Institute for System Dynamics
Prof. Dr.-Ing. O. Sawodny

University of Stuttgart
Germany

Bachelor thesis

Distributed Consensus-based Formation Control of Quadrotors with Formation Feedback

Verteilte, consensusbasierte Formationsregelung von Quadrocoptern mit Rückkopplung aus der Formation

by
Jens Petit

in coorporation with

Keio University
Namerikawa Laboratory

A thesis presented to the faculty in partial fulfillment of the requirements for the degree Bachelor of Science in
Mechatronics

Examiner
Prof. Dr.-Ing. Oliver Sawodny

Supervisors
Prof. Dr. Toru Namerikawa, M.Sc. Daniel Seitz

Examination Date
2016-11-15

Institute for System Dynamics, University of Stuttgart, Waldburgstr. 17/19, 70563 Stuttgart

Abstract

Due to falling prices of electronics, the abundance of cheap computing power and ongoing miniaturization, swarms of autonomous vehicles are starting to become reality. To control such large scale systems, the field of cooperative control was established and has gained significant attention in recent years. A multi-vehicle system may be able to solve tasks that cannot be achieved by a single agent. Applications are various and range from cooperative load carrying to monitoring large areas. In all these applications, formation maintenance plays a key role, which is covered in this work.

Traditionally, a formation consists of multiple vehicles with defined relative positions in a reference coordinate frame. Through a consensus algorithm, a common understanding of the reference frame is achieved and enables the vehicles to move accordingly. Hence, the vehicles follow the trajectory of the reference movement while keeping the geometric configuration in a simple feed forward manner as conventional formation control algorithms do not include feedback from the formation. This idea is realized in order to improve the formation keeping accuracy.

In a first step in the present thesis, a quadrotor is modeled as well as controlled and serves as a vehicle in the formation flight. Secondly, as in traditional formation control, a consensus algorithm enables each quadrotor to have a common understanding of the reference frame. Thirdly, the formation feedback comes through a maximum distance consensus module and consists of information about the maximum physical dimensions of the formation. This knowledge is then used to affect the trajectory of the reference frame. Finally, the performance of the proposed formation control framework is evaluated through simulations and compared to a solution without feedback. The results show a considerable improvement in position keeping accuracy. The formation shape persists recognizable even for fast dynamic movements of the reference frame. As the presented framework is independent of the quadrotor, it can be adapted for other vehicles. Furthermore, through the distributed approach, it is easily scalable and robust to changes in the communication network. An experimental validation could be a topic for future work.

Kurzfassung

Aufgrund von fallenden Elektronikpreisen, immer weiter voranschreitender Miniaturisierung gekoppelt mit einem exponentiellen Anstieg an Rechenleistung sind Schwärme autonomer Fahrzeuge keine ferne Zukunftsvision mehr. Das Forschungsfeld Cooperative Control wurde etabliert um solche Systeme, die aus einer Vielzahl von unabhängigen Agenten bestehen, zu kontrollieren. Ein Verbund aus einzelnen intelligenten Agenten ist in der Lage, komplexe Aufgaben zu lösen, die für ein Individuum nicht realisierbar wären. Dazu gehört zum Beispiel der gemeinsame Transport von Gegenständen oder die Überwachung großer Gebiete. In all diesen Applikationen spielt die genaue räumliche Verteilung der Fahrzeuge, also die Bildung von Formationen, eine entscheidende Rolle.

Eine Formation kann durch relative Positionen in einem gemeinsamen Referenzkoordinatensystem definiert werden. Üblicherweise erzielt ein Consensus Algorithmus ein gemeinsames Verständnis der Referenz in den einzelnen Fahrzeugen und erlaubt so die koordinierte Bewegung des Verbundes. Es handelt sich hierbei um eine einfache Steuerung, da keinerlei Information aus der Formation zurück bezogen werden. Die Idee der Rückkopplung aus der Formation ist in der vorliegenden Arbeit realisiert, um die Positionsgenauigkeit zu erhöhen.

Die Thesis beginnt mit der mathematisch Modellierung und Regelung eines Quadrocopters, der als zugrundeliegendes Fahrzeug dient. Anschließend wird wie oben erwähnt ein Consensus Algorithmus eingesetzt, um den beteiligten Fahrzeugen ein gemeinsames Verständnis des Referenzkoordinatensystems zu geben. Die neuartige Formationsrückkopplung wird durch einen weiteren Consensus Algorithmus erzielt und beinhaltet Information über die maximale physische Ausdehnung der gesamten Formation. Dieses Wissen wird dazu verwendet, die Trajektorie des Referenzkoordinatensystems zu beeinflussen. Abschließend wird die Wirksamkeit der vorgeschlagenen Regelungsstruktur im Vergleich zu einer Lösung ohne Formationsrückkopplung evaluiert. Die Ergebnisse zeigen eine deutlich erhöhte Positionsgenauigkeit. Des Weiteren ist das präsentierte Gesamtsystem unabhängig vom Typ der eingesetzten Fahrzeuge und der verteilte Ansatz erlaubt eine einfache Skalierbarkeit sowie Robustheit gegenüber Änderungen der Kommunikationsstruktur. Ziel zukünftiger Arbeiten könnte sein, die Ergebnisse experimentell zu bestätigen.

Contents

Abbreviations and Formula Symbols

Abbreviations

UAV	Unmanned Aerial Vehicle
DOF	Degrees Of Freedom
COM	Center Of Mass
VL	Virtual Leader
QP	Quadratic Programming
PD	Proportional-Derivative

Formula Symbols

Symbol	Unit	Description
$\mathcal{G}$	[-]	Undirected graph
$\mathcal{N}$	[-]	Set of nodes
$\mathcal{E}$	[-]	Set of edges
n	[-]	Number of vehicles
i	[-]	Generic index
j	[-]	Generic index
$\boldsymbol{A}$	[-]	Adjacency matrix
a_{ij}	[-]	Element of the adjacency matrix
$\boldsymbol{L}$	[-]	Laplacian matrix
l_{ij}	[-]	Element of the laplacian matrix
$\boldsymbol{D}$	[-]	Degree matrix
d_{ij}	[-]	Element of the Degree matrix
$\boldsymbol{A}_{\mathrm{VL}}$	[-]	Adjacency matrix including the VL
σ_i	[-]	Information state of the ith agent
w_i	[-]	Control input of the information state of the ith agent
$\boldsymbol{\sigma}$	[-]	Information state vector of all agents
t	s	Time

Symbol	Unit	Description
Quadrotor		
ω_i	$\mathrm{rad\,s^{-1}}$	Angular velocity of ith rotor
f_i	N	Produced thrust of ith rotor
I_{rot}	$\mathrm{kg\,m^2}$	Moment of inertia of a rotor
k_{lift}	[-]	Positive lift constant
k_{drag}	[-]	Positive aerodynamical drag constant of a rotor
$\tau_{\mathrm{M_i}}$	N m	Produced torque of ith rotor
$\mathcal{F}_W$	[-]	World coordinate frame
$\boldsymbol{O}_W$	[-]	World frame origin
$\boldsymbol{X}_W$	[-]	World frame x-Axis
$\boldsymbol{Y}_W$	[-]	World frame y-Axis
$\boldsymbol{Z}_W$	[-]	World frame z-Axis
$\mathcal{F}_B$	[-]	Body coordinate frame
$\boldsymbol{O}_B$	[-]	Body frame origin
$\boldsymbol{X}_W$	[-]	Body frame x-Axis
$\boldsymbol{Y}_W$	[-]	Body frame y-Axis
$\boldsymbol{Z}_W$	[-]	Body frame z-Axis
${}^W\boldsymbol{R}_B$	[-]	Rotation matrix from body to world coordinates
$\boldsymbol{Y}'$	[-]	Temporary y-Axis
$\boldsymbol{X}''$	[-]	Temporary x-Axis
ϕ	rad	Euler angle
θ	rad	Euler angle
ψ	rad	Euler angle
τ_{roll}	N m	Roll moment
τ_{pitch}	N m	Pitch moment
τ_{yaw}	N m	Yaw moment
$\boldsymbol{r}$	m	Center of mass position in the inertial frame
l	m	Distance from center of mass to motors
$\boldsymbol{\tau}$	N m	External torques
$\boldsymbol{\nu}$	$\mathrm{rad\,s^{-1}}$	Angular velocities in the body fixed frame
p	$\mathrm{rad\,s^{-1}}$	Angular velocity around $\boldsymbol{X}_B$
q	$\mathrm{rad\,s^{-1}}$	Angular velocity around $\boldsymbol{Y}_B$
r	$\mathrm{rad\,s^{-1}}$	Angular velocity around $\boldsymbol{Z}_B$
${}^B\boldsymbol{R}_E$	[-]	Transformation matrix from Euler angles to body fixed frame
$\boldsymbol{f}$	N	Vector of the generalized forces
m	kg	Mass
$\boldsymbol{a}$	$\mathrm{m\,s^{-2}}$	Acceleration of the center of mass
$\boldsymbol{I}$	$\mathrm{kg\,m^2}$	Inertia matrix
I_{xx}	$\mathrm{kg\,m^2}$	Inertia around body x-Axis
I_{yy}	$\mathrm{kg\,m^2}$	Inertia around body y-Axis
I_{zz}	$\mathrm{kg\,m^2}$	Inertia around body z-Axis
f_t	N	Sum of the produced thrust
$\boldsymbol{f}_{\mathrm{aer}}$	N	Aerodynamic resistance force
$\boldsymbol{K}_{\mathrm{aer}}$	[-]	Aerodynamic drag matrix
$k_{\mathrm{aer,x}}$	[-]	Aerodynamic drag constant in x-direction
$k_{\mathrm{aer,y}}$	[-]	Aerodynamic drag constant in y-direction

Symbol	Unit	Description
$k_{\mathrm{aer,z}}$	[-]	Aerodynamic drag constant in z-direction
g	$\mathrm{m\,s^{-2}}$	Gravitational constant
$\boldsymbol{\tau}_{\mathrm{gyro}}$	N m	Gyroscopic moments
Ω	$\mathrm{rad\,s^{-1}}$	Sum of the angular velocities of the rotors
u_1	[-]	Control input
u_2	[-]	Control input
u_3	[-]	Control input
u_4	[-]	Control input
$\tilde{u}_1$	[-]	Altitude controller output
x_{des}	m	Desired x-position
y_{des}	m	Desired y-position
z_{des}	m	Desired z-position
ϕ_{des}	rad	Desired ϕ angle
θ_{des}	rad	Desired θ angle
ψ_{des}	rad	Desired ψ angle
k_{1z}	[-]	Altitude controller parameter
k_{2z}	[-]	Altitude controller parameter
e_z	m	Altitude error
e_ψ	rad	ψ angle error
$k_{1\psi}$	[-]	ψ angle controller parameter
$k_{2\psi}$	[-]	ψ angle controller parameter
k_{1z}	[-]	Altitude controller parameter
x_1	[-]	State space variable
x_2	[-]	State space variable
x_3	[-]	State space variable
x_4	[-]	State space variable
$k_{1\theta}$	[-]	θ angle controller parameter
$k_{2\theta}$	[-]	θ angle controller parameter
e_θ	rad	θ angle error
k_{1x}	[-]	x-position controller parameter
k_{2x}	[-]	x-position controller parameter
e_x	m	x-position error
y_1	[-]	State space variable
y_2	[-]	State space variable
y_3	[-]	State space variable
y_4	[-]	State space variable
$k_{1\phi}$	[-]	ϕ angle controller parameter
$k_{2\phi}$	[-]	ϕ angle controller parameter
e_ϕ	rad	ϕ angle error
k_{1y}	[-]	y-position controller parameter
k_{2y}	[-]	y-position controller parameter
e_y	m	y-position error
$\ddot{z}_{\mathrm{max}}$	$\mathrm{m\,s^{-2}}$	Maximum acceleration in z-direction
$\ddot{z}_{\mathrm{min}}$	$\mathrm{m\,s^{-2}}$	Minimum acceleration in z-direction
$f_{\mathrm{t,max}}$	N	Maximum produced total thrust
$f_{\mathrm{t,min}}$	N	Minimum produced total thrust

Symbol	Unit	Description
$\ddot{x}_{\max}$	$\mathrm{m\,s^{-2}}$	Maximum acceleration in x-direction
$\ddot{x}_{\min}$	$\mathrm{m\,s^{-2}}$	Minimum acceleration in x-direction
$\ddot{y}_{\max}$	$\mathrm{m\,s^{-2}}$	Maximum acceleration in y-direction
$\ddot{y}_{\min}$	$\mathrm{m\,s^{-2}}$	Minimum acceleration in y-direction
$\theta_{\max}$	rad	Maximum θ angle
$\theta_{\min}$	rad	Minimum θ angle
$\phi_{\max}$	rad	Maximum ϕ angle
$\phi_{\min}$	rad	Minimum ϕ angle
$\boldsymbol{a}_{\text{quadpos}}$	$\mathrm{m\,s^{-2}}$	Maximum positive accelerations
$\boldsymbol{a}_{\text{quadneg}}$	$\mathrm{m\,s^{-2}}$	Maximum negative accelerations
Formation Control Framework		
$\boldsymbol{\xi}^r_{\text{contr}}$	[-]	Controlled state of reference frame
$\boldsymbol{\xi}^r$	[-]	Reference information state of reference frame
$\boldsymbol{\xi}_i$	[-]	Reference understanding of ith vehicle
$\boldsymbol{\zeta}_i$	m	Maximum distance information state of ith vehicle
$\mathcal{F}_R$	[-]	Virtual reference coordinate frame
$\boldsymbol{O}_R$	[-]	Origin of the reference frame
$\boldsymbol{X}_R$	[-]	x-Axis of the reference frame
$\boldsymbol{Y}_R$	[-]	y-Axis of the reference frame
$\boldsymbol{Z}_R$	[-]	z-Axis of the reference frame
x_c	m	x-coordinate of reference frame origin
y_c	m	y-coordinate of reference frame origin
z_c	m	z-coordinate of reference frame origin
$\boldsymbol{r}_c$	m	Reference frame origin position
α	rad	Reference frame Euler angle
β	rad	Reference frame Euler angle
γ	rad	Reference frame Euler angle
$\boldsymbol{\delta}$	rad	Reference frame Euler angle vector
$x_{c,i}$	m	x-coordinate of reference frame origin understanding of ith vehicle
$y_{c,i}$	m	y-coordinate of reference frame origin understanding of ith vehicle
$z_{c,i}$	m	z-coordinate of reference frame origin understanding of ith vehicle
α_i	rad	Reference frame Euler angle understanding of ith vehicle
β_i	rad	Reference frame Euler angle understanding of ith vehicle
γ_i	rad	Reference frame Euler angle understanding of ith vehicle
η_i	[-]	Sum row of adjacency matrix
κ	[-]	Constant influencing convergence speed of consensus
$\boldsymbol{r}^d_{rel,i}$	m	ith relative desired quadrotor position
$\boldsymbol{r}^d_i$	m	ith absolute desired quadrotor position
$\boldsymbol{\zeta}^r$	m	Global true maximum distance information state
$\boldsymbol{d}_{\max_i}$	m	Positive maximum distance understanding of ith vehicle
$\boldsymbol{d}_{\min_i}$	m	Maximum negative distance understanding of ith vehicle
d_1	m	First element of maximum distance feedback
d_2	m	Second element of maximum distance feedback
d_3	m	Third element of maximum distance feedback
d_4	m	Fourth element of maximum distance feedback

Symbol	Unit	Description
d_5	m	Fifth element of maximum distance feedback
d_6	m	Sixth element of maximum distance feedback
$\boldsymbol{a}_{\max_1}$	$\mathrm{m\,s^{-2}}$	Maximum acceleration in the formation for the maximum positive distance
$\boldsymbol{a}_{\max_2}$	$\mathrm{m\,s^{-2}}$	Maximum acceleration in the formation for the maximum negative distance
$\boldsymbol{\omega}$	rad	Reference frame angular velocity in inertial frame
$\boldsymbol{d}_{\max}$	m	Positive maximum distance information of reference controller
$\boldsymbol{d}_{\min}$	m	Negative maximum distance information of reference controller
k_p^r	[-]	Reference controller control parameter
k_d^r	[-]	Reference controller control parameter
$\boldsymbol{c}_a$	$\mathrm{m\,s^{-2}}$	Correction vector
$\boldsymbol{u}^r$	$\mathrm{m\,s^{-2}}$	Reference PD controller output
$\boldsymbol{H}_a$	[-]	Weighting matrix for optimization problem
h_1	[-]	Element of weighting matrix for optimization problem
h_2	[-]	Element of weighting matrix for optimization problem
h_3	[-]	Element of weighting matrix for optimization problem
h_4	[-]	Element of weighting matrix for optimization problem
h_5	[-]	Element of weighting matrix for optimization problem
h_6	[-]	Element of weighting matrix for optimization problem
$\boldsymbol{G}_a$	[-]	Constraint matrix for optimization problem
$\boldsymbol{h}_a$	[-]	Constraint vector for optimization problem
$\boldsymbol{G}_1$	[-]	Matrix in optimization problem
$\boldsymbol{h}_1$	[-]	Vector in optimization problem
$\boldsymbol{I}_3$	[-]	3×3 identity matrix
$\tilde{\boldsymbol{G}}_1$	[-]	Matrix in optimization problem constraint
$\tilde{\boldsymbol{h}}_1$	[-]	Vector in optimization problem constraint
$\boldsymbol{G}_2$	[-]	Matrix in optimization problem constraint
$\boldsymbol{h}_2$	[-]	Vector in optimization problem constraint
$\tilde{\boldsymbol{G}}_2$	[-]	Matrix in optimization problem constraint
$\tilde{\boldsymbol{h}}_2$	[-]	Vector in optimization problem constraint
$\boldsymbol{x}_l(s)$	m	Line parametrization
s	[-]	Parametrization variable
$s_{\max}$	[-]	Parametrization variable upper limit
$s_{\min}$	[-]	Parametrization variable lower limit
e_{α_i}	rad	α_i error of ith vehicle
J_α	rad s	α value consensus performance measurement
$\boldsymbol{e}_{r_i}$	m	Difference between desired and actual vehicle position
J_{r_i}	m s	Performance criteria of a single vehicle
J_{total}	m s	Performance criteria of a formation

Chapter 1

Introduction

1.1 Motivation and Research Background

In many ways, exact maintenance of a geometric shape among a group of vehicles moving together in a formation enables powerful systems that can achieve objectives impossible for single vehicles. Such systems with spatial *formation control* promise to have a number of advantages, including flexibility, scalability, lower costs as well as robustness to failures. The concept of formation control has been implemented in different vehicle types: ground vehicles [Gho+10], Unmanned Aerial Vehicles (UAVs) [Kur15], spacecrafts [RB08] or underwater autonomous vehicles [Cui+10]. Likewise, many application scenarios have been evaluated such as cooperative load carrying [BBS16] or large area monitoring [SKB07].

