ, 2009.
- [ZHO 98] ZHOU H., DENG R., “Capture model for mobile radio slotted ALOHA systems”, *IEE Proceedings Communications*, vol. 145, no. 2, pp. 91–97, 1998.



## Chapter 14

# Optimization of a Scannable Pattern for Bidimensional Antenna Arrays to Provide Maximum Performance

### 14.1. Introduction

In recent years, the development of wireless communication systems has grown extensively around the world. In this case, in many antenna applications, such as cellular systems, radar systems, unmanned aerial vehicles (UAVs), etc., higher coverage in a wide scanning range is generally required [STU 98, LIA 08]. For such requirements, the design of antenna systems includes a control gain as well as a control of the interferences. To this end, considering the antenna arrays as part of the antenna system is very suitable for such applications. An antenna array is defined as a group of antennas arranged in a singular geometry that provides certain characteristics of directive radiation [HAN 38]. To achieve this, the current distribution defined as a continuous function of an antenna is modified to a current distribution defined as a discrete function of an antenna array. The main advantage of the antenna arrays is adjustable radiation parameters, such as side lobe level and beamwidth, scanning beam, multibeam, and high directivity in a relative compact size. This technology increases the efficiency of the communication systems providing higher capacity of coverage, higher reliability, low power consumption, and reduction of interferences. The radiation pattern of an antenna array is defined as:

$$F(\theta, \phi) = g(\theta, \phi)af(\theta, \phi)$$

---

Chapter written by A. REYNA, M.A. PANDURO and A. MENDEZ.

where  $g(\theta, \phi)$  is the radiation of a singular antenna element, such as a microstrip antenna, patch antenna, dipole, etc., called the element pattern. The term  $af(\theta, \phi)$  is the radiation of the antenna array considering the antenna elements as isotropic sources that radiate the same electromagnetic energy in all directions of the space. To obtain the directive radiation of the antenna array, the electromagnetic fields of the antenna elements interfere constructively in the desired direction and destructively in the remaining directions of the space. The characteristics of the radiation pattern depends in a certain way on the geometry of the array, spacing between the antenna elements, amplitude and phase excitations of the antenna elements, and type of antenna. This technology has been adopted for many particular applications because of its great advantages to satisfy modern communication systems. In particular, this chapter presents the capability of the antenna arrays of beam-steering, it presents two different geometries, such as rectangular planar and concentric ring antenna arrays. For such geometries of arrays, the performance of the antenna array considering the optimization of the amplitude and phase excitations across the antenna elements by using the well-known genetic algorithms (GA) [RAH 99] is presented. Owing to the great variety of parameters involved, optimization techniques, such as GA, are the appropriate tools for searching for the best antenna array models. The main objective of this chapter is to show the behavior of radiation for the design of antenna arrays in a uniform rectangular and concentric ring geometry considering the optimization of a scannable pattern in a wide scanning range. These designs consider the synthesis of the array factor with desired characteristics of the side lobe level and the directivity in a wide steering range. This chapter is organized as follows: section 14.2 describes the scannable rectangular planar antenna array model. Section 14.3 presents the scannable concentric ring antenna array model. Discussions and open problems are mentioned in section 14.4. Finally, the conclusions of this work are presented in section 14.5.

## 14.2. Design of planar antenna arrays

### 14.2.1. Theoretical model

Consider a rectangular planar antenna array of  $MN$  antenna elements uniformly spaced in the  $x-y$  plane, as shown in Figure 14.1. The array factor for the planar array shown in Figure 14.1 is given by [BAL 97]:

$$af(\theta, \phi, W, P) = \sum_{n=1}^N \sum_{m=1}^M w_{mn} e^{j[u k x_{mn} + v k y_{mn} + \sigma_{mn}]}$$

where

$$u = \sin \theta \cos \phi - \sin \theta_0 \cos \phi_0$$

$$v = \sin \theta \sin \phi - \sin \theta_0 \sin \phi_0.$$

$(x_{mn}, y_{mn})$  indicates the position of the  $mn$ th element of the array,  $(\theta_0, \phi_0)$  is the direction of maximum radiation,  $k = 2\pi/\lambda$  is the number of wave,  $\theta$  is the angle of incidence of a plane wave in the elevation plane,  $\phi$  is the angle of incidence of a plane wave in the azimuth plane,  $\lambda$  is the signal wavelength,  $w_{mn}$  represents the amplitude excitation of the  $mn$ th element of the array in the set  $W$ , and  $\sigma_{mn}$  represents the phase excitation of the  $mn$ th element of the array in the set  $P$ . The sets  $W$  and  $P$  are arranged in the following way:

$$W = \begin{bmatrix} w_1 & w_2 & \cdots & w_M \\ w_{M+1} & w_{M+2} & \cdots & w_M \\ \vdots & \vdots & \cdots & \vdots \\ \vdots & \vdots & \cdots & \vdots \\ w_{M(N-1)+1} & w_{M(N-1)+2} & \cdots & w_{MN} \end{bmatrix}$$

$$P = \begin{bmatrix} \sigma_1 & \sigma_2 & \cdots & \sigma_M \\ \sigma_{M+1} & \sigma_{M+2} & \cdots & \sigma_M \\ \vdots & \vdots & \cdots & \vdots \\ \vdots & \vdots & \cdots & \vdots \\ \sigma_{M(N-1)+1} & \sigma_{M(N-1)+2} & \cdots & \sigma_{MN} \end{bmatrix}$$

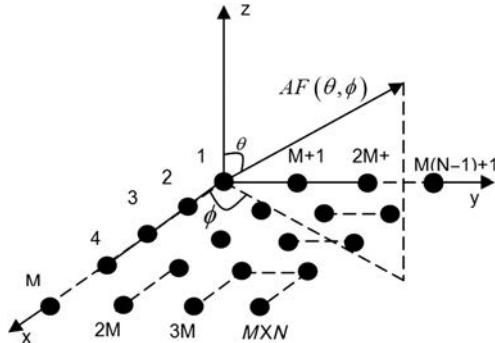


Figure 14.1. Rectangular planar antenna array

These last two equations are proposed as the optimization variables in the design procedure of the antenna array. In the following section, the fitness function is defined for designing this antenna array model.

#### 14.2.2. Objective function used to optimize planar arrays

The optimization problem shown in this chapter considers a scannable radiation pattern in a wide scanning range by optimizing the amplitude  $W$  and phase  $P$  excitations for two geometries of planar antenna arrays: rectangular and concentric