From a more general perspective, the task of formation control can be classified into the field of *cooperative control*. It is characterized through multiple loosely coupled dynamical systems called *agents* which are autonomous and have sensing as well as communication capacities. The aim of the group, a *multi-agent system*, is to achieve a common, sometimes multi-dimensional, objective. Thus, it can be stated that cooperative control consists of four essential key elements: agents, group objective(s), a communication network and individual control algorithms on the local level [BAW11]. Concerning formation control, the group objective is to form and keep a geometric configuration through individual vehicle positioning. The configuration can be predefined (*morphous*) or generated randomly (*amorphous*) [OPA14].

One of the first and most famous concepts is a flocking algorithm which mimics animals' behaviour, like fish or birds, in forming large swarms. The agents act upon three defined influences: collision avoidance, velocity matching and flock centering. Because of its resemblance to natural occurring phenomena, it was named a behavioral approach [Rey87]. However, as the formation shape cannot be influenced, the applications are limited.

In literature, there has not yet been identified a single common categorization of formation control schemes. One popular approach is to distinguish between position-, displacement- and distance-based concepts [OPA14]. In a position-based control scheme each vehicle is able to determine their own position relative to a global coordinate system. Based on this information, they control their movement and achieve the desired formation shape. Contrary, if the algorithm is displacement-based, knowledge of a global coordinate frame is not required. A vehicle only has to sense its neighbors position relative to its own. One step further go distance-based formation control algorithms. Here, only the inter-vehicle distances, not positions, are controlled and driven to a desired value. The orientation of the local coordinate

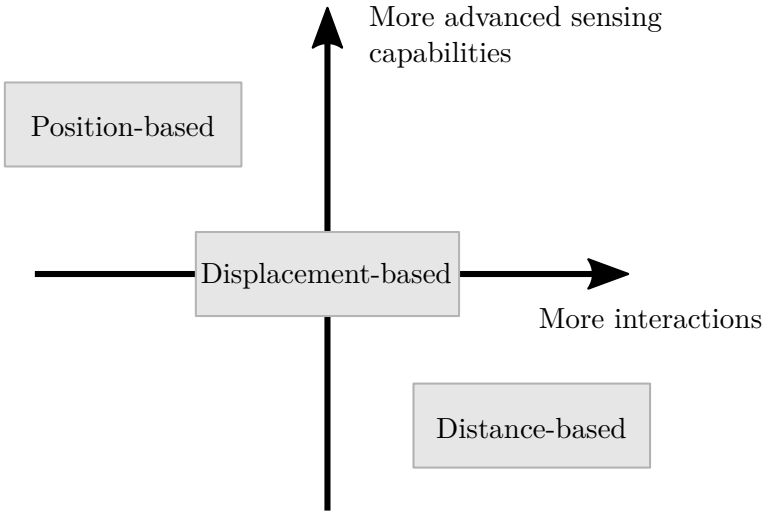


Figure 1.1: Classification of formation control schemes [OPA14].

systems does not matter. Within these three categories, the sensing capabilities have to increase from the position-based, over the displacement-based, to the distance-based approach, whereas the number of interactions between the vehicles decreases as shown in the matrix in Figure 1.1.

Throughout this thesis, a position-based approach is chosen. In that, a formation consists of vehicles with defined desired relative positions in a common coordinate frame, called *virtual reference frame* or simply *reference frame*, as illustrated in Figure 1.2 for six quadrotors. A movement of the formation is achieved through a movement of the virtual reference frame because the quadrotors should keep their assigned positions.

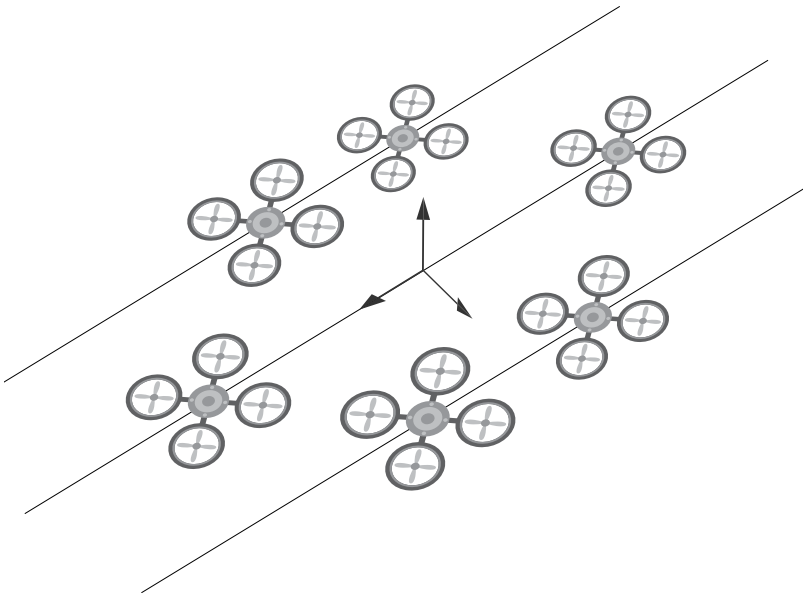


Figure 1.2: Six quadrotors in formation flight.

1.2 Objective and Problem Formulation

A major problem in formation control is that each vehicle has to accelerate differently in magnitude as well as potentially in direction if the movement of the reference frame includes a rotational component. This ultimately leads to high position errors when a vehicle with limited dynamic capabilities is used. In Figure 1.3, the dashed arrows indicate the required accelerations for each vehicle (diamonds) due to the rotation ω of the reference frame. More distant vehicles experience a larger absolute acceleration and lag behind their desired positions most. As a consequence, the formation can be lost. The objective of this work is to control

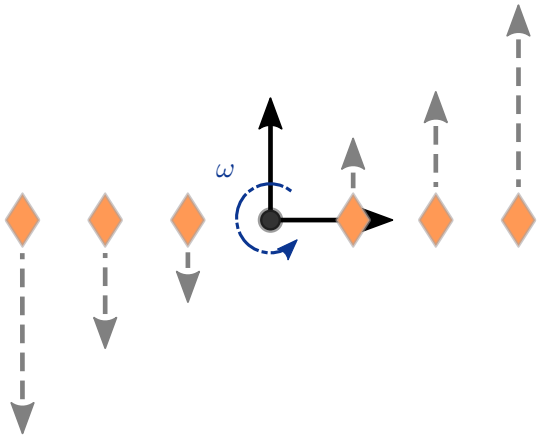


Figure 1.3: Different accelerations in a rotating reference frame.

the movement of the formation and therefore the virtual reference frame in such a way that the occurring maximum required accelerations stay within the possible physical limitations of the vehicles, thus increasing formation maintenance. A second condition is that the approach is distributed as well as flexible which guarantees generic formation shapes, vehicle numbers and communication structures. To verify the proposed formation control framework, a swarm of multiple quadrotors serves as a testbed. Therefore, the focus of this work is neither the quadrotor modeling, nor their control which are a trade-off between accuracy and simplicity.

The present thesis starts with a brief introduction to graph theory for network modeling and consensus algorithms as they build the basis of the formation control framework. In Chapter 3, a quadrotor is modeled using the Newton-Euler approach. To control the vehicle on a local level, a structure of two nested Proportional-Derivative (PD) controllers is used. The following Chapter 4 describes the design of the formation control scheme which consists of three major parts:

- **Reference Consensus Module**. Ensures a common understanding of the position and orientation of the reference frame.
- **Maximum Distance Consensus Module**. Provides the formation feedback in the form of consistent information about the global maximum physical extent in the formation available at every agent.
- **Reference controller**. Uses the available maximum distance information to solve an optimization problem that ultimately corrects the trajectory of the reference in such a way that the constraints of the vehicles are satisfied.

The developed control framework is then tested and validated through simulations with MATLAB. The results are described in Chapter 5, starting with the individual quadrotor controller to the consensus algorithms and finally the entire control scheme without and with reference correction. In Appendix A, the construction of rotation matrices is outlined as they are used throughout this thesis. Lastly, Appendix B introduces the fundamentals of rigid body dynamics.

Chapter 2

Fundamentals

This chapter gives a mathematical introduction of two underlying major aspects in cooperative control, namely network modeling and consensus algorithms. The terminology for the description of graphs is defined. Also, a proof of convergence for a simple consensus algorithm is outlined.

2.1 Network Modeling

An *undirected graph* $\mathcal{G} = (\mathcal{N}, \mathcal{E})$, here simply referenced as graph, can mathematically model a communication network among n independent vehicles where $\mathcal{N} = \{1, 2, \ldots, n\}$ is a finite nonempty set of *nodes*, and $\mathcal{E} \in \mathcal{N} \times \mathcal{N}$ is a set of unordered pairs of nodes, called *edges*. Each node corresponds to a vehicle. An edge (i, j) allows information flow from node i to j and vice versa. If (i, j) is an edge of $\mathcal{G}$, then the nodes i and j are *neighbors*. Self-edges of the form (i, i) are not allowed. A *path* consists of a sequence of nodes such that consecutive nodes are neighbors. The graph $\mathcal{G}$ is called *connected* if there exists a path between every pair of nodes. In the case that every pair of nodes are linked by exactly one path, the graph is a *tree*. A tree which includes all nodes of a graph is a *spanning tree* [GR04].

Graphs can easily be visualized, with numbered circles representing the nodes and lines serving as the edges like in Figure 2.1. Here, the thick lines represent a possible spanning tree. An

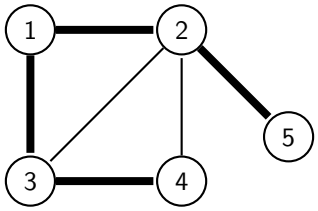


Figure 2.1: Example of a connected graph with five nodes and a spanning tree.

equivalent representation of $\mathcal{G}$ is its adjacency matrix $\boldsymbol{A} = [a_{ij}] \in \mathbb{R}^{n \times n}$. The elements of $\boldsymbol{A}$ are defined as $a_{ij} = a_{ji} = 1$ if $(j, i) \in \mathcal{E}$ and $a_{ij} = a_{ji} = 0$ otherwise. The diagonal entries are zero and the matrix is symmetric. To the example in Figure 2.1, the corresponding adjacency

matrix is

$$\boldsymbol{A} = \boldsymbol{A}^T = \begin{bmatrix} 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix}. \tag{2.1}$$

Each non-zero element a_{ij} states that there is information exchange between the ith and jth vehicle. In a connected graph, information from one vehicle can reach all other vehicles in the network. Another important related matrix is the *graph laplacian*

$$\boldsymbol{L} = [l_{ij}] = \boldsymbol{D} - \boldsymbol{A} \in \mathbb{R}^{n \times n} \tag{2.2}$$

where $\boldsymbol{D} = [d_{ij}] \in \mathbb{R}^{n \times n}$ denotes the *degree matrix* with entries

$$d_{ij} = \begin{cases} \sum_{j=1}^{n} a_{ij} & i = j \\ 0 & i \neq j \end{cases}. \tag{2.3}$$

The diagonal entries are the number of neighbors of node i [GR04].

The model can be expanded by an additional node to include a so called *Virtual Leader (VL)*. This particular node does not represent a physical vehicle, however contains reference information which is distributed to the other nodes. The vehicles 1 to n are referred to as *followers* [RB08]. In Figure 2.2 the VL is added to the previous example from Figure 2.1. The extended adjacency matrix

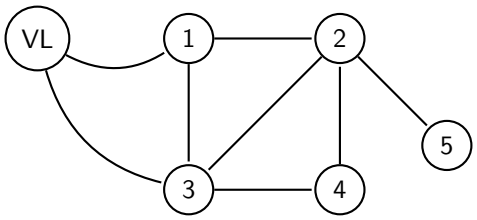


Figure 2.2: Example of a connected graph with five vehicles and a VL.

$$\boldsymbol{A}_{\mathrm{VL}} = \begin{bmatrix} 0 & 1 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 1 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 \end{bmatrix} \in \mathbb{R}^{(n+1)\times(n+1)}. \tag{2.4}$$

including the VL has an additional row and column with entries equal to one according to which vehicle exchanges information with the VL.

2.2 Consensus Algorithm

Sharing information is a necessary prerequisite for cooperative control and can be achieved through consensus algorithms. *Consensus* means to asymptotically converge via local communication to a common understanding of a quantity of interest [OFM07]. The term local expresses that only neighbor-to-neighbor communication is possible.

To introduce the concept of consensus algorithms, let

$$\dot{\sigma}_i(t) = w_i(t), \quad i = \{1, \ldots, n\} \tag{2.5}$$

denote a simple single-integrator information state with the control input w_i at time t. The system consists of n agents linked through a communication structure as explained in Chapter 2.1. As proposed in [OM03], each agent implements the distributed linear protocol

$$w_i(t) = -\sum_{j=1}^{n} a_{ij} \left(\sigma_i(t) - \sigma_j(t)\right), j = \{1, \ldots, n\}, \tag{2.6}$$

which can be rewritten for the whole system as

$$\dot{\boldsymbol{\sigma}}(t) = -\boldsymbol{L}\boldsymbol{\sigma}(t) \tag{2.7}$$

where $\boldsymbol{\sigma}(t) = [\sigma_1(t) \quad \ldots \quad \sigma_n(t)]^T$ and $\boldsymbol{L}$ is the graph laplacian. According to (2.6), each agent takes the locally available information and drives its own information state to the average value of its neighbors. Consensus is reached when $\sigma_1(t) = \ldots = \sigma_n(t)$. For a connected graph, convergence is guaranteed as well as bounded information states. The proof for these assumptions is outlined in the following paragraph.

Let i^* be the index such that

$$\sigma_{i^*}(t) = \max_i \sigma_i(t), \tag{2.8}$$

then, for all $j = \{1, \ldots, n\}$,

$$-a_{i^*j}[\sigma_{i^*}(t) - \sigma_j(t)] \leq 0. \tag{2.9}$$

The inequality is strict unless $a_{i^*j} = 0$ or $\sigma_j(t) = \sigma_{i^*}(t)$. It follows that

$$\dot{\sigma}_{i^*}(t) \leq 0, \tag{2.10}$$

and therefore the maximum never increases and it always decreases if there is any connection with non-maximum agents. Similarly, the minimum never decreases and it always increases if there is any connection with non-minimum agents. A more mathematical formulated proof can be found in [OFM07].

Chapter 3

Quadrotor Modeling and Control

In this chapter, a dynamic model of a quadrotor is developed which serves as a testbed for the formation control framework. The equations of motion are formulated using the Newton-Euler approach. Afterwards, a control strategy is proposed with nested PD controllers for the attitude dynamics and spatial positioning. Lastly, by introducing input constraints, the model becomes limited in accelerations and eventually velocities.

3.1 Basic Concepts

A quadrotor consists of four independent controllable electric motors connected through a rigid cross structure. Each motor is equipped with a propeller to produce upward thrust and torque. The blade's pitch is constant - in contrast to a conventional helicopter with a swashplate. Therefore, the only controllable variables are the angular velocities of the rotors. Because of their mechanical simplicity, quadrotors are one of the most common UAV and a favorite among researchers as well as hobbyists [Gar+13]. A challenge in designing and controlling a quadrotor is the fact that only four control inputs are available (the rotors speeds) whereas the UAV has six Degrees Of Freedom (DOF). Hence, it is an underactuated system [Bou07].

The angular velocity ω_i (for $i = \{1, \dots, 4\}$) of each propeller creates a force

$$f_i = k_{\text{lift}} \omega_i^2 \tag{3.1}$$

in the same upward direction and a torque

$$\tau_{M_i} = k_{\text{drag}} \omega_i^2 + I_{\text{rot}} \dot{\omega}_i, \tag{3.2}$$

where k_{lift} is a positive lift constant, k_{drag} the positive aerodynamical drag constant and I_{rot} denotes the moment of inertia around the rotor axes. This configuration is illustrated in Figure 3.1 with the straight arrows as the thrust and circular arrows as torques. The effect of the term $I_{\text{rot}}\dot{\omega}_i$ is considered small because $\dot{\omega}_i \approx 0$ in stationary flight and is thus omitted. As the rotors' directions are fixed they produce only positiv thrust. To minimize gyroscopic effects and aerodynamic torques in stationary flight, pairs of two rotors rotate clockwise (M_1 and M_3) and anti-clockwise (M_2 and M_4) [Gar+13].

Let $\mathcal{F}_W : \{\boldsymbol{O}_W; \boldsymbol{X}_W; \boldsymbol{Y}_W; \boldsymbol{Z}_W\}$ be an inertial world frame and $\mathcal{F}_B : \{\boldsymbol{O}_B; \boldsymbol{X}_B; \boldsymbol{Y}_B; \boldsymbol{Z}_B\}$ a moving frame fixed to the quadrotor's body at its Center Of Mass (COM). The rotational

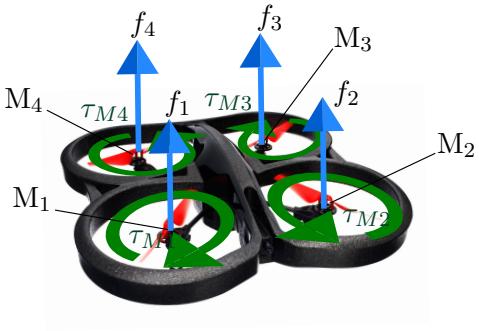


Figure 3.1: Quadrotor motor configuration, thrust and torques.

matrix

$$ {}^{W}\boldsymbol{R}_{B} = \begin{bmatrix} c_\psi c_\theta & c_\psi s_\theta s_\phi - s_\psi c_\phi & c_\psi s_\theta c_\phi + s_\psi s_\phi \\ s_\psi c_\theta & s_\psi s_\theta s_\phi + c_\psi c_\phi & s_\psi s_\theta c_\phi - c_\psi s_\phi \\ -s_\theta & c_\theta s_\phi & c_\theta c_\phi \end{bmatrix} \tag{3.3} $$

transforms vectors from the body frame to the world frame, where c_k represents $\cos(k)$ and s_k is $\sin(k)$. It consists of rotations around the inertial $\boldsymbol{Z}_W$ axis and then the temporary $\boldsymbol{Y}'$ and $\boldsymbol{X}''$ axes (see Appendix A for derivation). The three angles ψ, θ and ϕ are the Euler angles and give the orientation of the body frame (and therefore the quadrotor) relative to the world frame as can be seen in Figure 3.2. Dashed axes indicate a rotation. The vector

$$ \boldsymbol{r} = \boldsymbol{O}_{\mathrm{B}} = \begin{bmatrix} x & y & z \end{bmatrix}^T \tag{3.4} $$

gives the quadrotor's COM position in space in the inertial frame.

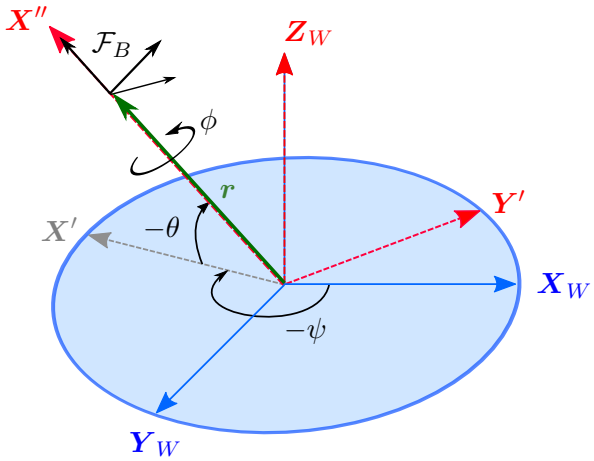


Figure 3.2: Body frame orientation and Euler angles.

From aeronautics, the terms roll, pitch and yaw are applied which indicate movements around the axes of the body frame as shown in Figure 3.3. A roll motion around the $\boldsymbol{X}_B$ axis is achieved through a difference in thrust between the second and fourth motor. Similarly, a pitch movement around the $\boldsymbol{Y}_B$ axis derives from the thrust difference between motor one and three. The sum of the momenta τ_{M1} to τ_{M4} results in a rotation around the $\boldsymbol{Z}_B$ axis.

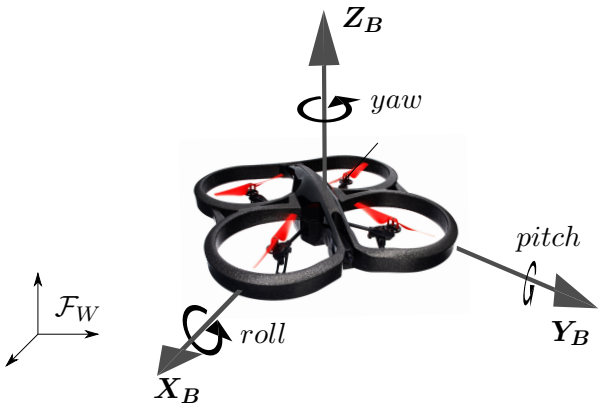


Figure 3.3: Roll, pitch and yaw motion of a Quadrotor.

These three basic rotations change the quadrotors orientation in space. Consequently, the direction of the combined thrust of the four rotors changes with reference to the world coordinate system and enables horizontal maneuverability.

The responsible momenta for the rotations with respect to the COM in the body fixed frame can be summed up as

$$\boldsymbol{\tau} = \begin{bmatrix} (f_2 - f_4)l \\ (f_3 - f_1)l \\ \sum_{i=1}^{4} \tau_{M_i} \end{bmatrix} = \begin{bmatrix} \tau_{\text{roll}} \\ \tau_{\text{pitch}} \\ \tau_{\text{yaw}} \end{bmatrix}, \tag{3.5}$$

where l denotes the distance from the COM to the motors.

The relationship between the angular velocity vector $\boldsymbol{\nu} = \begin{bmatrix} p & q & r \end{bmatrix}^T$ in the body fixed frame and the euler angles velocities is

$$\begin{bmatrix} p \\ q \\ r \end{bmatrix} = \begin{bmatrix} 1 & 0 & -\sin\theta \\ 0 & \cos\phi & \sin\phi\cos\theta \\ 0 & -\sin\phi & \cos\phi\cos\theta \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = {}^{B}\boldsymbol{R}_{E} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix}. \tag{3.6}$$

Left-multiplying $\boldsymbol{\nu}$ with the inverse transformation matrix $\left({}^{B}\boldsymbol{R}_{E}\right)^{-1}$ gives the reverse calculation

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \left({}^{B}\boldsymbol{R}_{E}\right)^{-1} \boldsymbol{\nu} = \begin{bmatrix} 1 & \sin\phi\tan\theta & \cos\phi\tan\theta \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi/\cos\theta & \cos\phi/\cos\theta \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix}. \tag{3.7}$$

3.2 Newton-Euler Model

The quadrotor can be seen as a rigid body with six DOF in space. Assuming that the quadrotor's COM coincides with the body frame origin, its translation and rotational dynamics can be described through the Newton-Euler equations

$$\boldsymbol{f} = m\boldsymbol{a} \tag{3.8}$$

$$\boldsymbol{\tau} = \boldsymbol{I}\dot{\boldsymbol{\nu}} + \boldsymbol{\nu} \times (\boldsymbol{I}\nu) \tag{3.9}$$

where $\boldsymbol{f}$ represents the vector of the external forces, m denotes the quadrotor's mass, $\boldsymbol{a}$ the acceleration of the COM, $\boldsymbol{\tau}$ the vector of the generalized torques [Gro+15].

Using Newton's second law for the motion of a particle, the motion of the COM is determined as if the resulting external force would apply to a particle mass with the bodie's weight concentrated in the COM [Ard06]. Those external forces acting on the quadrotor are the thrust

$$f_{\mathrm{t}} = \sum_{i=1}^{4} f_i \tag{3.10}$$

and the gravitational force. Furthermore, the aerodynamic resistance force

$$\boldsymbol{f}_{\mathrm{aer}} = \boldsymbol{K}_{\mathrm{aer}} \begin{bmatrix} \dot{x}|\dot{x}| & \dot{y}|\dot{y}| & \dot{z}|\dot{z}| \end{bmatrix}^T \tag{3.11}$$

is proportional to the squared velocity of the COM $\dot{\boldsymbol{r}}$ with the aerodynamic translation matrix

$$\boldsymbol{K}_{\mathrm{aer}} = \mathrm{diag}\left(k_{\mathrm{aer,x}}, k_{\mathrm{aer,y}}, k_{\mathrm{aer,z}}\right) \tag{3.12}$$

where the entries are positive constants [Sch13]. This is a highly simplified approximation of the resistance force. Summarized in the inertial frame, the external forces can be expressed by

$$\boldsymbol{f} = {}^{W}\boldsymbol{R}_{B} \begin{bmatrix} 0 \\ 0 \\ f_{\mathrm{t}} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ -mg \end{bmatrix} - \boldsymbol{K}_{\mathrm{aer}} \begin{bmatrix} \dot{x}|\dot{x}| \\ \dot{y}|\dot{y}| \\ \dot{z}|\dot{z}| \end{bmatrix}. \tag{3.13}$$

Using (3.8) one obtains

$$\begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} = \frac{f_t}{m} \begin{bmatrix} c_\psi s_\theta c_\phi + s_\psi s_\phi \\ s_\psi s_\theta c_\phi - c_\psi s_\phi \\ c_\theta c_\phi - mg/f_{\mathrm{t}} \end{bmatrix} - \frac{1}{m} \begin{bmatrix} k_{\mathrm{aer,x}} \dot{x}|\dot{x}| \\ k_{\mathrm{aer,y}} \dot{y}|\dot{y}| \\ k_{\mathrm{aer,z}} \dot{z}|\dot{z}| \end{bmatrix}. \tag{3.14}$$

While it is easier to have the linear equations of motion in the inertial frame, the rotational equations are better expressed in the body frame. If not stated otherwise the equations in the following paragraph are with reference to $\mathcal{F}_{\mathrm{B}}$. It is assumed that the quadrotor has a symmetric structure with its four arms aligned to the bodie's $\boldsymbol{X}_B$- and $\boldsymbol{Y}_B$-axis. Therefore, its inertia matrix

$$\boldsymbol{I} = \begin{bmatrix} I_{xx} & 0 & 0 \\ 0 & I_{yy} & 0 \\ 0 & 0 & I_{zz} \end{bmatrix} \tag{3.15}$$

is a diagonal matrix. Additionally to the external torques $\boldsymbol{\tau}$, gyroscopic effects

$$\boldsymbol{\tau}_{\mathrm{gyro}} = I_{\mathrm{M}}(\boldsymbol{\nu} \times \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix})\Omega, \tag{3.16}$$

have to be taken into account because each motor-rotor unit rotates in the moving body frame. In (3.16), I_{M} is the moment of inertia of one propulsive unit and Ω the sum $\Omega =$

$\omega_1 - \omega_2 + \omega_3 - \omega_4$. Individually, they can be seen as a rigid disks rotating around the $\boldsymbol{Z}_B$-axis with angular velocity ω_i. Now the rotational dynamics can be derived from (3.9) as

$$\begin{aligned} \dot{\boldsymbol{\nu}} &= \boldsymbol{I}^{-1}(-\boldsymbol{\nu} \times \boldsymbol{I}\boldsymbol{\nu} - \boldsymbol{\tau}_{\text{gyro}} + \boldsymbol{\tau}) \\ \begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} &= \begin{bmatrix} (I_{yy} - I_{zz})qr/I_{xx} \\ (I_{zz} - I_{xx})pr/I_{yy} \\ (I_{xx} - I_{yy})pq/I_{zz} \end{bmatrix} - I_{\text{M}} \begin{bmatrix} q/I_{xx} \\ -p/I_{yy} \\ 0 \end{bmatrix} \Omega + \begin{bmatrix} \tau_{\text{roll}}/I_{xx} \\ \tau_{\text{pitch}}/I_{yy} \\ \tau_{\text{yaw}}/I_{zz} \end{bmatrix}. \end{aligned} \tag{3.17}$$

To conclude, the quadrotor dynamics can be modeled through six equations

$$m\ddot{x} = f_{\text{t}}(\sin\phi\sin\psi + \cos\phi\cos\psi\sin\theta) - k_{\text{aer,x}}|\dot{x}|\dot{x} \tag{3.18}$$

$$m\ddot{y} = f_{\text{t}}(\cos\phi\sin\theta\sin\psi - \cos\psi\sin\phi) - k_{\text{aer,y}}|\dot{y}|\dot{y} \tag{3.19}$$

$$m\ddot{z} = f_{\text{t}}\cos\theta\cos\phi - mg - k_{\text{aer,z}}|\dot{z}|\dot{z} \tag{3.20}$$

$$\dot{p} = \frac{I_{yy} - I_{zz}}{I_{xx}}qr - \frac{I_{\text{M}}}{I_{xx}}q\Omega + \frac{\tau_{\text{roll}}}{I_{xx}} \tag{3.21}$$

$$\dot{q} = \frac{I_{zz} - I_{xx}}{I_{yy}}pr + \frac{I_{\text{M}}}{I_{yy}}p\Omega + \frac{\tau_{\text{pitch}}}{I_{yy}} \tag{3.22}$$

$$\dot{r} = \frac{I_{xx} - I_{yy}}{I_{zz}}qr + \frac{\tau_{\text{yaw}}}{I_{zz}}. \tag{3.23}$$

The relationship between the euler angles velocities and the body fixed velocities can be simplified by assuming only small angles θ and ϕ. Under the premise that $\sin\phi = \phi, \sin\theta = \theta$, $\cos\phi, \cos\theta = 1$ and $\dot{\psi} = 0$, the transformation matrix ${}^{B}\boldsymbol{R}_{E}$ from (3.6) equals the 3×3 identity matrix $\boldsymbol{I}_3$. It follows that

$$\begin{bmatrix} p \\ q \\ r \end{bmatrix} = \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix}. \tag{3.24}$$

Furthermore, the quadratic terms involving two angles as well as cross-coupling effects in (3.21) to (3.23) are omitted assuming near hovering state [Fer+13]. Thus, the quadrotor rotational dynamics can be modeled by

$$\ddot{\phi} = \tau_{\text{roll}}/I_{xx} \tag{3.25}$$

$$\ddot{\theta} = \tau_{\text{pitch}}/I_{yy} \tag{3.26}$$

$$\ddot{\psi} = \tau_{\text{yaw}}/I_{zz}. \tag{3.27}$$

Together with the three equations (3.18), (3.19) and (3.20), they constitute the complete dynamical model. It is strongly simplified and ignores various effects such as blade flapping which contribute to the highly nonlinear dynamics of a quadrotor.

3.3 Control Strategy

The aim of the quadrotor controller is to stabilize the vehicle and achieve acceptable positioning speeds. For this, feedback from the quadrotor comes through an on-board inertial measurement unit for the angular velocities and a global position measuring system realized through an external motion capture setup. As a controller, a cascaded control strategy

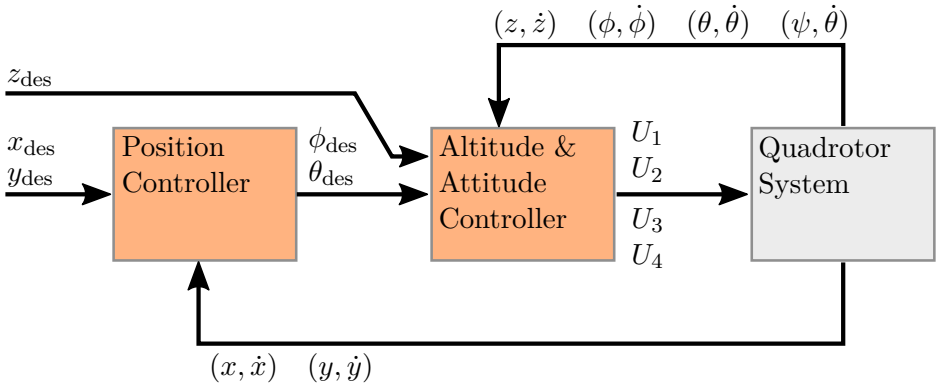


Figure 3.4: Hierarchical control structure [Gar+13].

as proposed in [Gar+13] is implemented because the rotational and altitude dynamics are considerably faster than the horizontal ones. Specifically, state feedback controls corresponding to PD controllers stabilize the subsystems as illustrated in Figure 3.4. To control the quadrotor, four control inputs

$$\begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix} = \begin{bmatrix} f_{\text{t}} \\ \tau_{\text{roll}}/I_{xx} \\ \tau_{\text{pitch}}/I_{yy} \\ \tau_{\text{yaw}}/I_{zz} \end{bmatrix} = \begin{bmatrix} k_{\text{lift}} & k_{\text{lift}} & k_{\text{lift}} & k_{\text{lift}} \\ 0 & \frac{k_{\text{lift}} l}{I_{xx}} & 0 & -\frac{k_{\text{lift}} l}{I_{xx}} \\ -\frac{k_{\text{lift}} l}{I_{yy}} & 0 & \frac{k_{\text{lift}} l}{I_{yy}} & 0 \\ \frac{k_{\text{drag}}}{I_{zz}} & \frac{k_{\text{drag}}}{I_{zz}} & \frac{k_{\text{drag}}}{I_{zz}} & \frac{k_{\text{drag}}}{I_{zz}} \end{bmatrix} \begin{bmatrix} \omega_1^2 \\ \omega_2^2 \\ \omega_3^2 \\ \omega_4^2 \end{bmatrix} \tag{3.28}$$

are chosen. In (3.28), the physical control variables ω_1^2 to ω_4^2 are uniquely solved to the chosen control inputs. This results in the following system

$$\ddot{x} = \frac{1}{m}\left[u_1(\sin\phi\sin\psi + \cos\phi\cos\psi\sin\theta) - k_{\text{aer,x}}|\dot{x}|\dot{x}\right] \tag{3.29}$$

$$\ddot{y} = \frac{1}{m}\left[u_1(\cos\phi\sin\theta\sin\psi - \cos\psi\sin\phi) - k_{\text{aer,y}}|\dot{y}|\dot{y}\right] \tag{3.30}$$

$$\ddot{z} = \frac{1}{m}\left[u_1\cos\theta\cos\phi - mg - k_{\text{aer,z}}|\dot{z}|\dot{z}\right] \tag{3.31}$$

$$\ddot{\phi} = u_2 \tag{3.32}$$

$$\ddot{\theta} = u_3 \tag{3.33}$$

$$\ddot{\psi} = u_4. \tag{3.34}$$

for $t > 0$ with the initial conditions $x(0) = x_0$, $\dot{x}(0) = v_{x_0}$, ..., $\theta(0) = \theta_0$, $\dot{\theta}(0) = \dot{\theta}_0$, ...