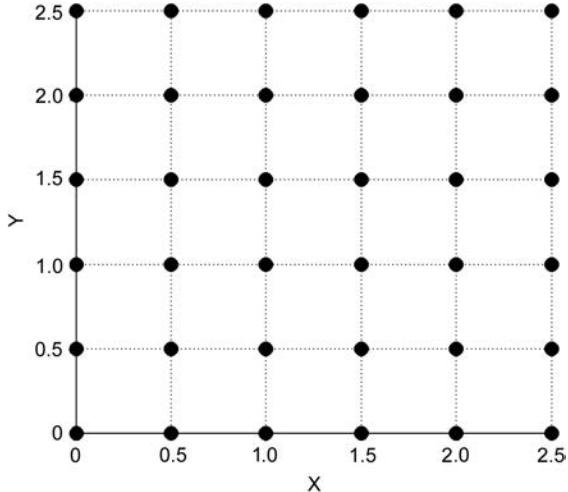
ring. To deal with this design problem, the method of GAs is used. In this case, this algorithm is chosen for its ease of implementation. Besides its simplicity for implementation, this algorithm has shown to be one of the most effective algorithms over an important set of difficult optimization problems. Note that with this selection we do not claim that GA is the best option for this particular problem: determining the best optimization algorithm for a particular antenna design problem remains as an open problem. The procedure for GA is described as follows. An individual is generated by the GA (amplitude excitations and phase perturbations of antenna elements). Each individual is, in general, represented by a vector of real numbers, which represents the amplitudes, and a vector of real numbers restrained on the range  $(0, 2\pi)$ , which represents the phase perturbation of antenna elements. The value of the objective function is calculated as [PAN 09a]:

$$\begin{aligned} of = & (|af(\theta_{SLL}, \phi_{SLL}, W, P)| / \max|af(\theta, \phi, W, P)|) \\ & + (1/|DIR(\theta, \phi, W, P)|) + |\phi_0 - \phi_0^{obt}| \end{aligned}$$

where  $(\theta_{SLL}, \phi_{SLL})$  is the angle where the maximum side lobe is attained,  $DIR$  is the directivity of the array factor, and  $\phi_0^{obt}$  is the direction of the main lobe generated for each possible solution in the optimization process. In this case, both the objectives ( $SLL$  and  $DIR$ ) are uniformly weighted in the cost function. In this case, the design problem is formulated as – minimize the objective function  $of$ . A random population of individuals is generated and the genetic mechanisms of crossover, survival, and mutation are used to obtain better and better individuals, until the GA converges to the best solution or the desired goals are achieved. The results of using the evolutionary algorithm for the design of scannable uniform planar arrays are described in the following section.

#### 14.2.3. Results obtained for the design of planar arrays

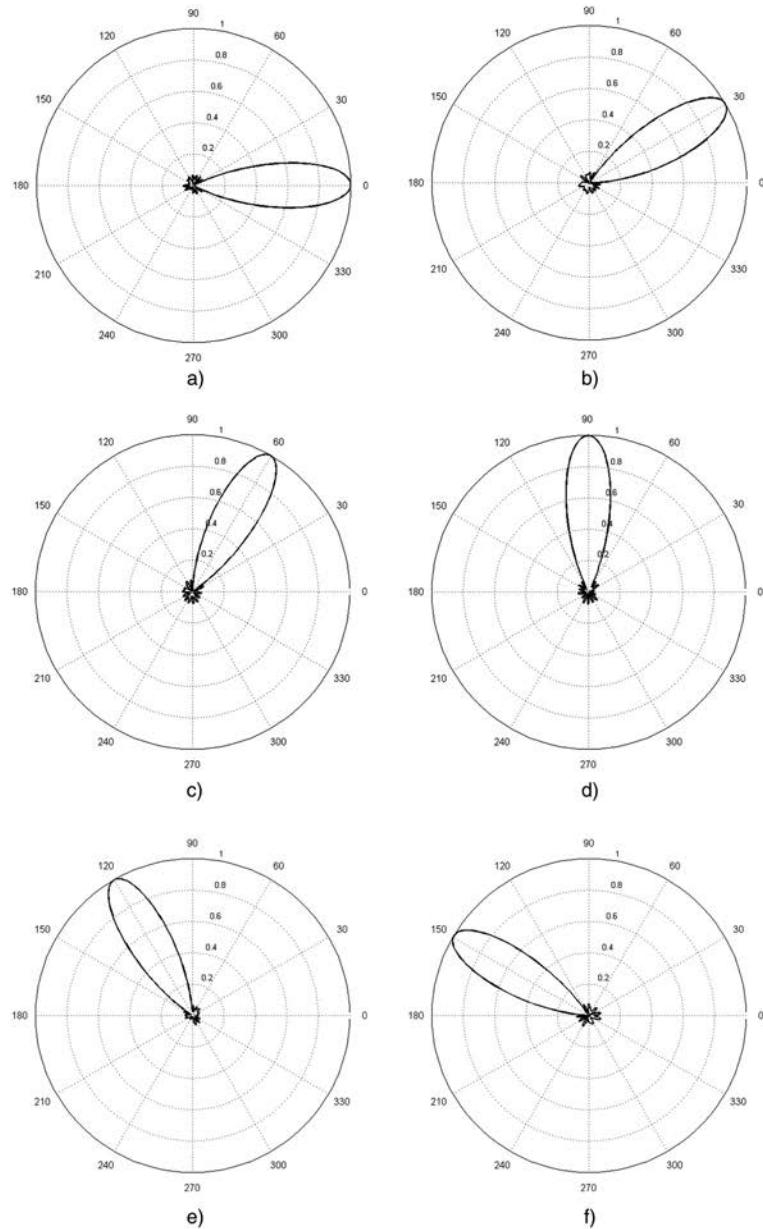
The technique of GA is implemented to study the behavior of the array factor for scannable planar arrays. As an example, the array factor generated in the azimuth plane in the cut of  $\theta = 45^\circ$  is considered, i.e. the array factor for the steering range of  $0^\circ \leq \phi_0 \leq 360^\circ$  with an angular step of  $30^\circ$ . After a trial-and-error procedure, the parameters of the simulations algorithm were set as follows: maximum number of generation  $i_{max} = 500$ , population size  $gsize = 200$ , crossover probability  $pc = 0.95$ , and mutation probability  $pm = 0.1$ . As an example, Figure 14.2 shows a rectangular planar antenna array with the next configuration: the separation between antenna elements is set as  $d = 0.5\lambda$  and the number of the antenna elements is  $N = M = 6$ . As shown in the example of Figure 14.2, the evolutionary algorithm generates a set of amplitude and phase excitations in each angle of the scanning range to provide a normalized array factor with a side lobe level  $\leq 20$  dB in the steering range. In this case, Figures 14.3 and 14.4 illustrate the behavior of the array factor for a steerable planar array with the amplitude and phase excitation optimized by the GA.



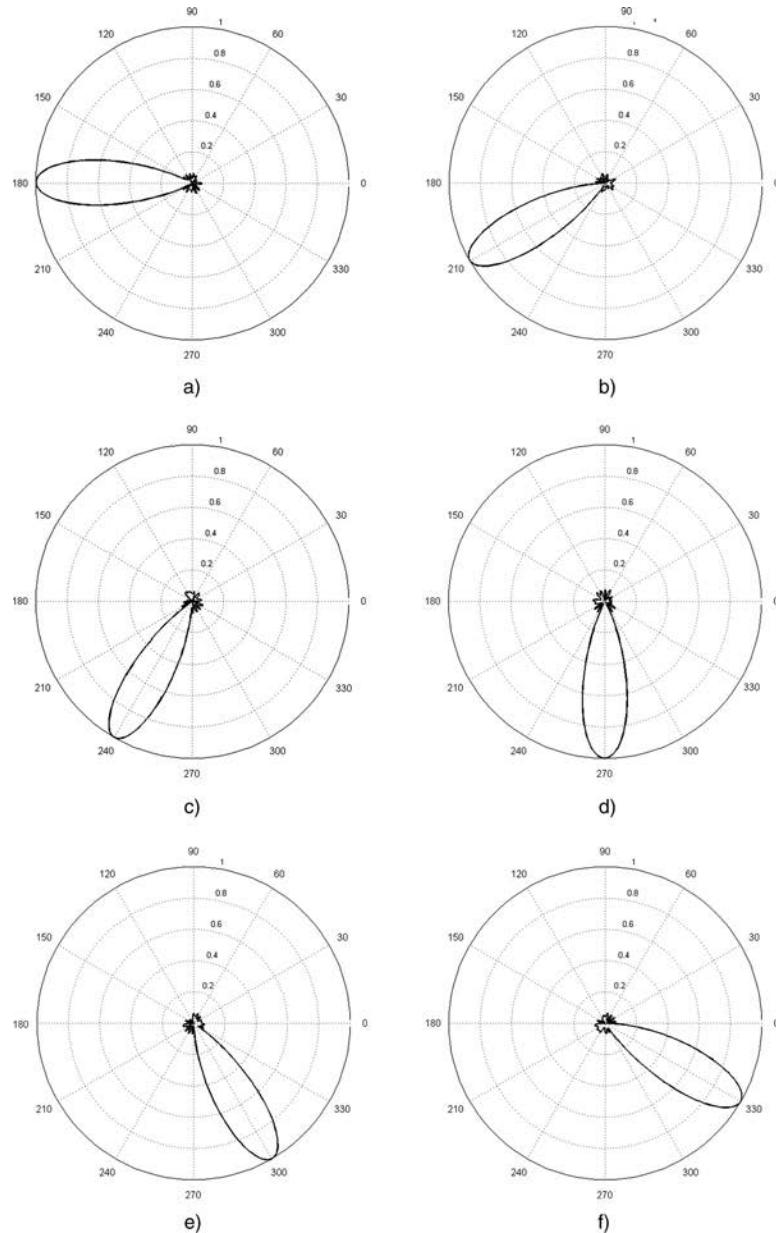
**Figure 14.2.** Rectangular planar antenna array  $6 \times 6$  with uniform spacing of  $d = 0.5\lambda$