3.4 Position and Attitude Control

3.4.1 Altitude and Yaw Control

The altitude dynamics in (3.31) are reduced to a linear double integrator system by implementing feedback linearization through

$$u_1 = (\tilde{u}_1 + mg)\frac{1}{\cos\theta\cos\phi}, \tag{3.35}$$

where $\tilde{u}_1$ denotes the controller output [Gar+13]. Through state feedback control, the poles can be placed in the open left half plane and the closed loop system stabilized. This corresponds to a PD controller

$$\begin{aligned}\tilde{u}_1 &= k_{1z} z_{\text{des}} - k_{1z} z - k_{2z} \dot{z} \\ &= k_{1z} e_z - k_{2z} \dot{z}\end{aligned} \tag{3.36}$$

with the gains k_{1z}, k_{2z} to influence convergence speed and the control error $e_z = z_{\text{des}} - z$. Using the control input u_1 in (3.29) to (3.31) gives

$$m\ddot{x} = (\tilde{u}_1 + mg)\left(\frac{\sin\psi\tan\phi}{\cos\theta} + \cos\psi\tan\theta\right) - k_{\text{aer,x}}|\dot{x}|\dot{x} \tag{3.37}$$

$$m\ddot{y} = (\tilde{u}_1 + mg)\left(\sin\psi\tan\theta - \frac{\cos\psi\tan\phi}{\cos\theta}\right) - k_{\text{aer,y}}|\dot{y}|\dot{y} \tag{3.38}$$

$$m\ddot{z} = k_{1z} e_z - k_{2z}\dot{z} - k_{\text{aer,z}}|\dot{z}|\dot{z} \tag{3.39}$$

For the yaw angle subsystem (3.34) the same controller

$$u_4 = k_{1\psi} e_\psi - k_{2\psi}\dot{\psi} \tag{3.40}$$

is applied, with the gains $k_{1\psi}, k_{2\psi}$ and the control error $e_\psi = \psi_{\text{des}} - \psi$. As both subsystems for yaw and altitude control are asymptotically stable with fast dynamics, it can be assumed $z \to z_{\text{des}}$, $\psi \to \psi_{\text{des}}$ and $\tilde{u}_1 \to 0$ for a time t_1 large enough. The control gains should be chosen to ensure a fast and well-damped response.

The desired yaw angle ψ_{des} is set to zero, therefore (3.37) and (3.38) reduce to

$$\ddot{x} = g\tan\theta - \frac{k_{\text{aer,x}}|\dot{x}|\dot{x}}{m} \tag{3.41}$$

$$\ddot{y} = -g\frac{\tan\phi}{\cos\theta} - \frac{k_{\text{aer,y}}|\dot{y}|\dot{y}}{m}. \tag{3.42}$$

3.4.2 Control of x-Position and Pitch Angle

To fulfill the assumption of

$$tan(\theta) \approx \theta, \tag{3.43}$$

a very small upper bound on $|\theta|$ is set which linearizes (3.41) and gives

$$\ddot{x} = g\theta - \frac{k_{\text{aer,x}}|\dot{x}|\dot{x}}{m} \tag{3.44}$$

$$\ddot{\theta} = u_3 \tag{3.45}$$

which can be described as a chain of four integrators

$$\dot{x}_1 = x_2 \tag{3.46}$$

$$\dot{x}_2 = gx_3 - \frac{k_{\text{aer,x}}|x_2|x_2}{m} \tag{3.47}$$

$$\dot{x}_3 = x_4 \tag{3.48}$$

$$\dot{x}_4 = u_3, \tag{3.49}$$

with $x_1 = x$, $x_2 = \dot{x}$, $x_3 = \theta$, $x_4 = \dot{\theta}$. The goal is to control the x-position. This is achieved by a cascaded controller where the inner loop controls the pitch angle θ and the outer loop sets the reference value θ_{des} (see Figure 3.4). As for the yaw angle, a stabilizing state feedback controller

$$u_3 = k_{1\theta} e_\theta - k_{2\theta} \dot{\theta}, \tag{3.50}$$

where $k_{1\theta}$, $k_{2\theta}$ are the gains and e_θ the error between θ_{des} and actual θ, is implemented. In order to be able to omit the angular dynamics for the position controller, the dynamics of the inner system have to be fast [Lun14]. Based on the actual x-position and velocity, the secondary control loop sets the desired angle to

$$\theta_{\text{des}} = k_{1x} e_x - k_{2x} \dot{x}, \tag{3.51}$$

where k_{1x}, k_{2x} are the gains and e_x the error between x_{des} and actual x. The output θ_{des} is limited to a small upper bound to fulfill (3.43).

3.4.3 Control of y-Position and Roll Angle

Using the small-angle approximation for

$$\tan\phi \approx \phi, \qquad \cos\theta \approx 1, \tag{3.52}$$

a very small upper bound on $|\phi|$ is set which linearizes (3.42) and gives

$$\ddot{y} = -g\phi - \frac{k_{\text{aer,y}} |\dot{y}| \dot{y}}{m} \tag{3.53}$$

$$\ddot{\phi} = u_2 \tag{3.54}$$

which can be described as a chain of four integrators

$$\dot{y}_1 = y_2 \tag{3.55}$$

$$\dot{y}_2 = -g y_3 - \frac{k_{\text{aer,y}} |y_2| y_2}{m} \tag{3.56}$$

$$\dot{y}_3 = y_4 \tag{3.57}$$

$$\dot{y}_4 = u_2, \tag{3.58}$$

with $y_1 = y$, $y_2 = \dot{y}$, $y_3 = \phi$, $x_4 = \dot{\phi}$. The goal is to control the y-position. This is achieved by a cascaded controller where the inner loop controls the roll angle ϕ and the outer loop sets the reference value ϕ_{des} (see Figure 3.4). As for the yaw angle, a stabilizing state feedback controller

$$u_2 = k_{1\phi} e_\theta - k_{2\phi} \dot{\phi}, \tag{3.59}$$

where $k_{1\phi}$, $k_{2\phi}$ are the gains and e_ϕ the error between ϕ_{des} and actual ϕ, is implemented. In order to be able to omit the angular dynamics for the position controller, the dynamics of the inner system have to be fast [Lun14]. Based on the actual y-position and velocity, the secondary control law sets the desired angle

$$\phi_{\text{des}} = k_{1x} e_y - k_{2y} \dot{x}, \tag{3.60}$$

where k_{1y}, k_{2y} are the gains and e_y the error between y_{des} and actual y. The output ϕ_{des} is limited to a small upper bound to fulfill (3.52).

3.5 Acceleration Limits

The control input u_1 corresponds to the total thrust produced by the four rotors and is thus limited by an upper bound $f_{\mathrm{t,max}}$ and lower bound $f_{\mathrm{t,min}} = 0$ as the rotors direction is fixed. This limits the maximum acceleration and velocity due to the modeled air friction. For zero pitch and roll angle, the maximum acceleration in $\boldsymbol{Z}_W$ direction derived from (3.31) is

$$\ddot{z}_{\max} = \frac{f_{\mathrm{t,max}} - k_{\mathrm{aer,z}}|\dot{z}|\dot{z}}{m} - g \tag{3.61}$$

$$\ddot{z}_{\min} = \frac{f_{\mathrm{t,min}} - k_{\mathrm{aer,z}}|\dot{z}|\dot{z}}{m} - g = -\frac{k_{\mathrm{aer,z}}|\dot{z}|\dot{z}}{m} - g. \tag{3.62}$$

A similar examination can be made in the horizontal $x\,y$-plane. Here, the maximum pitch and roll angle limit the acceleration. Assuming a constant altitude and an angle range of $-|\theta_{\max}| \leq \theta \leq \theta_{\max}$ and $-|\phi_{\max}| \leq \phi \leq \phi_{\max}$ respectively gives

$$\ddot{x}_{\max} = -\ddot{x}_{\min} = g \tan\theta_{\max} - \frac{k_{\mathrm{aer,x}}|\dot{x}|\dot{x}}{m} \tag{3.63}$$

$$\ddot{y}_{\min} = -\ddot{y}_{\max} = -g\frac{\tan\phi_{\max}}{\cos\theta_{\max}} - \frac{k_{\mathrm{aer,y}}|\dot{y}|\dot{y}}{m} \tag{3.64}$$

for the maximum accelerations in positive as well as negative coordinate directions. Summarizing these constraints into single vectors gives

$$\boldsymbol{a}_{\mathrm{quadpos}} = \begin{bmatrix} \ddot{x}_{\max} \\ \ddot{y}_{\max} \\ \ddot{z}_{\max} \end{bmatrix}, \tag{3.65}$$

$$\boldsymbol{a}_{\mathrm{quadneg}} = \begin{bmatrix} \ddot{x}_{\min} \\ \ddot{y}_{\min} \\ \ddot{z}_{\min} \end{bmatrix}. \tag{3.66}$$

Chapter 4

Formation Control Framework

In this chapter, the complete formation control framework as the main contribution of the thesis is outlined. First an overview of the formation control scheme is given before introducing the three parts in detail: the reference consensus module, the maximum distance consensus module and the reference controller. As for the reference consensus module, it is adopted from [RS08], whereas the latter two represent a novel extension to distributed formation control.

4.1 Concept

Through the proposed quadrotor controller, each vehicle is able to position itself in space. However, for a formation to form, it also needs an understanding of its desired position which is dependent on the current reference frame status where the relative desired positions of each vehicle $\boldsymbol{r}^d_{\text{rel,i}}$ are defined. Let $\boldsymbol{\xi}^r_{\text{contr}}$ be the state of this coordinate frame $\mathcal{F}_R : \{\boldsymbol{O}_R; \boldsymbol{X}_R; \boldsymbol{Y}_R; \boldsymbol{Z}_R\}$. In Figure 4.1, a triangular formation consisting of three vehicles is illustrated. The round dots are the desired positions of the agents set in $\mathcal{F}_R$ and the diamonds represent the actual locations $\boldsymbol{r}_i$. The formation can move through a change of the reference state $\boldsymbol{\xi}^r_{\text{contr}}$ of which each agent carries his own understanding $\boldsymbol{\xi}_i$.

Subsequently, the task of the so-called reference consensus module is to ensure a common understanding which is achieved if

$$\boldsymbol{\xi}_1 = \ldots = \boldsymbol{\xi}_n = \boldsymbol{\xi}^r_{\text{contr}}. \tag{4.1}$$

Contrary to Figure 4.1, a configuration where consensus is not reached is shown in Figure 4.2. Here, the desired vehicle positions deviate from the triangular configuration because the $\boldsymbol{\xi}_i$ information states have not converged. The reference consensus module is implemented locally in each vehicle and forms a part of the larger formation control framework as presented in Figure 4.3. The illustrated communication network allows information exchange with the index j indicating neighbors. Through the reference controller comes the information about the reference position and orientation $\boldsymbol{\xi}^r_{\text{contr}}$ comes. It takes a desired reference frame trajectory $\boldsymbol{\xi}^r$ and provides a corrected output $\boldsymbol{\xi}^r_{\text{contr}}$ which is subsequently distributed over the communication network through the before mentioned reference consensus module. The corrected output considers the dynamic limitations of the involved vehicles and ultimately leads to reduced position errors. For the reference controller to calculate its correction it needs information about the physical extent of the formation. The reason for this is that the maximum accelerations occur for the most distant vehicles as stated in the problem formulation. Therefore, the formation feedback comes in form of a $\boldsymbol{\zeta}_i$ information state of

each quadrotor carrying the maximum distance data. It is provided by the maximum distance consensus module which is, as the reference consensus module, implemented locally in each quadrotor. Contrary, the reference controller exists only once outside of the vehicles.

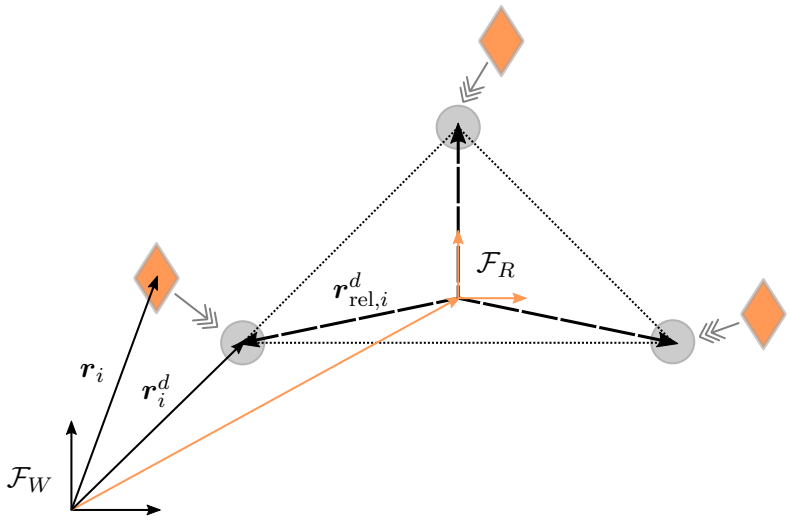


Figure 4.1: Triangular formation with common reference frame understanding.

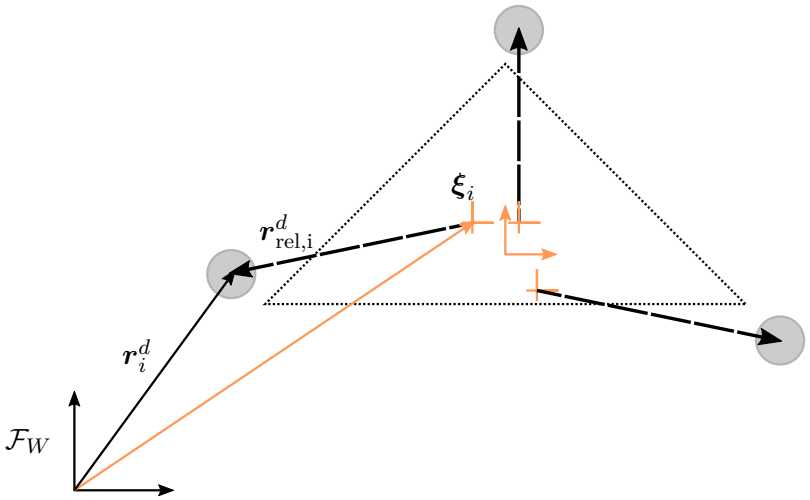


Figure 4.2: Desired vehicle positions with inconsistent reference frame understanding.

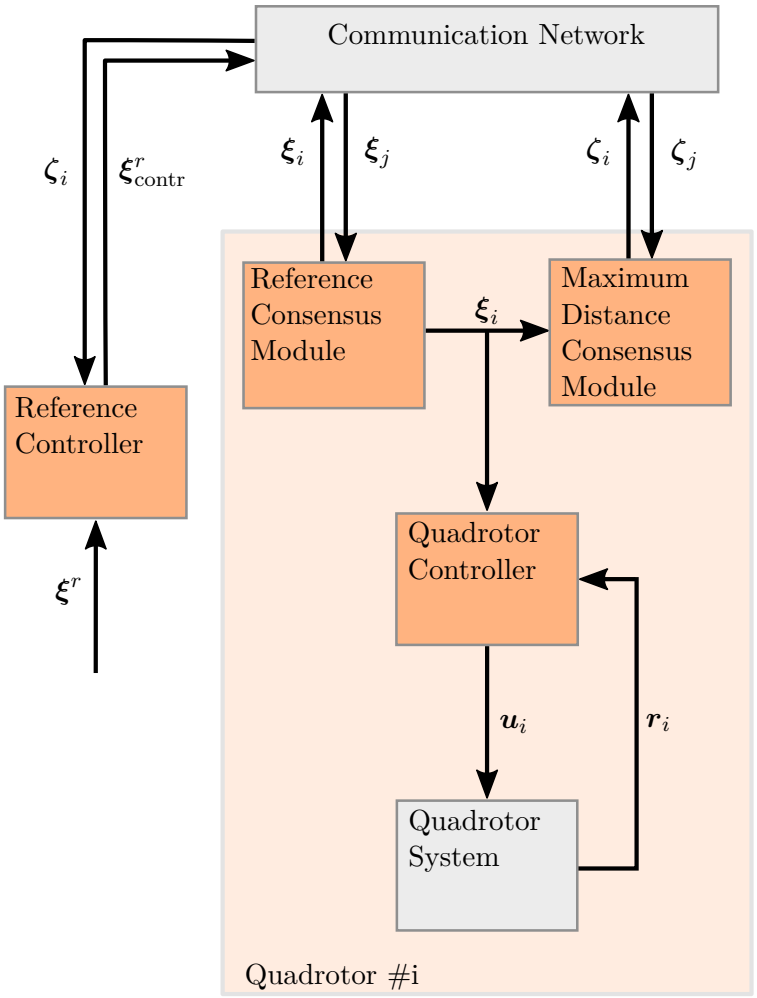


Figure 4.3: Overview of the complete formation control framework.

4.2 Reference Consensus Module

The reference frame state $\boldsymbol{\xi}^r_{\text{contr}}$ can be described in space through its center position in the inertial frame $\mathcal{F}_W$ and three Euler angles defined as in Figure 4.4 as

$$\boldsymbol{\xi}^r_{\text{contr}} = \Big[\underbrace{x_c \quad y_c \quad z_c}_{\boldsymbol{r}_c} \quad \underbrace{\alpha \quad \beta \quad \gamma}_{\boldsymbol{\delta}}\Big]^T. \tag{4.2}$$

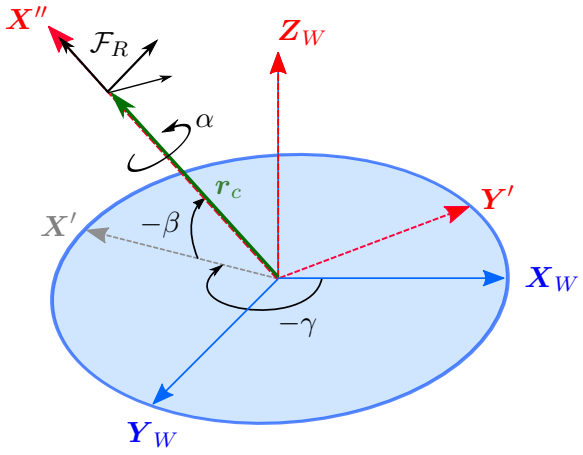


Figure 4.4: Reference frame orientation and Euler angles.

Each vehicle i has his own understanding

$$\boldsymbol{\xi}_i = \begin{bmatrix} x_{c,i} & y_{c,i} & z_{c,i} & \alpha_i & \beta_i & \gamma_i \end{bmatrix} \tag{4.3}$$

of the reference frame with $\dot{\boldsymbol{\xi}}_i$ being the control input. The aim is to drive its own value $\boldsymbol{\xi}_i$ to the desired value $\boldsymbol{\xi}^r_{\text{contr}}$. Therefore, an extended consensus law

$$\dot{\boldsymbol{\xi}}_i = \underbrace{\frac{1}{\eta_i}\sum_{j=1}^{n} a_{ij}\left[\dot{\boldsymbol{\xi}}_j - \kappa(\boldsymbol{\xi}_i - \boldsymbol{\xi}_j)\right]}_{\text{Neighbor's influence}} + \underbrace{\frac{1}{\eta_i} a_{i(n+1)}\left[\dot{\boldsymbol{\xi}}^r_{\text{contr}} - \kappa(\boldsymbol{\xi}_i - \boldsymbol{\xi}^r_{\text{contr}})\right]}_{\text{Reference influence}} \tag{4.4}$$

based on Chapter 2.2 is introduced where a_{ij} is the (i,j) entry of the extended adjacency matrix $\boldsymbol{A}_{\text{VL}}$, κ is a positive scalar, and η_i is the ith row sum of $\boldsymbol{A}_{\text{VL}}$. The first term uses the information states of the vehicle neighbors whereas the second term considers a possible connection to the desired value $\boldsymbol{\xi}^r_{\text{contr}}$. In addition to the simple differences in the basic consensus law, a derivative term is added which allows the algorithm to follow any time-variant reference value. The row sum η_i ensures an equally weighted influence for every vehicle regardless of its number of connections. For a connected graph, consensus to the reference value is guaranteed [RS08]. In Figure 4.5, a scenario with six quadrotors and a communication network indicated by the dashed lines is illustrated. Here, only one vehicle receives the reference value. The remaining five agents update their information states solely based on the values of their neighbors.

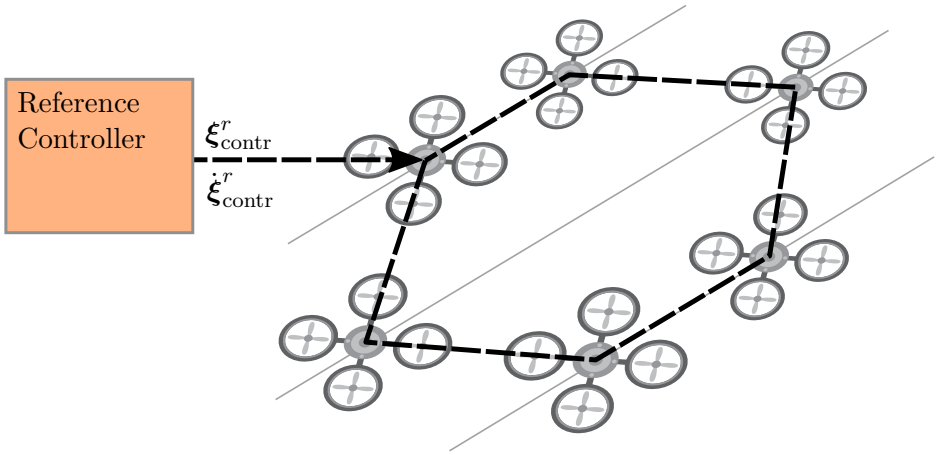


Figure 4.5: Reference frame consensus with six vehicles.

Each vehicle i uses its own understanding $\boldsymbol{\xi}_i$ and its known relative desired position $\boldsymbol{r}^d_{rel,i}$ with respect to the virtual coordinate frame to calculate its desired absolute position

$$\boldsymbol{r}^d_i = \begin{bmatrix} x_{c,i} \\ y_{c,i} \\ z_{c,i} \end{bmatrix} + {}^W\boldsymbol{R}_R \boldsymbol{r}^d_{rel,i} \tag{4.5}$$

where ${}^W\boldsymbol{R}_R$ is the rotation matrix from reference frame to world coordinates based on the three euler angles α_i, β_i, γ_i (Appendix A). Finally, the desired absolute position is passed on to the local quadrotor controller to position the vehicle.

4.3 Maximum Distance Consensus Module

Similarly to the reference consensus module, the task of the maximum distance consensus module is to ensure a common group understanding of the maximum dimensions of the formation. The motivation is that this information should be accessible to the reference controller through any agent in the network as feedback from the formation. Thereby, the flexibility of the framework is guaranteed. The maximum dimensions of a formation refer to the most distant desired vehicle position in each coordinate direction from the origin of the virtual reference frame as illustrated in Figure 4.6. Mathematically, it can be expressed as

$$\boldsymbol{\zeta}^r = \begin{bmatrix} \max(\boldsymbol{r}^d_{rel,i}) \\ \min(\boldsymbol{r}^d_{rel,i}) \end{bmatrix} \in \mathbb{R}^6, \quad i = \{1, \ldots, n\} \tag{4.6}$$

with $\boldsymbol{\zeta}^r$ being the global correct value. As each vehicle inherently only has knowledge about its own desired position, it has to exchange information.

Let

$$\boldsymbol{\zeta}_i = \begin{bmatrix} \boldsymbol{d}_{\max_i} \\ \boldsymbol{d}_{\min_i} \end{bmatrix} \in \mathbb{R}^6, \quad i = \{1, \ldots, n\} \tag{4.7}$$

be each vehicle's own understanding of the maximum positive $\boldsymbol{d}_{\max_i}$ as well as negative distances $\boldsymbol{d}_{\min_i}$ in the formation. To achieve consensus, a so-called synchronous maximum

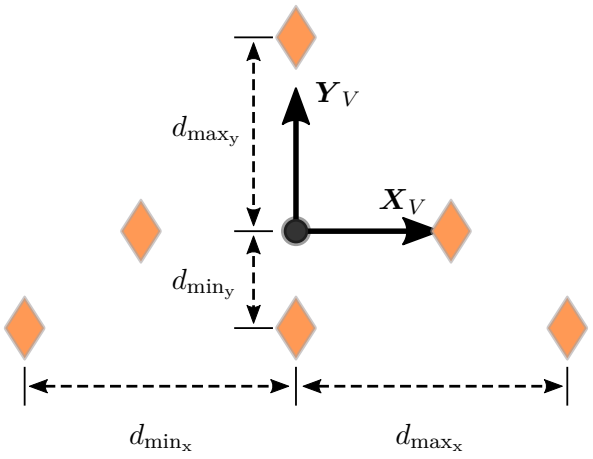


Figure 4.6: 2D maximum postive and negative distance to virtual center.

broadcast algorithm after [Lyn96] is applied

$$\boldsymbol{\zeta}_j = \begin{bmatrix} \max\left(\boldsymbol{d}_{\max_j}, \boldsymbol{d}_{\max_i}\right) \\ \min\left(\boldsymbol{d}_{\min_j}, \boldsymbol{d}_{\min_i}\right) \end{bmatrix}, \tag{4.8}$$

where $j \in \mathcal{N}_i$ and $\mathcal{N}_i$ is the set of neighbors of node i. Each vehicle sends out his information state $\boldsymbol{\zeta}_i$ and all connected nodes update their own information states $\boldsymbol{\zeta}_j$ only if the value is greater or smaller respectively. The information state is initialized as $\boldsymbol{\zeta}_i = \begin{bmatrix} \boldsymbol{r}^d_{rel,i} & \boldsymbol{r}^d_{rel,i} \end{bmatrix}^T$ before any information exchange takes place. The consensus algorithm is distributed in the sense that only local neighbor information exchange is required. The virtual leader can connect to any follower or subgroup of vehicles and obtain $\boldsymbol{\zeta}_i$. It can be shown that the consensus algorithms guarantees $\boldsymbol{\zeta}_i(t) \to \boldsymbol{\zeta}^r$ if a spanning tree exists [Boy+04].

4.4 Reference Controller

The reference controller has knowledge of the current as well as desired state of the reference frame $\boldsymbol{\xi}^r_{\text{contr}}$ and $\boldsymbol{\xi}^r$. Furthermore, it receives $\boldsymbol{\zeta}_i$, an estimate about the maximum distance from the virtual center, out of the communication network assuming a connection to the ith vehicle. Let

$$\boldsymbol{d}_{\max} = \begin{bmatrix} d_1 & d_2 & d_3 \end{bmatrix}^T, \quad \boldsymbol{d}_{\min} = \begin{bmatrix} d_4 & d_5 & d_6 \end{bmatrix}^T, \tag{4.9}$$

denote this information available to the reference controller. Its objective is to generate a controlled reference frame state $\boldsymbol{\xi}^r_{\text{contr}}$ that closely follows the desired information state $\boldsymbol{\xi}^r$ but also considers the dynamic limitations of the vehicles through the available $\boldsymbol{\zeta}_i$ feedback from the formation.

The current maximal required accelerations in the formation are calculated, following Appendix B, in the inertial frame as

$$\boldsymbol{a}_{\max_1} = \ddot{\boldsymbol{r}}_{\boldsymbol{c}} + \dot{\boldsymbol{\omega}} \times \boldsymbol{d}_{\max} + \boldsymbol{\omega} \times (\boldsymbol{\omega} \times \boldsymbol{d}_{\max}) \tag{4.10}$$

$$\boldsymbol{a}_{\max_2} = \ddot{\boldsymbol{r}}_{\boldsymbol{c}} + \dot{\boldsymbol{\omega}} \times \boldsymbol{d}_{\min} + \boldsymbol{\omega} \times (\boldsymbol{\omega} \times \boldsymbol{d}_{\min}) \tag{4.11}$$

with the angular velocity vector

$$\begin{aligned}\boldsymbol{\omega} &= \begin{bmatrix}\cos\gamma & -\sin\gamma & 0\\ \sin\gamma & \cos\gamma & 0\\ 0 & 0 & 1\end{bmatrix}\begin{bmatrix}\cos\beta & 0 & \sin\beta\\ 0 & 1 & 0\\ -\sin\beta & 0 & \cos\beta\end{bmatrix}\begin{bmatrix}\dot\alpha\\0\\0\end{bmatrix} + \begin{bmatrix}\cos\gamma & -\sin\gamma & 0\\ \sin\gamma & \cos\gamma & 0\\ 0 & 0 & 1\end{bmatrix}\begin{bmatrix}0\\\dot\beta\\0\end{bmatrix} + \begin{bmatrix}0\\0\\\dot\gamma\end{bmatrix}\\ &= \begin{bmatrix}\cos\gamma\cos\beta & -\sin\gamma & 0\\ \sin\gamma\cos\beta & \cos\gamma & 0\\ -\sin\beta & 0 & 1\end{bmatrix}\begin{bmatrix}\dot\alpha\\\dot\beta\\\dot\gamma\end{bmatrix} = {}^W\boldsymbol{R}_E\,\dot{\boldsymbol{\delta}}\end{aligned}. \tag{4.12}$$

It is derived from the three Euler angles time derivatives $\dot{\boldsymbol{\delta}}$ with respect to the inertial system transformed through the rotation matrix ${}^W\boldsymbol{R}_E$. The angular acceleration vector

$$\begin{aligned}\dot{\boldsymbol{\omega}} &= {}^W\boldsymbol{R}_E\,\ddot{\boldsymbol{\delta}} + {}^W\dot{\boldsymbol{R}}_E\,\dot{\boldsymbol{\delta}}\\ &= \begin{bmatrix}\ddot\alpha\cos\beta\cos\gamma - \dot\alpha\dot\beta\sin\beta\cos\gamma - \dot\alpha\cos\beta\dot\gamma\sin\gamma - \ddot\beta\sin\gamma - \dot\beta\dot\gamma\cos\gamma\\ \ddot\alpha\cos\beta\sin\gamma - \dot\alpha\dot\beta\sin\beta\sin\gamma + \dot\alpha\cos\beta\dot\gamma\cos\gamma + \ddot\beta\cos\gamma - \dot\beta\dot\gamma\sin\gamma\\ -\cos\beta\dot\beta\dot\alpha - \sin\beta\ddot\alpha + \ddot\gamma\end{bmatrix}\end{aligned} \tag{4.13}$$

is the derivative of (4.12). The acceleration constraints can now be expressed in the inequalities

$$\boldsymbol{a}_{\text{quadneg}} \overset{!}{\leq} \boldsymbol{a}_{\max_1} \overset{!}{\leq} \boldsymbol{a}_{\text{quadpos}} \tag{4.14}$$

$$\boldsymbol{a}_{\text{quadneg}} \overset{!}{\leq} \boldsymbol{a}_{\max_2} \overset{!}{\leq} \boldsymbol{a}_{\text{quadpos}}. \tag{4.15}$$

If one of the two maximum accelerations exceeds the physical limits of the quadrotor from (3.65) and (3.66), position errors will occur. Therefore, the task of the reference controller is to limit the desired state of the reference frame $\boldsymbol{\xi}^r$ in such a way that (4.14) and (4.15) are fulfilled. Particularly problematic is that there exists additional DOF through the translational $\ddot{\boldsymbol{r}}_c$ and rotational $\ddot{\boldsymbol{\delta}}$ components of the reference frame acceleration $\ddot{\boldsymbol{\xi}}^r$. Moreover, the relationship between $\ddot{\boldsymbol{\xi}}^r$ and the maximum accelerations is nonlinear because of the coordinate transformation of the angular velocity vector $\boldsymbol{\omega}$ in (4.12).