In Figure 14.5, a particular case for  $\phi_0 = 150^\circ$  in the cut of  $\theta = 45^\circ$  is shown. In this figure, a comparison of the array factor between this particular case, optimized with the evolutionary algorithm and the conventional case, i.e. the conventional progressive phase excitation, is shown. From this figure, a considerable reduction of the side lobe level for the case optimized with the evolutionary algorithm with respect to the conventional case [REY 08] can be observed.

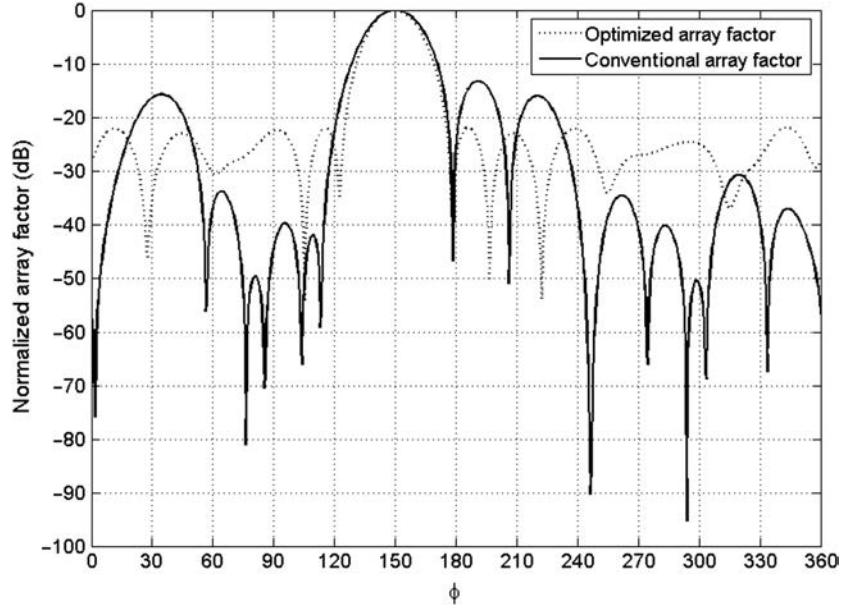
The numerical values of the side lobe level and the directivity for the array factor illustrated in Figures 14.3 and 14.4 are presented in Table 14.1. Table 14.1 illustrates that the design case with the amplitude and phase optimized by the GA could provide a better performance in side lobe level with respect to the conventional case. These low values of side lobe level for the optimized design case could be achieved with better values of directivity and the same aperture in both the design cases. The values of amplitude excitations and phase perturbations for the array factor illustrated in Figures 14.3 and 14.4 are shown graphically in Figures 14.6 and 14.7. To be more illustrative, in these figures the antenna element index of the planar array, as shown in Figure 14.1, is considered, i.e. for the first linear array the element index is from 1 to  $M = 6$ , for the second linear array the element index is from  $M + 1 = 11$  to  $2M = 12$ , and so on, until we have  $MN = 36$ . The obtained sets  $W$  and  $P$  for each main lobe direction permit a scannable main lobe in the whole azimuth scanning range. In this case, the amplitude weights could be realized in a real hardware setup rescaling all the complex coefficients before applying them to a real system. In a real system, the amplitudes will be adjusted in such a way that the conservation law of the energy will be preserved.



**Figure 14.3.** Behavior of the array factor for a scannable rectangular planar antenna array in the azimuth scanning range (a)  $\phi_0 = 0^\circ$ , (b)  $\phi_0 = 30^\circ$ , (c)  $\phi_0 = 60^\circ$ , (d)  $\phi_0 = 90^\circ$ , (e)  $\phi_0 = 120^\circ$ , and (f)  $\phi_0 = 150^\circ$  with the amplitude and phase excitations optimized by the GA for  $N = M = 6$



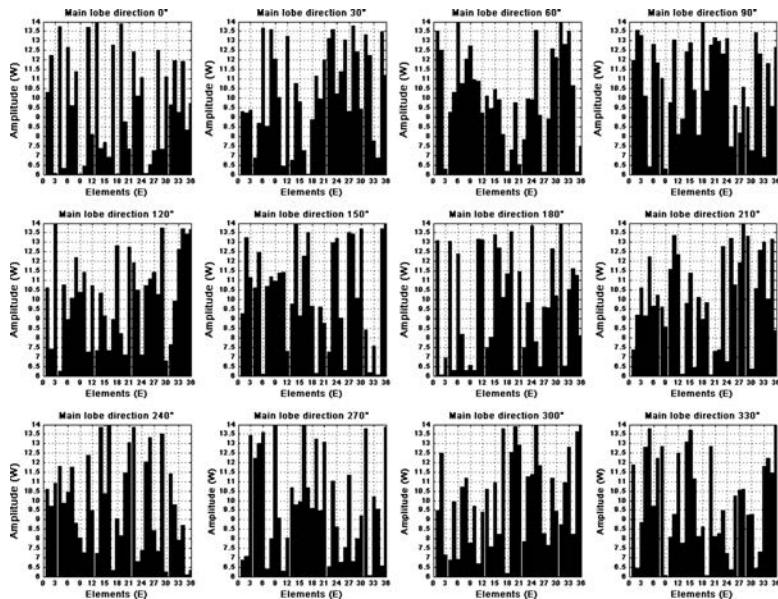
**Figure 14.4.** Behavior of the array factor for a scannable rectangular planar antenna array in the azimuth scanning range (a)  $\phi_0 = 180^\circ$ , (b)  $\phi_0 = 210^\circ$ , (c)  $\phi_0 = 240^\circ$ , (d)  $\phi_0 = 270^\circ$ , (e)  $\phi_0 = 300^\circ$ , and (f)  $\phi_0 = 330^\circ$  with the amplitude and phase excitation optimized by the GA for  $N = M = 6$



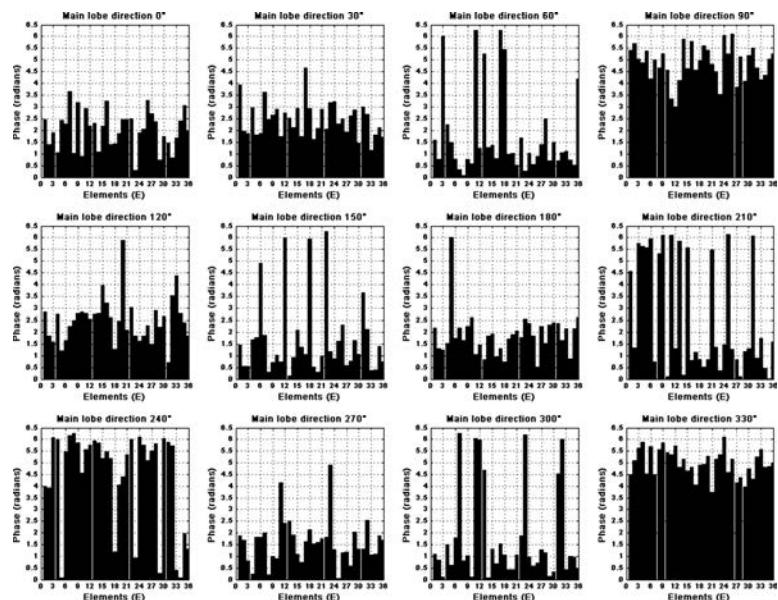
**Figure 14.5.** Comparison of the array factor between the conventional case and the optimized array factor with GA for  $\phi_0 = 150^\circ$  in the cut of  $\theta = 45^\circ$  for the scannable rectangular planar antenna array

$\phi_0$	Design case optimized with GA		Conventional case	
	SLL (dB)	DIR (dB)	SLL (dB)	DIR (dB)
$0^\circ$	-23.1732	14.6027	-16.3149	14.3479
$30^\circ$	-22.7823	14.5787	-13.2963	14.2590
$60^\circ$	-22.1089	14.7714	-13.2964	14.2591
$90^\circ$	-22.1238	14.7141	-16.3155	14.4471
$120^\circ$	-23.8875	14.8871	-13.2977	14.2586
$150^\circ$	-21.9566	14.8052	-13.2980	14.2581
$180^\circ$	-23.0816	14.6700	-16.3183	14.4437
$210^\circ$	-23.6124	14.8196	-13.2980	14.2581
$240^\circ$	-22.3970	14.9283	-13.2977	14.2586
$270^\circ$	-21.6519	14.6731	-16.3155	14.4471
$300^\circ$	-22.8402	14.7573	-13.2964	14.2591
$330^\circ$	-23.1753	14.6430	-13.2963	14.2590

**Table 14.1.** Numerical values of the side lobe level and the directivity for the array factor illustrated in Figure 14.4



**Figure 14.6.** Amplitude distribution for the array factor illustrated in Figure 14.4



**Figure 14.7.** Phase distribution for the array factor illustrated in Figure 14.4

### 14.3. Design of concentric ring arrays

#### 14.3.1. Theoretical model

Let us consider a concentric rings array of  $N_T$  elements uniformly spaced on the  $x-y$  plane, as shown in Figure 14.8. The array factor for this concentric ring array, shown in Figure 14.8, is given by [HAU 08]:

$$af(\theta, \phi, W, P) = \sum_{n=1}^{N_r} \sum_{m=1}^{N_n} w_{nm} e^{j[kr_n(u \cos \phi_m + v \sin \phi_m) + \sigma_{nm}]}$$

where

$$u = \sin \theta \cos \phi - \sin \theta_0 \cos \phi_0$$

$$v = \sin \theta \sin \phi - \sin \theta_0 \sin \phi_0$$

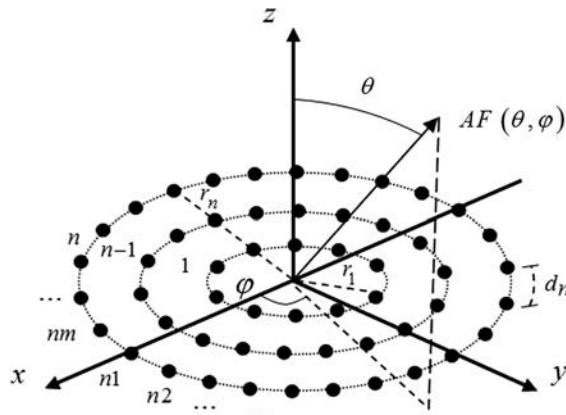


Figure 14.8. Concentric ring antenna array

$N_r$  represents the number of rings,  $N_n$  represents the number of elements on the ring  $n$ ,  $r_n$  is the radius of ring  $n$ ,  $\lambda$  is the signal wavelength,  $k = 2\pi/\lambda$  is the phase constant,  $\phi_m = 2\pi(m-1)/N_n$  represents the angular position of the element  $m$  on the ring  $n$ ,  $(\theta_0, \phi_0)$  is the direction of the maximum radiation,  $\theta$  is the angle of a plane wave in the elevation plane,  $\phi$  is the angle of a plane wave in the azimuth plane, and  $w_{nm}$  and  $\sigma_{nm}$  are the amplitude excitations and phase perturbations of the element  $m$  on the ring  $n$  defined by the sets  $W$  and  $P$ . In these sets, the amplitude excitations and phase perturbations for the entire rings are arranged in two vectors of real numbers as follows:

$$W = [w_{11}, w_{12}, \dots, w_{1N_1}, w_{21}, w_{22}, \dots, w_{2N_2}, \dots, w_{Nr1}, w_{Nr2}, \dots, w_{NrN_n}]$$

$$P = [\sigma_{11}, \sigma_{12}, \dots, \sigma_{1N_1}, \sigma_{21}, \sigma_{22}, \dots, \sigma_{2N_2}, \dots, \sigma_{Nr1}, \sigma_{Nr2}, \dots, \sigma_{NrN_n}]$$

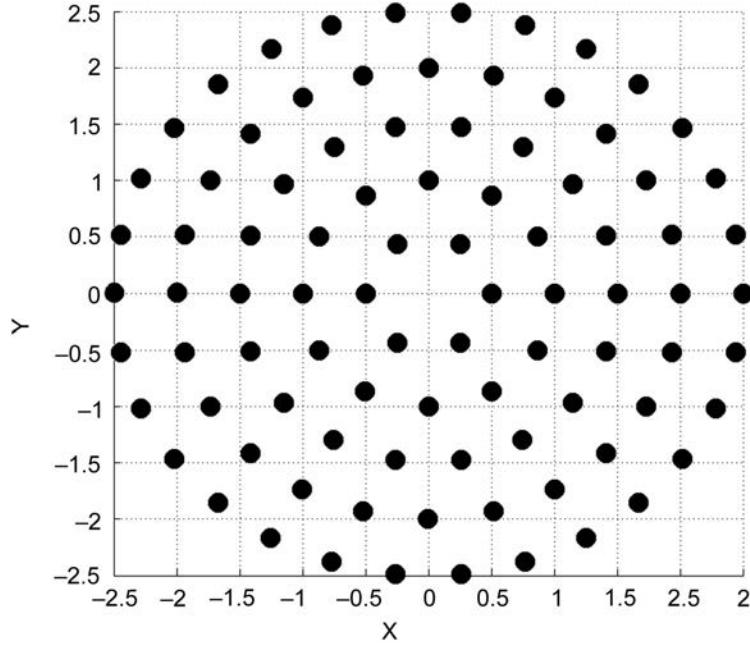
The sets  $W$  and  $P$  are divided in subsets for each ring on the plane. Thus, there exist  $N_r$  subsets for amplitude excitations and  $N_r$  subsets for phase excitations. This model considers the center of each ring as the phase reference in the array factor. Furthermore, it is considered a symmetrical excitation for the rings. Since  $N_n$  must be an even number, so that the symmetry could be correctly applied, the subset of amplitude excitations for the ring  $n$  is given as  $w_{n1}, w_{n2}, \dots, w_{nN_n/2}, w_{n(N_n/2)+1} = w_{n1}, w_{nN_n} = w_{nN_n/2}$ . And the subset of phase excitations for the ring  $n$  is given as  $\sigma_{n1}, \sigma_{n2}, \dots, \sigma_{nN_n/2}, \sigma_{n(N_n/2)+1} = \sigma_{n1}, \sigma_{nN_n} = \sigma_{nN_n/2}$ .