The idea is now to use the additional DOF in the form of a correction vector $\boldsymbol{c}_a \in \mathcal{R}^6$ which is only active if the required acceleration exceeds the physical limits. The correction is added to $\ddot{\boldsymbol{\xi}}^r$, the desired reference frame acceleration. As a result, the controlled reference state $\boldsymbol{\xi}^r_{\text{contr}}$ deviates from the desired value $\boldsymbol{\xi}^r$ and has to be readjusted. Consequently, an additional PD controller is added. The complete dynamics of the controlled reference frame are

$$\ddot{\boldsymbol{\xi}}^r_{\text{contr}} = \ddot{\boldsymbol{\xi}}^r + k^r_d\left(\dot{\boldsymbol{\xi}}^r - \dot{\boldsymbol{\xi}}^r_{\text{contr}}\right) + k^r_p\left(\dot{\boldsymbol{\xi}}^r - \dot{\boldsymbol{\xi}}^r_{\text{contr}}\right) + \boldsymbol{c}_a. \tag{4.16}$$

A schematic is provided in Figure 4.7. If the inequalities (4.14)-(4.15) are fulfilled the correction vector $\boldsymbol{c}_a$ becomes zero. The result is a PD controller with asymptotic stable error dynamics

$$\boldsymbol{c}_a = 0: \qquad \left(\ddot{\boldsymbol{\xi}}^r - \ddot{\boldsymbol{\xi}}^r_{\text{contr}}\right) + k_d\left(\dot{\boldsymbol{\xi}}^r - \dot{\boldsymbol{\xi}}^r_{\text{contr}}\right) + k_p\left(\boldsymbol{\xi}^r - \boldsymbol{\xi}^r_{\text{contr}}\right) = 0 \tag{4.17}$$

$$\Leftrightarrow \qquad \ddot{\boldsymbol{e}}_\xi + k^r_d\dot{\boldsymbol{e}}_\xi + k^r_p\boldsymbol{e}_\xi = 0 \tag{4.18}$$

where $\boldsymbol{e}_\xi = \boldsymbol{\xi}^r - \boldsymbol{\xi}^r_{\text{contr}}$.

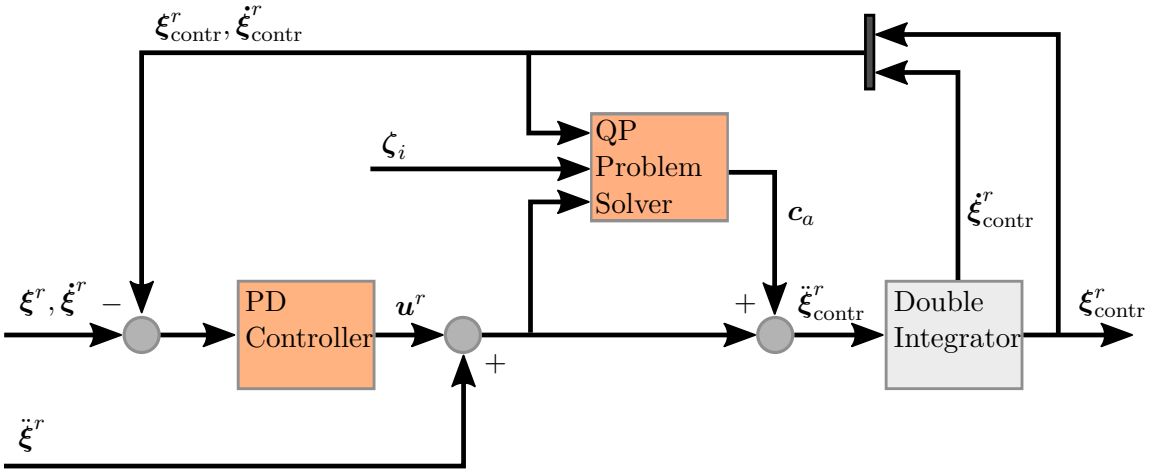


Figure 4.7: Reference controller structure.

In the case that $\boldsymbol{a}_{\max_1}$ or $\boldsymbol{a}_{\max_2}$ exceeds its limits, the vector $\boldsymbol{c}_a$ is calculated by solving a Quadratic Programming (QP) problem. It minimizes the vector norm of $\boldsymbol{c}_a$ subject to the constraints (4.14) and (4.15) and gives, added with the PD controller output $\boldsymbol{u}^r$, acceptable values for the reference frame acceleration $\ddot{\boldsymbol{\xi}}^r_{\text{contr}}$. In other words, the minimum value of the correction vector is calculated that keeps the maximum acceleration within the quadrotor's physical limits.

Expressed in mathematical terms, the QP problem is

$$\min_{\boldsymbol{c}_a} \quad \frac{1}{2}\boldsymbol{c}_a^T \boldsymbol{H}_a \boldsymbol{c}_a, \quad \boldsymbol{H}_a = \begin{bmatrix} h_1 & & 0 \\ & \ddots & \\ 0 & & h_6 \end{bmatrix} \in \mathcal{R}^{6\times 6}, \tag{4.19}$$

$$\text{s.t.} \quad \boldsymbol{G}_a \boldsymbol{c}_a \leq \boldsymbol{h}_a \tag{4.20}$$

where $\boldsymbol{H}_a$ is a weighting matrix and $\boldsymbol{G}_a$ as well as $\boldsymbol{h}_a$ are derived from (4.14)-(4.15) in the following steps. In the inequality

$$\boldsymbol{a}_{\max_1} = \ddot{\boldsymbol{r}}_c + \dot{\boldsymbol{\omega}} \times \boldsymbol{d}_{\max} + \boldsymbol{\omega} \times (\boldsymbol{\omega} \times \boldsymbol{d}_{\max}) \leq \boldsymbol{a}_{\text{quadpos}} \tag{4.21}$$

$$\Leftrightarrow \qquad \ddot{\boldsymbol{r}}_c + \dot{\boldsymbol{\omega}} \times \boldsymbol{d}_{\max} \leq \boldsymbol{a}_{\text{quadpos}} - \boldsymbol{\omega} \times (\boldsymbol{\omega} \times \boldsymbol{d}_{\max}) \tag{4.22}$$

the cross product $\dot{\boldsymbol{\omega}} \times \boldsymbol{d}_{\max}$ is rewritten as

$$\begin{aligned} \dot{\boldsymbol{\omega}} \times \boldsymbol{d}_{\max} &= \left({}^W\boldsymbol{R}_E\, \ddot{\boldsymbol{\delta}} + {}^W\dot{\boldsymbol{R}}_E\, \dot{\boldsymbol{\delta}}\right) \times \boldsymbol{d}_{\max} \\ &= \underbrace{\left({}^W\boldsymbol{R}_E\, \ddot{\boldsymbol{\delta}} \times \boldsymbol{d}_{\max}\right)}_{\boldsymbol{G}_1 \ddot{\boldsymbol{\delta}}} + \underbrace{\left({}^W\dot{\boldsymbol{R}}_E\, \dot{\boldsymbol{\delta}} \times \boldsymbol{d}_{\max}\right)}_{\boldsymbol{h}_1} \end{aligned} \tag{4.23}$$

with the matrix

$$\boldsymbol{G}_1 = \begin{bmatrix} d_3 \cos\beta \sin\gamma + d_2 \sin\beta & \cos\gamma d_3 & -d_2 \\ d_1 \sin\beta - d_3 \cos\beta \cos\gamma & d_3 \sin\gamma & d_1 \\ d_2 \cos\beta \cos\gamma - d_1 \cos\beta \sin\gamma & -d_2 \sin\gamma - d_1 \cos\gamma & 0 \end{bmatrix}. \tag{4.24}$$

Summarizing,

$$\ddot{\boldsymbol{r}}_c + \dot{\boldsymbol{\omega}} \times \boldsymbol{d}_{\max} \leq \boldsymbol{a}_{\text{quadpos}} - \boldsymbol{\omega} \times (\boldsymbol{\omega} \times \boldsymbol{d}_{\max}) \tag{4.25}$$

$$\Leftrightarrow \quad \ddot{\boldsymbol{r}}_c + \boldsymbol{G}_1 \ddot{\boldsymbol{\delta}} + \boldsymbol{h}_1 \leq \boldsymbol{a}_{\text{quadpos}} - \boldsymbol{\omega} \times (\boldsymbol{\omega} \times \boldsymbol{d}_{\max}) \tag{4.26}$$

$$\Leftrightarrow \quad \begin{bmatrix} \boldsymbol{I}_3 & \boldsymbol{G}_1 \end{bmatrix} \ddot{\boldsymbol{\xi}}^r_{\text{contr}} \leq \boldsymbol{a}_{\text{quadpos}} - \boldsymbol{\omega} \times (\boldsymbol{\omega} \times \boldsymbol{d}_{\max}) - \boldsymbol{h}_1. \tag{4.27}$$

In the second step, by replacing $\ddot{\boldsymbol{\xi}}^r_{\text{contr}} = \boldsymbol{u}^r + \boldsymbol{c}_a$ with its components, (4.27) becomes

$$\begin{bmatrix} \boldsymbol{I}_3 & \boldsymbol{G}_1 \end{bmatrix} (\boldsymbol{u}^r + \boldsymbol{c}_a) \leq \boldsymbol{a}_{\text{quadpos}} - \boldsymbol{\omega} \times (\boldsymbol{\omega} \times \boldsymbol{d}_{\max}) - \boldsymbol{h}_1 \tag{4.28}$$

$$\Leftrightarrow \quad \begin{bmatrix} \boldsymbol{I}_3 & \boldsymbol{G}_1 \end{bmatrix} \boldsymbol{c}_a \leq \boldsymbol{a}_{\text{quadpos}} - \boldsymbol{\omega} \times (\boldsymbol{\omega} \times \boldsymbol{d}_{\max}) - \boldsymbol{h}_1 - \begin{bmatrix} \boldsymbol{I}_3 & \boldsymbol{G}_1 \end{bmatrix} \boldsymbol{u}^r \tag{4.29}$$

$$\Leftrightarrow \quad \tilde{\boldsymbol{G}}_1 \boldsymbol{c}_a \leq \boldsymbol{a}_{\text{quadpos}} + \tilde{\boldsymbol{h}}_1 \tag{4.30}$$

with $\tilde{\boldsymbol{G}}_1 = [\boldsymbol{I}_3 \quad \boldsymbol{G}_1]$ and $\tilde{\boldsymbol{h}}_1 = -\boldsymbol{\omega} \times (\boldsymbol{\omega} \times \boldsymbol{d}_{\max}) - \boldsymbol{h}_1 - [\boldsymbol{I}_3 \quad \boldsymbol{G}_1]\, \boldsymbol{u}^r$. For the negative acceleration limit $\boldsymbol{a}_{\text{quadneg}}$, a similar transformation results in

$$-\tilde{\boldsymbol{G}}_1 \boldsymbol{c}_a \leq -\boldsymbol{a}_{\text{quadneg}} - \tilde{\boldsymbol{h}}_1. \tag{4.31}$$

Likewise, using the maximum negative distance from (4.15) gives

$$\tilde{\boldsymbol{G}}_2 \boldsymbol{c}_a \leq \boldsymbol{a}_{\text{quadpos}} + \tilde{\boldsymbol{h}}_2 \tag{4.32}$$

$$-\tilde{\boldsymbol{G}}_2 \boldsymbol{c}_a \leq -\boldsymbol{a}_{\text{quadneg}} - \tilde{\boldsymbol{h}}_2 \tag{4.33}$$

with

$$\dot{\boldsymbol{\omega}} \times \boldsymbol{d}_{\min} = \left({}^W\boldsymbol{R}_E\, \ddot{\boldsymbol{\delta}} + {}^W\dot{\boldsymbol{R}}_E\, \dot{\boldsymbol{\delta}} \right) \times \boldsymbol{d}_{\min} = \boldsymbol{G}_2\, \ddot{\boldsymbol{\delta}} + \boldsymbol{h}_2 \tag{4.34}$$

$$\tilde{\boldsymbol{h}}_2 = -\boldsymbol{\omega} \times (\boldsymbol{\omega} \times \boldsymbol{d}_{\min}) - \boldsymbol{h}_2 - \begin{bmatrix} \boldsymbol{I}_3 & \boldsymbol{G}_2 \end{bmatrix} \boldsymbol{u}^r \tag{4.35}$$

$$\tilde{\boldsymbol{G}}_2 = \begin{bmatrix} \boldsymbol{I}_3 & \boldsymbol{G}_2 \end{bmatrix}. \tag{4.36}$$

Combining the four inequalities into a single inequality gives

$$\boldsymbol{G}_a \boldsymbol{c}_a \leq \boldsymbol{h}_a \tag{4.37}$$

$$\Leftrightarrow \quad \begin{bmatrix} \tilde{\boldsymbol{G}}_1 \\ -\tilde{\boldsymbol{G}}_1 \\ \tilde{\boldsymbol{G}}_2 \\ -\tilde{\boldsymbol{G}}_2 \end{bmatrix} \boldsymbol{c}_a \leq \begin{bmatrix} \boldsymbol{a}_{\text{quadpos}} \\ -\boldsymbol{a}_{\text{quadneg}} \\ \boldsymbol{a}_{\text{quadpos}} \\ -\boldsymbol{a}_{\text{quadneg}} \end{bmatrix} + \begin{bmatrix} \tilde{\boldsymbol{h}}_1 \\ -\tilde{\boldsymbol{h}}_1 \\ \tilde{\boldsymbol{h}}_2 \\ -\tilde{\boldsymbol{h}}_2 \end{bmatrix} \tag{4.38}$$

which are the linear inequality constraints for the QP problem. As the weighting matrix only consists of elements greater than zero, it is positive definite and therefore the optimization problem is convex. In addition, it is of low dimensionality with $\boldsymbol{c}_a \in \mathcal{R}^6$. A global optimum can be found with a standard MATLAB function like `quadprog`.

Chapter 5

Simulation Results

In this chapter, the previously proposed formation control framework is gradually simulated and evaluated using the software MATLAB® with Simulink®. Starting with the individual quadrotor modeling and control, the system is extended for multiple vehicles including a communication network. Subsequently, the consensus modules are simulated. Finally, the complete formation control structure is simulated with and without active reference controller. Both solutions are compared to each other and quantitatively assessed.

5.1 Quadrotor Control

The vehicle model used in the simulation is based upon (3.29) to (3.34). As for the physical constants, they are extracted from the Parrot AR Drone 2.0 specifications and the aerodynamical coefficients taken from [Rud+14] and stated in Table 5.2. At first, the altitude controller is tuned using the step response from Figure 5.1. High controller gains are set to ensure fast dynamics. Similarly, the angle controller is tuned. Therefore, the dynamics of the inner control loop are considerably fast. In a second step, the position controller gains are chosen to ensure stable and well-damped step responses as seen in Figure 5.2 and 5.3. The simulation parameters are summed up in Table 5.2.

The results show a model which fulfills the requirements formulated in the beginning of Chapter 3. It is stable as well as fast in positioning. Furthermore, the physical limits stated in Chapter 3.5 are validated in the simulation with the angle and thrust limits from Table 5.2. Numerical results can be found in Table 5.1.

Table 5.1: Quadrotor acceleration limits simulation results.

Direction in $\mathcal{F}_W$	Maximum Acceleration [$\mathrm{m\,s^{-2}}$]
x (unidirectional)	0.5
y (unidirectional)	0.5
z (positive)	1.8
z (negative)	9.81

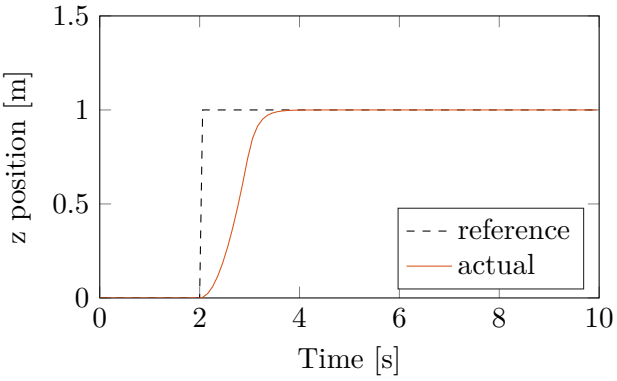


Figure 5.1: Step response of the altitude controller.

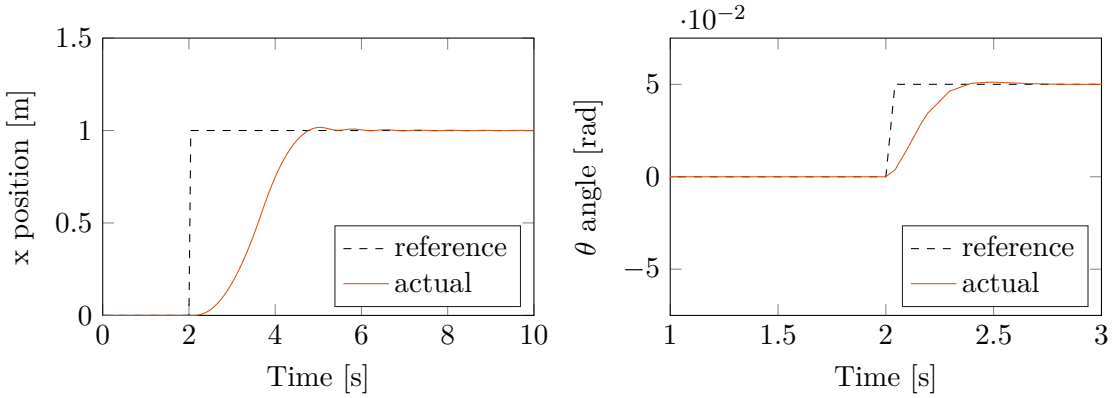


Figure 5.2: Step response of cascaded controller in x-axis direction.