The amplitude and phase excitations in the subsets of the entire rings could be rotated to antenna element position for beam steering. To illustrate this, it could be considered an array of just one ring of  $m$  elements, i.e. a circular array. If we select the number of antenna elements as  $N = 6$  (as an example) and there is a constant angular separation of  $60^\circ$  between different beam-steering angles, the optimal excitation for a particular beam-steering angle should also apply to other beam-steering angles by simply substituting  $(w_m, \sigma_m)$  into  $(w_{m+1}, \sigma_{m+1})$  when the beam-steering angle is increased by  $60^\circ$  [PAN 09b]. The rotation properties could be similarly applied for the entire rings as mentioned in the case of one ring. In this design case, we consider a concentric rings array with  $N_r$  rings. The main lobe is steered with an angular step of  $60^\circ$  in azimuth plane. Because of that, the number of elements for the rings is given in the next way  $N_1 = 360^\circ/60^\circ, N_2 = 2N_1, \dots, N_n = nN_1$  elements. In order to keep a uniform element distribution, the radius of the rings are  $r_1, r_2 = 2r_1, r_3 = 3r_1, \dots, r_n = nr_1$ , and the spacing element is set as  $d_n = 2\pi r_n/N_n$ .

Note that the antenna elements distribution does not consider a central element in the origin. In this case, the optimal sets of amplitude and phase excitations for a certain direction of the beam steering  $\phi_0$  should also apply to other directions of beam steering ( $\phi_0 + 60^\circ$ ) by rotating the subsets of amplitude and phase excitations for the rings of the array. Hence, the excitations  $(w_{nm}, \sigma_{nm})$  for the element  $m$  of ring  $n$  are now substituted into the positions  $(w_{n(m+n)}, \sigma_{n(m+n)})$  in the subsets of amplitude and phase excitations of the ring  $n$  for each steering direction. By means of this design, only one optimization is required for beam steering in steps of  $60^\circ$  degrees rather than one optimization for each beam-steering direction [REY 10]. The objective function and the evolutionary algorithm of this design problem are similar to that of the previous design shown with the geometry of the rectangular planar antenna array.

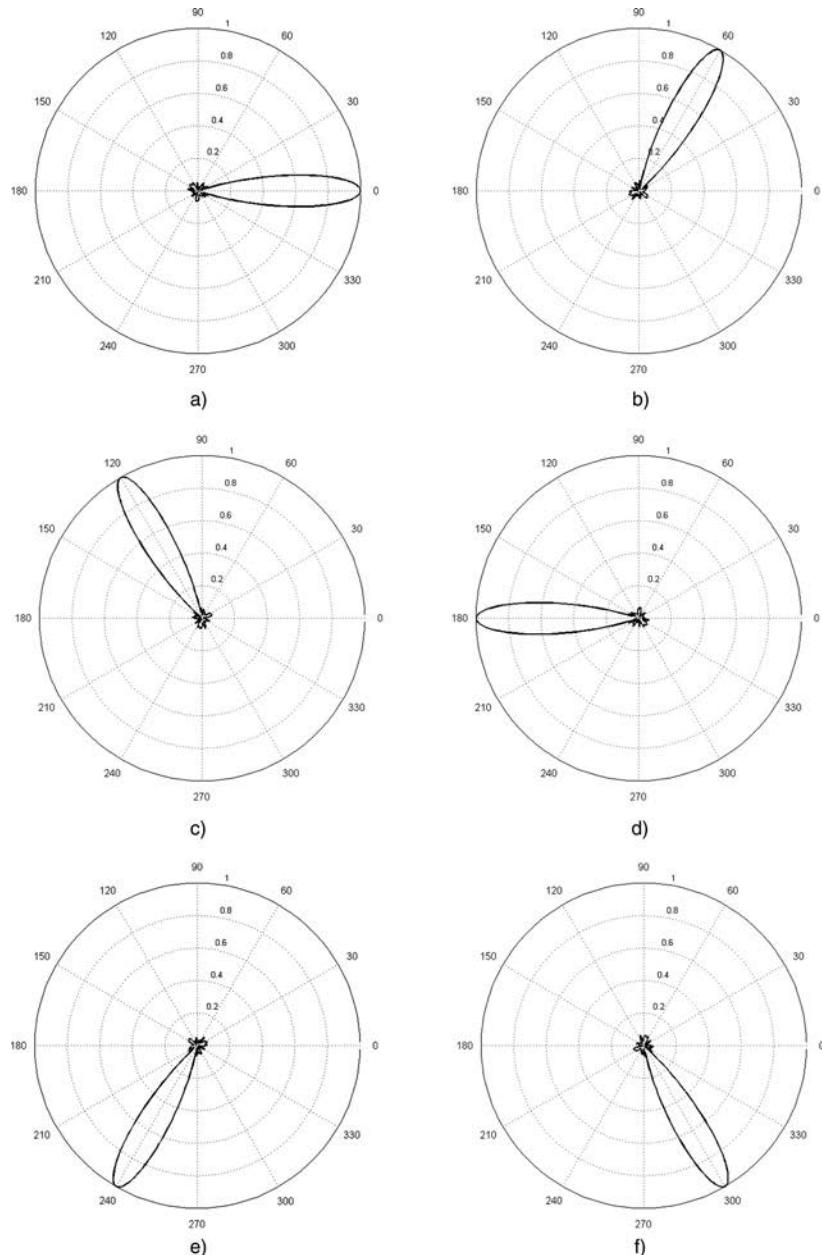
#### 14.3.2. Results obtained for the design of concentric ring arrays

The method of GA was implemented to study the behavior of the array factor in the azimuth plane ( $0^\circ \leq \phi_0 \leq 360^\circ$ ) with an angular step of  $60^\circ$  in the cut of  $\theta = 45^\circ$ . As mentioned previously, the steerable concentric rings array considers a uniform distribution on a plane. To achieve such, we also propose  $N_T = 90$  elements distributed in  $N_r = 5$  rings (see Figure 14.9). Thus, the elements distribution is  $N_1 = 6$ ,  $N_2 = 12$ ,  $N_3 = 18$ ,  $N_4 = 24$ , and  $N_5 = 30$  for the array. Furthermore, the radius for the ring  $n = 1$  is defined as  $r_1 = 0.5\lambda$ . For this configuration, the element spacing is  $d_n 0.5\lambda$ . The population size for GA was set to 200. Two-point crossover along with standard single-point mutation and ranking selection are used, crossover probability  $pc = 0.95$  and mutation probability  $pm = 0.1$ . The stopping criterion is 500 iterations.

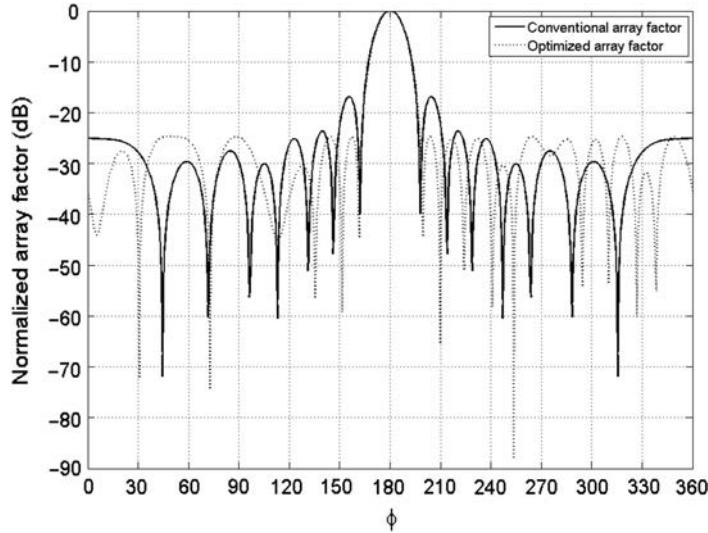


**Figure 14.9.** Concentric ring antenna array with  $N_r = 5$  rings

Figure 14.10 is an illustration of the behavior of the array factor for the beam steering directions of: a)  $\phi_0 = 0^\circ$ , b)  $\phi_0 = 60^\circ$ , c)  $\phi_0 = 120^\circ$ , d)  $\phi_0 = 180^\circ$ , e)  $\phi_0 = 240^\circ$ , and f)  $\phi_0 = 300^\circ$  in the cut of  $\theta = 45^\circ$ . In this case, the set of amplitude and phase excitations provides the normalized array factor with reduction of the side lobe level for the beam-steering directions.



**Figure 14.10.** Behavior of the array factor for a scannable concentric ring antenna array in the azimuth scanning range (a)  $\phi_0 = 0^\circ$ , (b)  $\phi_0 = 60^\circ$ , (c)  $\phi_0 = 120^\circ$ , (d)  $\phi_0 = 150^\circ$ , (e)  $\phi_0 = 180^\circ$ , and (f)  $\phi_0 = 300^\circ$  with the amplitude and phase excitations optimized by the GA for  $N_r = 5$



**Figure 14.11.** Comparison of the array factor between the conventional case and the optimized array factor with GA for  $\phi_0 = 180^\circ$  in the cut of  $\theta = 45^\circ$  for the scannable concentric ring antenna array

Figure 14.11 illustrates the obtained results of the array factor for steerable concentric rings array with the amplitude and phase excitations optimized by GA. In particular, a comparison between the array factor obtained by the GA with respect to the array factor for the conventional case is shown, i.e. uniform amplitudes and conventional case of progressive phase excitations in  $\phi_0 = 180^\circ$ . As expected, it could be appreciated that the evolutionary algorithm notably obtained a reduction of the side lobe level of  $SLL \leq 24$  dB for all the beam-steering directions. The numerical values of the side lobe level and directivity for the optimized array factor shown in Figures 14.10 and 14.11 are presented in Table 14.2. In this table, a better performance of the  $SLL$  for the optimized array factor for GA with respect to the array factor of conventional case could be observed, with very similar values of  $DIR$  whilst maintaining the same aperture. Furthermore, this table shows the amplitude and phase distribution required to provide the steerable array factor. In this case, the sets of the optimized array factor for the direction of  $\phi_0 = 0^\circ$  in the cut of  $\theta = 45^\circ$  are provided. It is important to mention that the obtained values of side lobe level and directivity for the optimized designs are the same when the main lobe is steered in the proposed directions, by rotating the subsets of amplitude and phase excitations of each ring, as mentioned previously. This concentric rings array design could simplify the feeding network for beam steering.

Design cases	SLL (dB)	DIR (dB)	Subsets generated of amplitude excitations	Subsets generated of phase excitations
Conventional case	-16.85	16.45	Uniform	Null
GA	-24.69	16.42	12.9015, 8.2923, 13.9415, 7.6378, 10.0822, 12.3294, 7.0392, 10.6093, 12.1678, 11.4617, 9.5110, 12.5828, 12.0086, 10.7699, 6.8523, 8.0527, 11.1262, 9.0646, 13.7147, 9.7198, 11.9074, 3.3033, 6.8822, 13.8054, 10.5268, 12.3796, 8.0237, 7.0469, 8.3315, 10.8190, 12.3531, 9.4656, 10.0904, 9.4105, 13.7833, 8.3681, 13.6672, 6.8766, 12.8570, 6.8189, 11.6363, 9.4353, 7.8853, 6.7770, 6.9521	0.2252, 0.3028, 6.2187, 5.4334, 2.3174, 5.5525, 6.0432, 6.1526, 0.2015, 0.6108, 5.2642, 5.7834, 1.3069, 4.9479, 1.5308, 1.2285, 0.3278, 0.1392, 5.2154, 1.2879, 0.0052, 5.0622, 0.5903, 5.1782, 6.0481, 6.1694, 0.4777, 5.9340, 4.1003, 0.9716, 6.2586, 0.1365, 6.1502, 1.4738, 4.7567, 5.8460, 0.0134, 6.0667, 0.4768, 0.3449, 0.9378, 6.0610, 0.0995, 0.1732, 0.0323

**Table 14.2.** Numerical values of the side lobe level and the directivity for the array factor illustrated in Figure 14.10: (a)  $\phi_0 = 0^\circ$

#### 14.4. Discussions and open problems

The technology of antenna arrays applied to the modern communication system is very suitable to provide a directive radiation in a wide scanning range. The different geometries of the array play an important role in the performance of the radiation pattern. As seen, the rectangular geometry could provide a wide scanning range by means of certain excitations of the antenna elements for each main lobe direction. An important advantage of this particular geometry is the fact that it is possible to point out the main lobe direction to many directions of the space with a proper performance of the radiation, i.e. reduction of the side lobe level as well as the maximum directivity by optimizing the excitations' coefficients of the array factor. Otherwise, there exist antenna applications that require the reduction of the antenna excitations to provide the scanning of the main lobe in a wide scanning range. For such applications, changing the geometry of the array to a concentric ring antenna array, it is possible to scan the

beam in the azimuth plane by using rotation and symmetry properties of this particular geometry. These properties of this geometry could provide an acceptable solution of a scannable radiation pattern by using less excitation devices. It is important to state that the capacity of the number of directions depends on the number of the antenna elements of the first ring of the array, i.e. applying the rotation and symmetry properties it is possible to reduce the number of antenna excitations, but with a tradeoff with the number of the main lobe directions, the radiation could be reached. In order to satisfy the requirements of modern communication systems, in the technology of antenna arrays, there are open problems such as new geometries of antenna arrays to improve the radiation for particular applications, compensation of the mutual coupling effect, extension of the scanning range, simplification of the feeding network of the antenna array, technological limitations to build antenna arrays, etc.

#### **14.5. Conclusions**

The current demand of new technologies in wireless communications guides the antenna community to develop antenna system, which satisfy the requirements of modern communication system. In this case, this chapter showed several improvements in the antenna array field to improve the performance of the antenna system for applications with requirements of scanning the main beam. The antenna arrays are capable of scanning the beam in a wide scanning range. Two different geometries were studied: rectangular planar antenna array and concentric ring antenna arrays. Simulation results reveal that the design of scannable antenna arrays with amplitude and phase optimized with the use of GAs could provide lower side lobe levels with respect to a conventional phased antenna array. In this case, these results for the optimized designs are achieved with better values of directivity and the same aperture in both geometries.