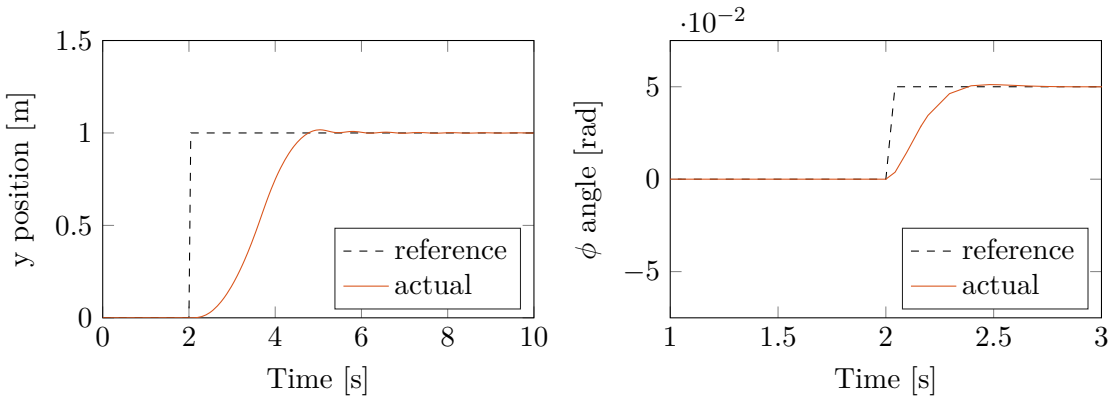


Figure 5.3: Step response of cascaded controller in y-axis direction.

Table 5.2: Quadrotor simulation parameters.

System	Parameter	Value
Quadrotor physical constants	m	0.45 kg
	g	9.81 kg m s^{-2}
	$k_{\text{aer,x}}$	0.1
	$k_{\text{aer,y}}$	0.1
	$k_{\text{aer,z}}$	0.2
	$\theta_{\max} = -\theta_{\min}$	0.05 rad
	$\phi_{\max} = -\phi_{\min}$	0.05 rad
	$f_{\text{t,max}}$	5.2245 N
Quadrotor altitude and attitude controller	k_{1z}	100
	k_{2z}	20
	$k_{1\psi}$	100
	$k_{2\psi}$	15
	$k_{1\theta}$	100
	$k_{2\theta}$	15
	$k_{1\phi}$	100
	$k_{2\phi}$	15
Quadrotor position controller	k_{1x}	1.2
	k_{2x}	1
	k_{1y}	1.2
	k_{2y}	1

5.2 Formation Control Setup

5.2.1 Communication Network

To highlight the flexibility of the formation control framework, a random directed spanning tree with n nodes each are created at the start of a simulation. This is achieved by using the Erdõs-Rényi algorithm to create random graphs. Each possible link in the graph is created with a probability of sixty percent [Wes01]. From the graph, a minimum spanning tree is extracted and its corresponding adjacency matrix calculated. The connection to the VL are specified separately and multiple connections are possible. For the following simulations, the number of vehicles n equals six and the generated minimum spanning tree $\mathcal{G}$ in Figure 5.4 is used. The VL has a connection only to vehicle one.

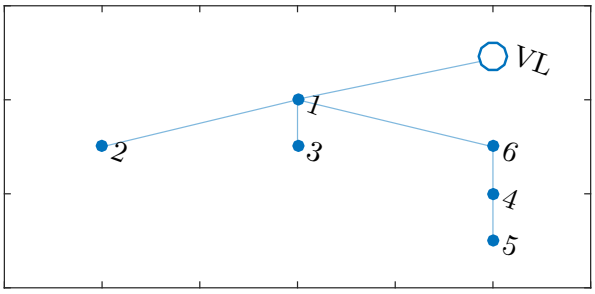


Figure 5.4: Minimum spanning tree for $\mathcal{G}$ with VL connection.

5.2.2 Formation Shape

The formation shape is given through the desired relative positions $\boldsymbol{r}^d_{\text{rel,i}}$ of the quadrotors with reference to the virtual reference frame. To set them, parametric equations are used with the flexibility of implementing any number of vehicles with minimal effort. A line on the y-axis is defined as

$$\boldsymbol{x}_l(s) = \begin{bmatrix} 0 \\ s \\ 0 \end{bmatrix}, \qquad s_{\min} < s < s_{\max}. \tag{5.1}$$

The n vehicles are uniformly distributed on the curve with

$$\boldsymbol{r}^d_{\text{rel,i}} = \boldsymbol{x}_l \left(s_{\min} + (i-1) \frac{s_{\max} - s_{\min}}{n-1} \right), \quad i = \{1, \ldots, n\}. \tag{5.2}$$

More complicated parametrizations can be easily realized. For the rest of the thesis, the vehicles are in the line formation $\boldsymbol{x}_l$ with the parameters $s_{\min} = -2\,\text{m}$ and $s_{\max} = 2\,\text{m}$.

5.2.3 Trajectory of the Reference Frame

The reference state $\boldsymbol{\xi}^r$ with the position of the virtual center and the three euler angles can be set through inputs into its second time derivative

$$\ddot{\boldsymbol{\xi}}^r(t) = \begin{bmatrix} \ddot{x}^r_c & \ddot{y}^r_c & \ddot{z}^r_c & \ddot{\alpha}^r & \ddot{\beta}^r & \ddot{\gamma}^r \end{bmatrix}^T \tag{5.3}$$

with the the initial conditions of

$$\boldsymbol{\xi}^r(0) = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T, \quad \dot{\boldsymbol{\xi}}^r(0) = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T. \tag{5.4}$$

From the following on, the evolution of this frame will fixed to

$$\ddot{x}^r_c(t) = \begin{cases} 0.5\,\text{m}\,\text{s}^{-2} & 0\,\text{s} \le t \le 2\,\text{s} \\ 0\,\text{m}\,\text{s}^{-2} & \text{otherwise} \end{cases} \tag{5.5}$$

$$\ddot{y}^r_c(t) = 0\,\text{m}\,\text{s}^{-2} \tag{5.6}$$

$$\ddot{z}^r_c(t) = \begin{cases} 1.5\,\text{m}\,\text{s}^{-2} & 0\,\text{s} \le t \le 2\,\text{s} \\ -1.5\,\text{m}\,\text{s}^{-2} & 8\,\text{s} \le t \le 10\,\text{s} \\ 0\,\text{m}\,\text{s}^{-2} & \text{otherwise} \end{cases} \tag{5.7}$$

$$\ddot{\alpha}^r(t) = \begin{cases} \pi/2\,\text{rad}\,\text{s}^{-2} & 4\,\text{s} \le t \le 5\,\text{s} \\ -\pi/2\,\text{rad}\,\text{s}^{-2} & 5\,\text{s} \le t \le 6\,\text{s} \\ 0\,\text{rad}\,\text{s}^{-2} & \text{otherwise} \end{cases} \tag{5.8}$$

$$\ddot{\beta}^r(t) = 0\,\text{rad}\,\text{s}^{-2} \tag{5.9}$$

$$\ddot{\gamma}^r(t) = 0\,\text{rad}\,\text{s}^{-2}. \tag{5.10}$$

It consists of a fast rising motion plus simultaneously turning the reference frame 90° while also maintaining a forward movement in positive x-direction. The result is shown in Figure 5.5. An important aspect is that the VL is rising with a velocity greater then the terminal possible velocity of the quadrotor in positive z-direction. Therefore, the effectiveness of the proposed formation feedback can be proved. The simulation time t_{sim} is twelve seconds.

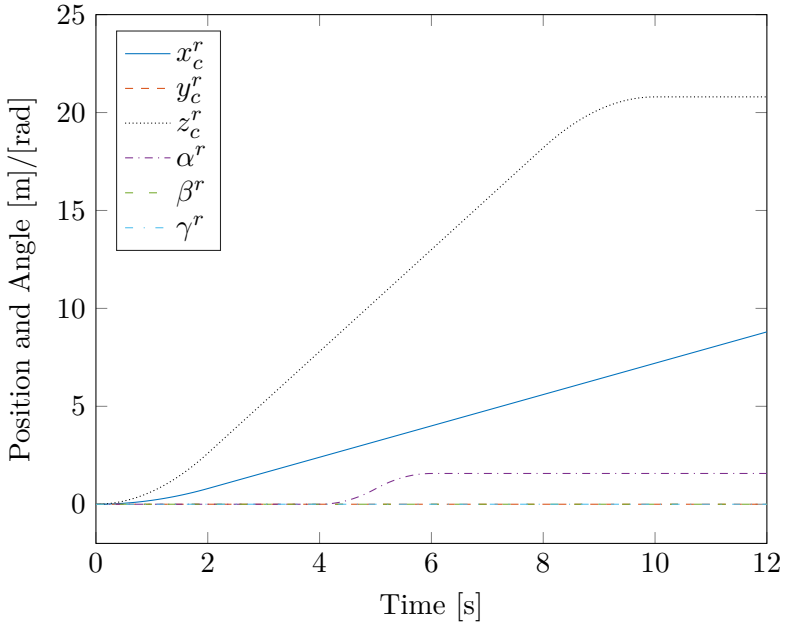


Figure 5.5: Evolution of the virtual reference frame state $\boldsymbol{\xi}^r$.

5.3 Reference Frame Consensus Module

The consensus algorithm for the virtual reference frame drives the information states of the individual vehicles $\boldsymbol{\xi}_i$ to the reference $\boldsymbol{\xi}^r_{\text{contr}}$. The reference correction is assumed inactive, therefore $\boldsymbol{\xi}^r_{\text{contr}} = \boldsymbol{\xi}^r$. Exemplary, the evolvement of the α_i elements of $\boldsymbol{\xi}_i$ from each vehicle is plotted in Figure 5.6. The reference $a^r(t)$ value is adopted from the section before. Interesting to note is the effect of different errors $e_{\alpha_i} = |\alpha^r(t) - \alpha_i(t)|$ which can be explained by looking at the current communication network $\mathcal{G}$. Let

$$J_\alpha = \int_{0\,\text{s}}^{12\,\text{s}} |\alpha^r(t) - \alpha_i^r(t)| \text{d}t, \quad i = \{1, \ldots, n\} \tag{5.11}$$

be a performance criteria. Figure 5.7 shows that the information state of the fifth quadrotor performs worst. The reason is that three other nodes lie between this vehicle and the VL. Therefore, information has first to be passed down until it eventually reaches quadrotor number five. Nevertheless, it is able to follow a time-variant reference. Equally, the errors of the other vehicles can be explained. The control parameter κ influences the convergence speed and is set to $\kappa = 100$ to ensure a fast consensus.

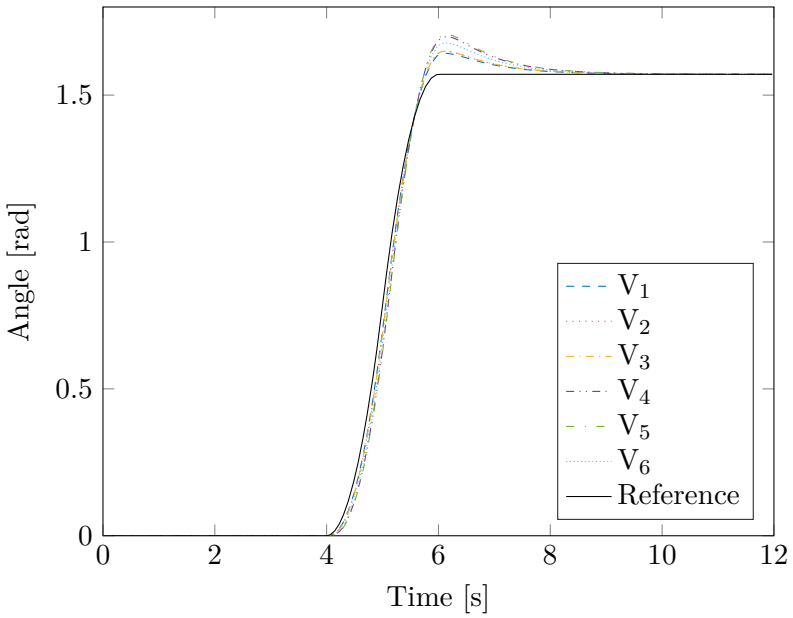


Figure 5.6: Consensus algorithm for α_i value of $\boldsymbol{\xi}_i$.

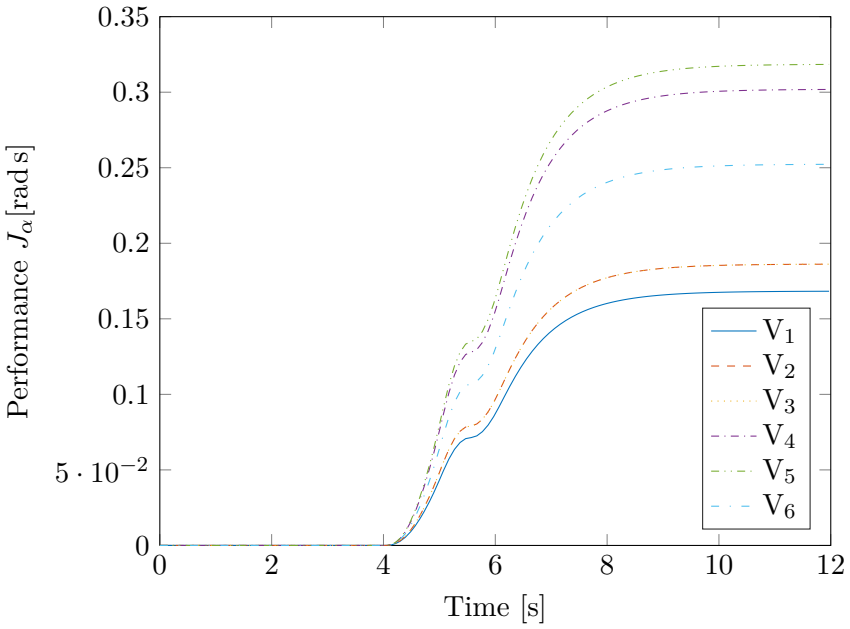


Figure 5.7: Performance of consensus algorithm for α_i value of $\boldsymbol{\xi}_i$.

5.4 Maximum Distance Consensus Module

To validate the algorithm introduced in Chapter 4.3, the line formation of six vehicles is simulated moving with the trajectory (5.5) - (5.10). Figure 5.8 shows the first three elements of the information state $\boldsymbol{\zeta}_3$ of vehicle three compared to the true global values. Notably, there is no difference between the actual global positive maximum distances and the information stored in the vehicle. This is due to the fact that communication delay and restrictions are not accounted for in the simulation. In a single time-step, the broadcast algorithm reaches consensus. However, it serves the purpose of simulating the possibilities of a distributed framework.

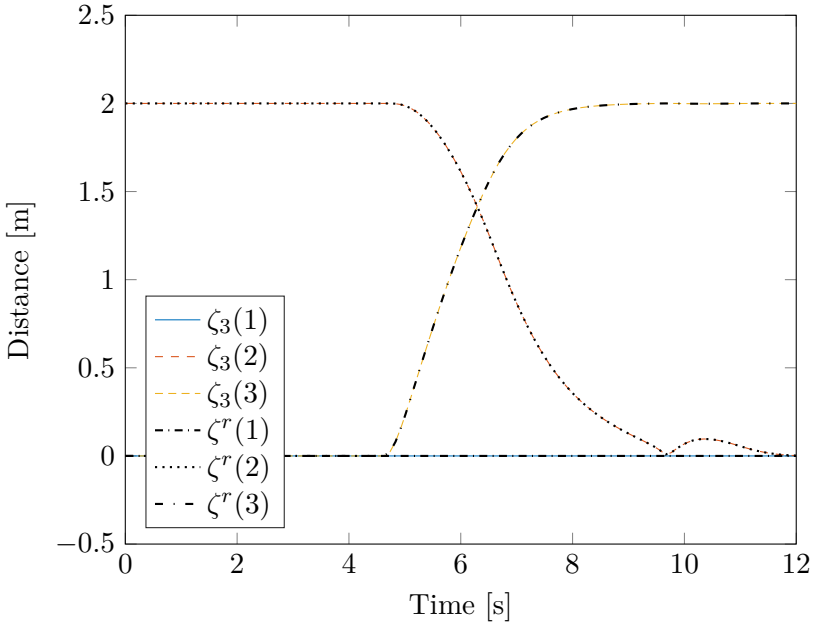


Figure 5.8: Maximum distance estimate $\boldsymbol{\zeta}_3$ of vehicle three and true values $\boldsymbol{\zeta}^r$.