#### **14.6. Acknowledgments**

This work was supported by the Mexican National Science and Technology Council, CONACyT, under Grant 127919 and by the Autonomous University of Tamaulipas under Grant UAT10-ING-0110.

#### **14.7. Bibliography**

- [BAL 97] BALANIS C., *Antenna Theory-Analysis and Design*, 2<sup>nd</sup> edition, Wiley, New York, 1997.
- [LIA 08] LIANG C.-H., LI L., DANG X.-J., “Inequality condition form grating lobes of planar phased array”, *Progress In Electromagnetics Research B*, vol. 4, pp. 101–113, 2008.

- [HAN 38] HANSEN W., WOODYARD J.R., "A new principle in directional antenna design", *Proceedings of the IRE*, vol. 26, pp. 333–445, 1938.
- [HAU 08] HAUPT R., "Optimized element spacing for low sidelobe concentric ring arrays", *IEEE Transactions on Antenna and Propagation*, vol. 56, no. 1, 2008.
- [PAN 09a] PANDURO M.A., REYNA A., CAMACHO J., "Design of scannable linear arrays with amplitude and phase optimization for maximum side lobe level reduction", *International Journal of Electronics*, vol. 96, no. 3, pp. 323–329, 2009.
- [PAN 09b] PANDURO M.A., BRIZUELA C.A., BALDERAS L.I., ACOSTA D., "A comparison of genetic algorithms, particle swarm optimization and the differential evolution method for the design of scannable circular antenna arrays", *Progress in Electromagnetic Research B*, vol. 13, pp. 171–186, 2009.
- [RAH 99] RAHMAT-SAMII Y., MICHELSSEN E., *Electromagnetic Optimisation by Genetic Algorithms*, Wiley-Interscience, New York, 1999.
- [REY 08] REYNA A., PANDURO M., "Optimization of a scannable pattern for uniform planar antenna arrays to minimize the side lobe level", *Journal of Electromagnetic Waves and Applications*, vol. 22, no. 16, pp. 2241–2250, 2008.
- [REY 10] REYNA A., PANDURO M., "Design of steerable concentric rings array using rotation properties and evolutionary optimization", *Proceedings of the 4<sup>th</sup> European Conference on Antennas and Propagation (EuCAP)*, Barcelona, Spain, 2010.
- [STU 98] STUTZMAN W., THIELE G., *Antenna Theory and Design*, 2<sup>nd</sup> edition, John Wiley and Sons Inc., West Sussex, 1998.



## List of Authors

**Yasmina BESTAOUI**

IBISC UFR

Evry Val d'Essonne University

France

**Pedro CASTILLO**

Heudiasyc UMR-CNRS 6599

Compiègne University of Technology  
France

**Yacine CHALLAL**

Heudiasyc UMR-CNRS 6599

Compiègne University of Technology  
France

**Nikhil CHOPRA**

Mechanical Engineering Dept.

University of Maryland

USA

**David COVARRUBIAS**

CICESE Research Center

Electronics and

Telecommunications Dept.

Ensenada, B.C.

Mexico

**Alejandro DZUL**

Electronic Eng. Dept.

UAM Reynosa-Rodhe

Instituto Tecnológico de la Laguna  
Torreón, Mexico

**Omar ELIZARRARAS**

Electronic Eng. Dept.

UAM Reynosa-Rodhe

Universidad Autónoma de Tamaulipas  
Reynosa, Mexico

**Juan Antonio ESCAREÑO**

Heudiasyc UMR-CNRS 6599

Compiègne University of Technology  
France

**Gerardo FLORES COLUNGA**

Heudiasyc UMR-CNRS 6599

Compiègne University of Technology  
France

**Luis GARCÍA**

Electronic Eng. Dept.

UAM Reynosa-Rodhe

Instituto Tecnológico de la Laguna  
Torreón, Mexico

Luis Rodolfo GARCÍA CARRILLO  
Heudiasyc UMR-CNRS 6599  
Compiègne University of Technology  
France

José Alfredo GUERRERO  
Heudiasyc UMR-CNRS 6599  
Compiègne University of Technology  
France

José Ernesto GOMEZ-BALDERAS  
Heudiasyc UMR-CNRS 6599  
Compiègne University of Technology  
France

Y.-C. LIU  
Mechanical Engineering Dept.  
University of Maryland  
USA

Rogelio LOZANO  
Heudiasyc UMR-CNRS 6599  
Compiègne University of Technology  
France  
and  
LAFMIA UMI 3175  
CINVESTAV, Mexico

Aldo MENDEZ  
Electronic Eng. Dept.  
UAM Reynosa-Rodhe  
Universidad Autónoma de  
Tamaulipas  
Reynosa, Mexico

Marco PANDURO  
Electronic Eng. Dept.  
UAM Reynosa-Rodhe  
Universidad Autónoma de  
Tamaulipas  
Reynosa, Mexico

Alberto REYNA  
Electronic Eng. Dept.  
UAM Reynosa-Rodhe  
Universidad Autónoma de  
Tamaulipas  
Reynosa, Mexico

Gerardo ROMERO  
Electronic Eng. Dept.  
UAM Reynosa-Rodhe  
Universidad Autónoma de  
Tamaulipas  
Reynosa, Mexico

Sergio SALAZAR  
Unité Mixte International  
France-Mexique LAFMIA  
Mexico DF, Mexico

Mark W. SPONG  
University of Texas at Dallas  
Erik Jonsson School of Engineering  
and Computer Science  
Richardson  
Texas  
USA

# Index

## A

adaptive tracking, 77, 79, 81, 83, 88, 89f, 90, 92, 93f, 94  
aerial refueling, 1-3, 135  
affine distortion, 200-201, 205, 206f  
affine nonlinear system, 19  
air density, 106, 120  
ALOHA protocol, 268, 290  
altitude control, 10, 76, 117, 171  
amplitude excitation, 307-309, 314, 315, 319  
angular momentum, 102  
angular motion, 102-103  
angular rate, 103, 115f, 116f, 214, 262  
antenna array, 15, 305-308, 309f-312f, 314f, 316f-318f, 319, 320  
array factor, 306, 308-309, 310f-313f, 314-316, 317f, 318, 319  
attractive force, 168-170, 181, 182, 184, 185  
aviation history, 4-7

## B

balanced graph, 70-73, 77  
blade profile, 105  
body frame, 101, 103, 119, 139, 194

## C

camera model, 197-199, 204, 206, 211  
Canny edge detector, 202, 205  
carrier sense multiple access (CSMA), 15, 268, 272, 286, 290-292, 294-297, 298f, 299-301  
center of mass, 12, 13, 27, 28, 32, 38-40, 41f, 42f, 49, 105, 106, 139, 155, 159, 162, 166, 183, 186, 247, 249, 251, 258, 259, 261f-263f, 264  
chain of integrators, 171, 172  
characteristic equation, 23, 54, 61-63, 67, 68, 71, 107, 108, 111, 112  
code division multiple access (CDMA), 15, 286, 296, 297, 298f, 299-301  
collision avoidance, 12, 148, 167, 169, 235, 268, 272, 286  
concentric ring array, 314-319  
connected graph, 13, 21, 75, 76, 79, 83, 89  
connected set, 23, 24f  
controllability, 13, 27-29, 31, 33-37, 49, 162  
conventional array, 320  
cost function, 52, 53, 55, 308  
cycle, 21, 299

**D**

- delay-free synchronization, 79-81, 83-86
- delay-output synchronize, 78, 82, 83
- directivity, 305, 306, 308, 309, 312f, 318-320
- drag force, 121
- dynamic uncertainty
  - additive, 22
  - multiplicative, 22

**E**

- edge detection, 202, 205
- eigenvalues, 30-32, 54, 84, 85, 178, 179
- eigenvectors, 30-32, 43, 54, 156, 157
- element pattern, 306
- Euclidean norm, 78
- evolution of UAVs, 7-9
- evolutionary algorithm, 308, 309, 315, 318
- extrinsic parameters, 197, 199, 209, 231-233

**F**

- family of polynomials, 23-25
- flight control
  - LQR, 10, 11, 55
  - nonlinear control, 14, 52, 101, 117, 135, 136, 194-195, 248, 264
  - PID, 10, 107f
- formation control, 11, 14, 76, 94, 135, 136, 147-148, 152-153, 159, 166-170, 174, 225, 227, 248-250, 258
- formation flying, 1, 2f, 3, 11, 14, 135, 137, 142, 148, 158f, 159, 160f, 162, 258, 72
- formation flying schemes
  - behavior-based control, 12
  - leader/follower, 11, 12, 27, 51, 136, 267

- passivity-based control, 12-13, 19
- virtual structure, 11, 12, 136, 267
- formation leader tracking, 37-40

**G**

- genetic algorithms (GA/s), 306, 308, 309, 310f-312f, 316, 317f, 318-320
- graph, 12, 13, 19-21, 27, 31, 47, 57, 62, 70, 71f, 77-80, 82
- graph theory, 12, 13, 19-21, 95, 136, 226, 227, 248

**H**

- hamiltonian wind velocity, 252, 254, 255
- Hamilton-Jacobi-Bellman equations, 53, 56
- high-order consensus, 14, 27, 143, 154, 158

**I**

- image rectification, 191, 204-205
- inertia matrix, 102, 140
- inertial frame, 101, 102, 118f, 139, 141, 166, 226, 233, 251, 254, 258
- inhibit sense multiple access (ISMA), 286, 292-295, 296f, 300, 301
- integrators in cascade, 143, 146, 149, 151, 171, 175, 182, 195
- interaction topology
  - balanced, 70, 71, 73
  - chain, 68, 138, 142
  - cyclic, 137, 139, 151
- intrinsic parameters, 197, 199, 205

**K, L**

- k-sector region, 57
- Lagrangian system, 76, 77, 88, 94

Laplacian matrix, 31, 32, 34, 42, 53, 54, 57, 151, 155, 235, 259, 270  
lateral subsystem, 215  
lift force, 103, 121  
longitudinal subsystem, 47, 142, 146, 149, 260  
Luenberger observer, 39, 41f  
Lur'e systems, 51, 52, 57, 60, 61  
Lyapunov candidate function, 55, 128

**M**

maximum Euclidian, 59  
medium access control (MAC)  
protocol, 14, 228, 267, 268, 272, 273, 278, 285-296, 299, 300, 301  
minimal energy, 53  
minimal time, 252, 273  
minimal tracking error, 53  
minimum Euclidian, 59  
multiagent system  
consensus, 12, 13, 51, 54, 62, 73  
coordination, 13, 54, 62  
multiple formations, 187-188

**N**

NED system, 141, 251  
nested saturations, 10, 14, 135, 148, 151, 154, 159, 160f, 166, 170-172, 174, 175, 189, 192, 248, 282  
network simulator (NS2), 268, 273, 278  
networked robotic system, 13, 75, 76, 78, 79, 84, 90, 92f, 93  
Newton's equations, 101

**O**

observability, 13, 27-29, 31, 33, 34, 36, 37, 49, 162

obstacle avoidance, 12, 169, 185-187, 189, 248  
optical flow, 226, 230, 233, 234  
optimal control, 51-55, 61, 63, 67, 71, 73, 248  
orthographic projection, 210, 212f

**P**

parametric uncertainty  
box, 23  
diamond, 23  
sphere, 23  
passivity, 12, 13, 19-20, 76  
path, 21, 95, 99, 251  
path planning, 251-257  
phase excitation, 306-309, 310f, 311f, 315, 316, 317f, 318, 319  
pitch subsystem, 107-109, 115f  
planar antenna array, 306-308, 309f-312f, 315, 320  
polynomial plant, 57, 58, 60  
potential function, 14, 94, 165-167  
progressive phase, 309, 318  
projective distortion, 199, 205, 206f  
PVTOL, 10, 103, 104, 105f

**Q, R**

quad-tilting rotor convertible MAV, 117-139  
quasipolynomials, 112  
repulsive force, 169, 170, 181, 185  
robust stability, 23, 25, 51, 61, 62, 73, 110-112  
robust stability analysis, 56-61, 248  
robust strict positive realness, 51, 57-60  
robust tracking, 13, 75, 77, 84, 85, 90, 91-94  
robustness, 10, 13, 21, 75, 77, 216, 248  
roll subsystem, 104-107, 109-113

rotation matrix, 14, 101, 191, 195, 210, 211, 213, 232

## S

saturation function, 38, 56, 64, 68, 71, 175, 189, 195, 196, 215, 251  
scannable pattern, 15, 305, 306, side lobe level, 305, 306, 308, 309, 312, 316, 318-320  
sign decomposition, 58-60, 73  
skew symmetric matrix, 101  
slotted ALOHA (S-ALOHA) 286-290, 294, 295, 300, 301  
strict positive realness, 51, 57  
strong connectivity, 21, 47, 77, 78, 82-84, 86, 93, 95, 145, 146, 148  
subgraph, 20, 21  
symmetric, positive definite matrix, 53, 94  
symmetric matrix, 101, 201  
synchronization with time delay, 81-83, 86, 87

## T

tailsitter, 6, 9, 13, 100, 103, 109, 114, 115, 131, 132  
thrust, 103, 105, 123, 125, 127, 130f, 140, 141, 167, 194, 214, 228, 238, 250, 269  
time division multiple access (TDMA) 268, 272-275, 278-280  
time varying reference tracking, 155-157  
time delay, 13, 51, 52, 61, 63, 64f, 67, 68, 71-73  
torque, 56, 102, 167, 228, 250

Transport Control Protocol (TCP),

268

transfer function, 54, 57-60, 64, 65, 68, 69, 71, 72, 104, 105, 107, 108

## U, V

UAV classification, 9  
Uncertainty, 13, 19, 21-25, 51-54, 57, 59, 61-63, 66, 70, 71f, 73, 75, 77-79, 89, 109-111, 248  
uncertainty structure  
affine, 25  
independent, 25  
multilinear, 25  
polynomial, 25  
User Datagram Protocol (UDP), 268, 278  
value set, 51, 52, 54, 61-64, 68, 71, 72f, 110-113, 131  
vanishing points, 14, 191, 192, 209  
variable pitch propeller system, 13, 103, 105-107, 113, 131  
vision-based control, 14, 191, 193, 221  
vision-based coordination, 14, 191, 225

## W, Y, Z

wind, 14, 106t, 119-123, 132, 216, 247-249, 251-254, 257-260, 262-264  
wind time-optimal trajectory, 254  
yaw subsystem, 108, 109, 116f, 281f  
Zermelo navigation, 14, 247, 251, 253, 254, 258, 264