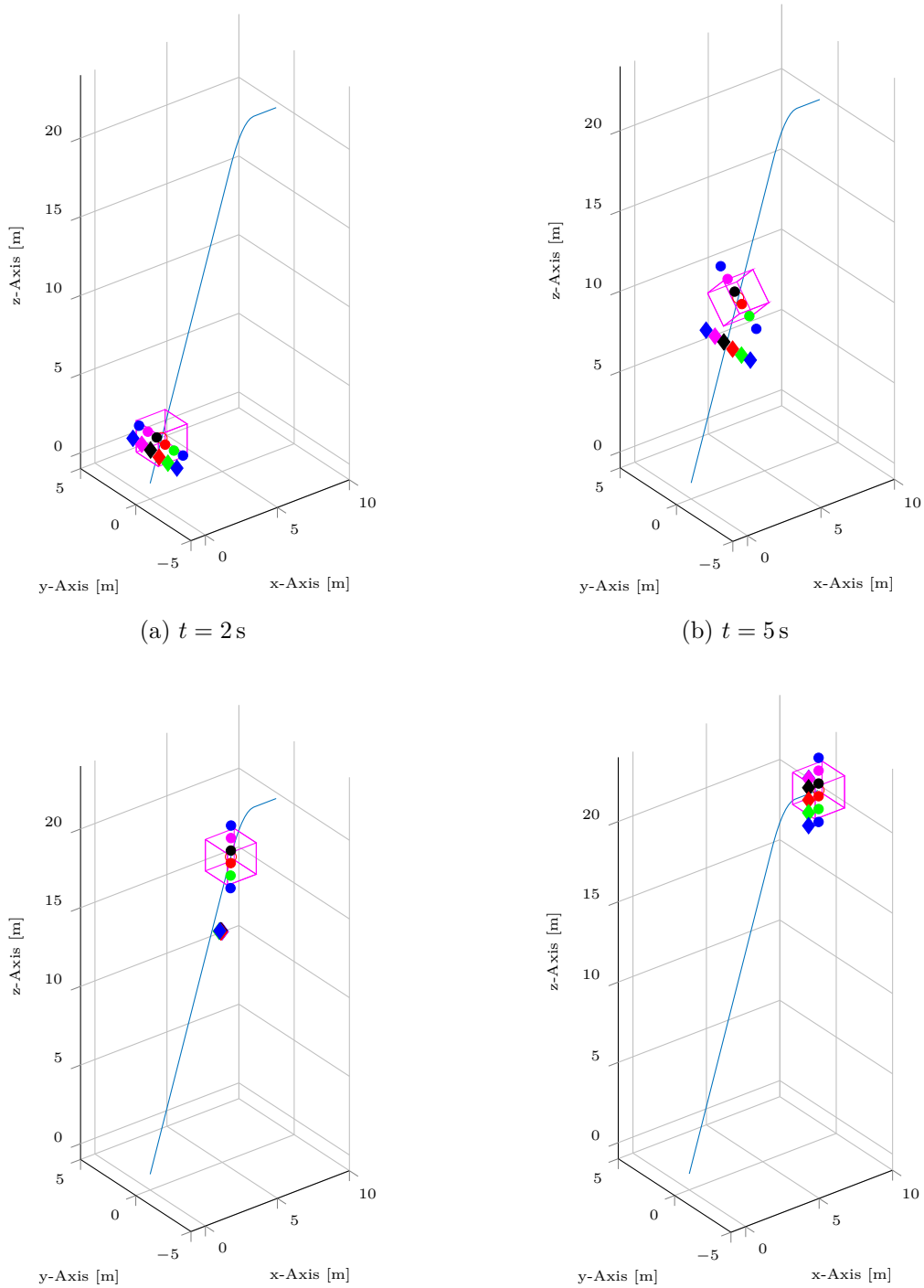
5.5 Formation Control without and with Reference Correction

5.5.1 Formation Control without Reference Correction

For the line formation moving along the proposed reference coordinate frame trajectory, the formation control framework is examined without reference correction $\boldsymbol{c}_a$. This means that the available information available to the vehicles is $\boldsymbol{\xi}^r_{\text{contr}} = \boldsymbol{\xi}^r$ as $\boldsymbol{c}_a = 0$. In Figure 5.9, the vehicle positions are drawn for four moments during the simulation time t_{sim} of twelve seconds. The round dots refer to the desired quadrotor positions and the diamonds to the actual ones. The cube represents the reference coordinate frame orientation as well as position. Its origin is moving along the trajectory visible in each picture. In the beginning, the vehicles are able to follow the rising motion of their desired positions as seen in Figure 5.9a. However, in Figure 5.9b the vehicles lack notably behind when the reference rotation sets in. This

ultimately leads to a complete formation loss as shown at $t = 8\,\mathrm{s}$ in Figure 5.9c. Only when the rising motion of the reference frame stops, the formation forms again in Figure 5.9d.

The maximum required accelerations calculated after (4.10) and (4.11) are shown in Figure 5.10 for the maximum positive distances $\boldsymbol{d}_{\max}$ and in Figure 5.11 for the maximum negative distances $\boldsymbol{d}_{\min}$. It can be seen that the rotation around the first coordinate axis of the reference frame (increasing alpha value) starting at $t = 4\,\mathrm{s}$ leads to high accelerations in the z- and y-coordinate direction which are beyond the physical limits of the quadrotors. Furthermore, the initial acceleration in z-direction of the reference frame results in a final velocity greater than the possible altitude gain of the simulated quadrotors due to air friction. Therefore, the quadrotors are already rising with terminal velocity when the rotation sets in it. They are only able to move in the horizontal plane and eventually converge close to a single point in Figure 5.9c.



(a) $t = 2\,\mathrm{s}$

(b) $t = 5\,\mathrm{s}$

(c) $t = 8\,\mathrm{s}$

(d) $t = 2\,\mathrm{s}$

Figure 5.9: Vehicle positions for four instants of time without reference correction.

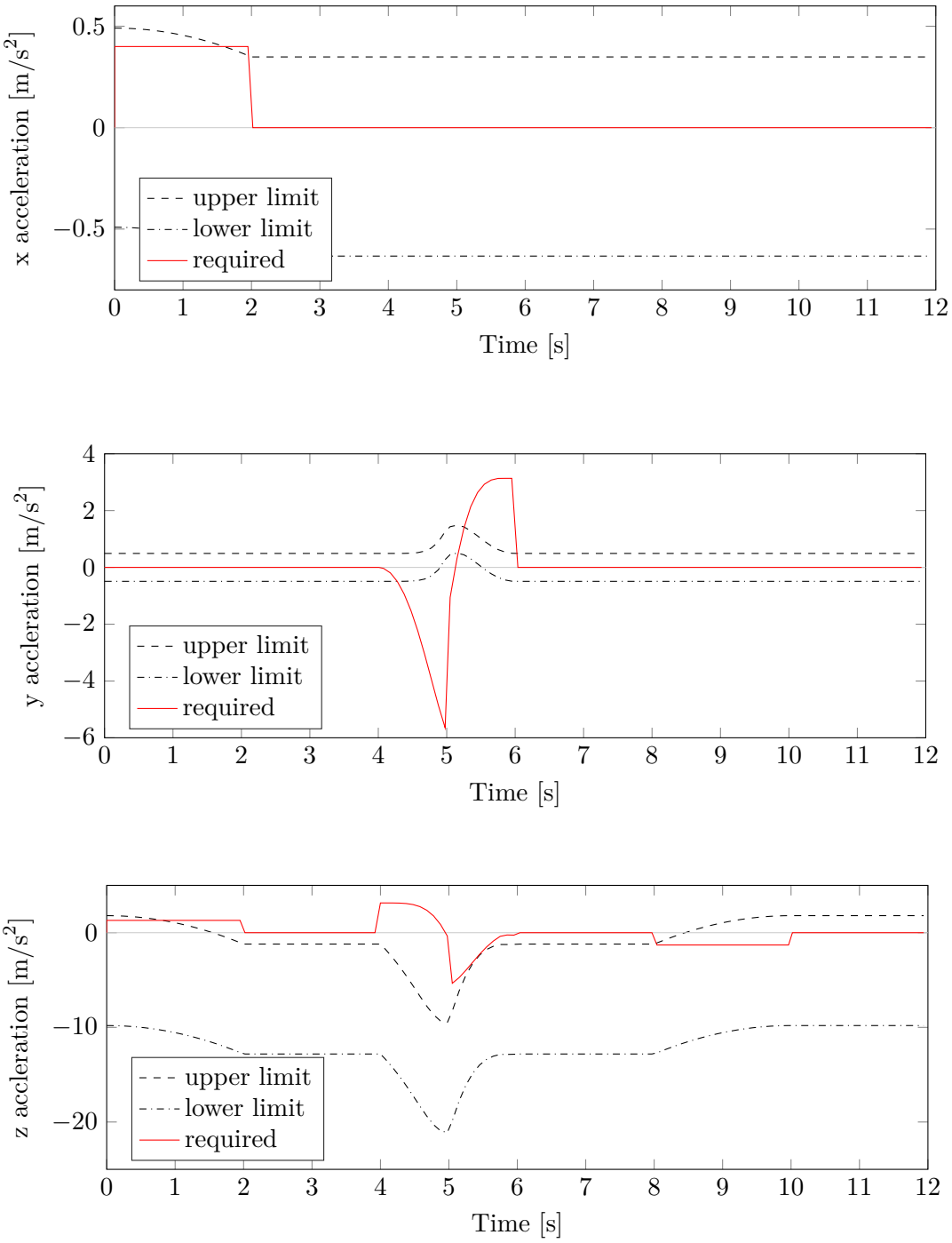


Figure 5.10: Required acceleration $\boldsymbol{a}_{\max_1}$ and quadrotor limits without reference correction.

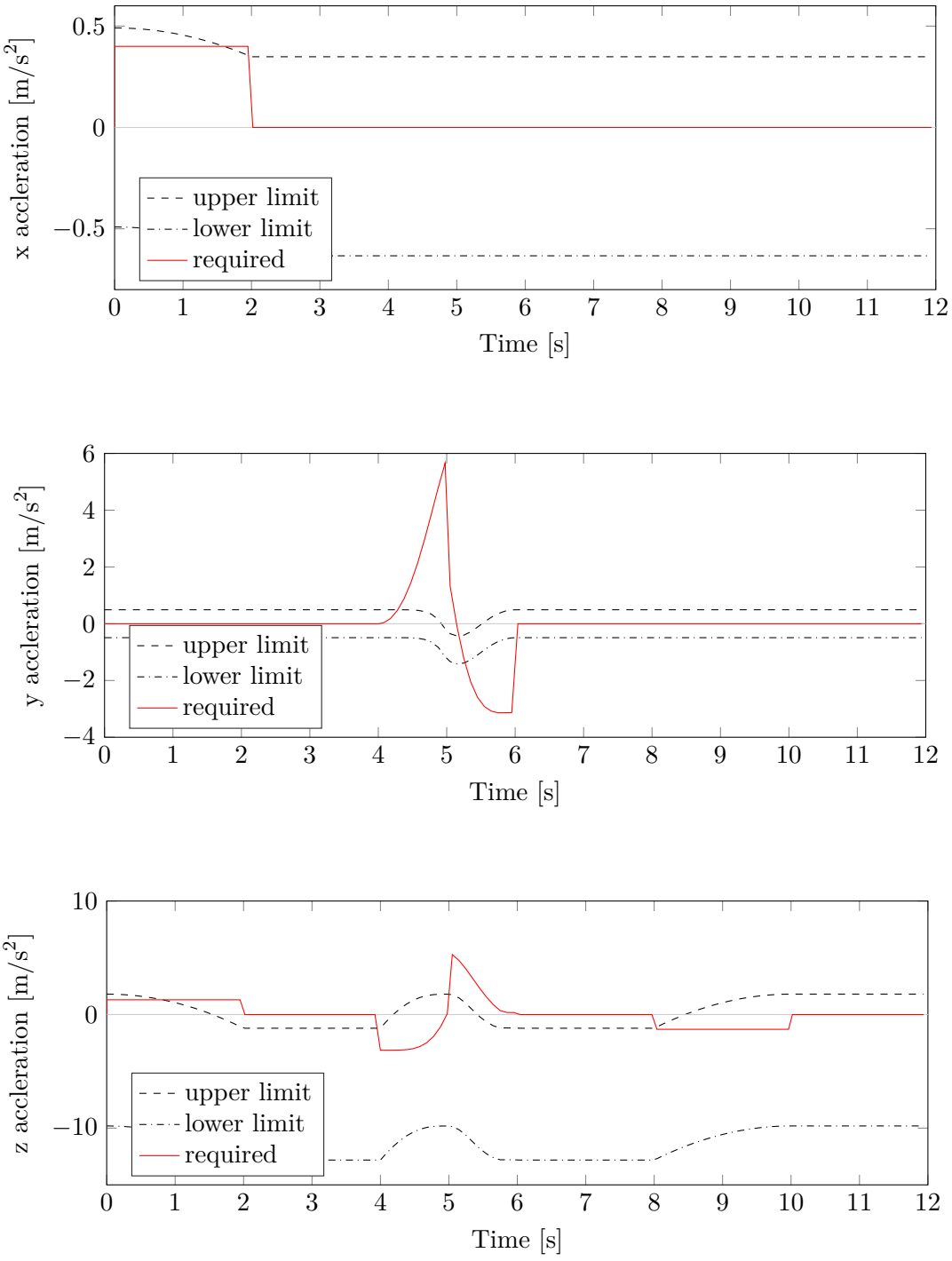


Figure 5.11: Required acceleration $\boldsymbol{a}_{\max_2}$ and quadrotor limits without reference correction.

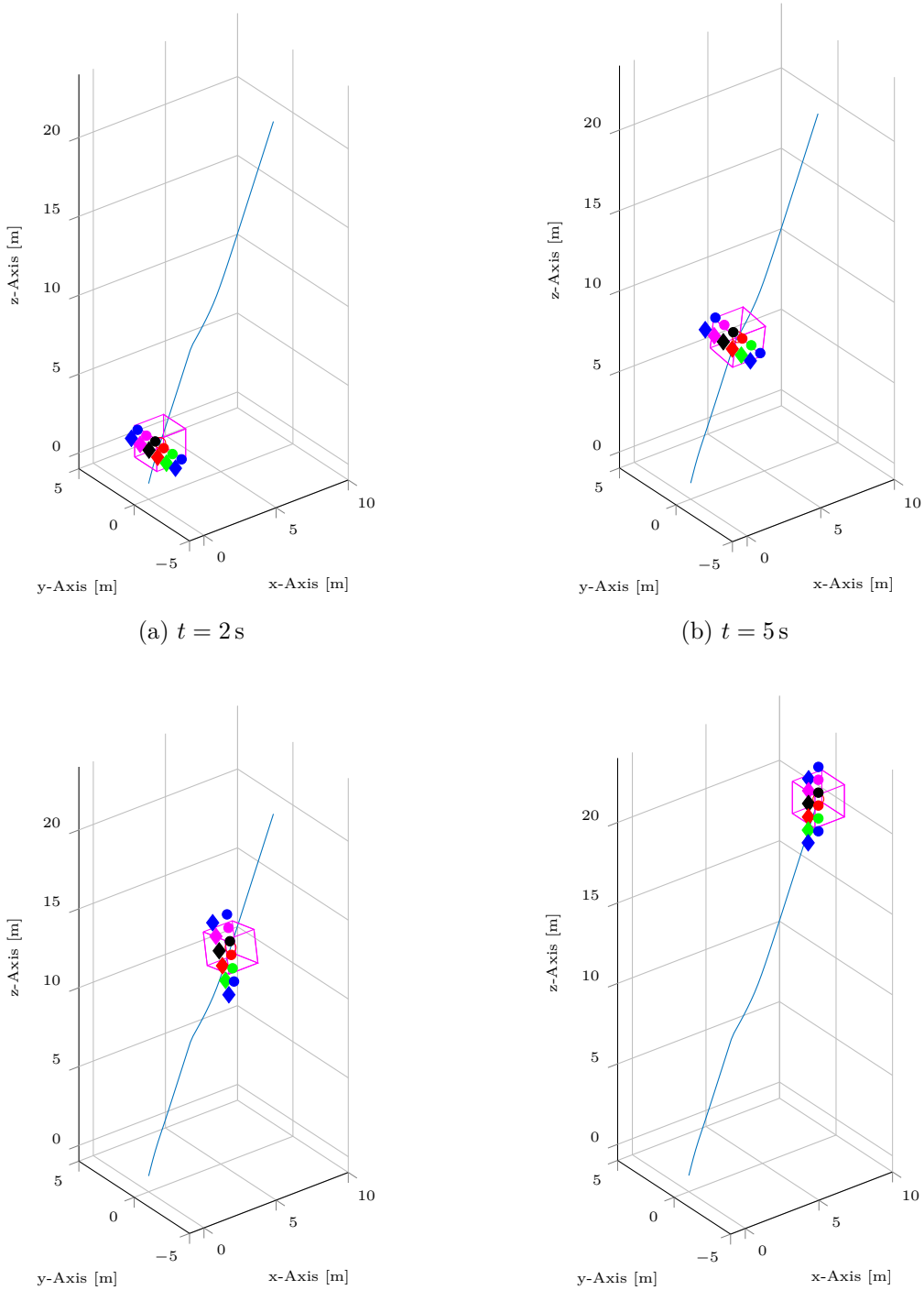
5.5.2 Formation Control with Reference Correction

In a next step, the reference correction is active during the simulation and the correction vector $\boldsymbol{c}_a$ can influence the reference trajectory. The control parameters for the reference controller are defined in Table 5.3. In Figure 5.12, the current state of the formation is shown as in Chapter 5.5.1 again for four moments in time. The rising motion of the reference frame starts slower which allows the vehicles to stay close to their desired positions in Figure 5.9a. Even when the rotation sets in, the quadrotors are able to follow as shown in Figure 5.12b and 5.12c. Throughout the twelve seconds simulation time, the formation shape is recognizable until the end in Figure 5.12d.

Figure 5.13 shows the accelerations for the most distant vehicles in positive coordinate directions and Figure 5.14 for the maximum negative distances. It can be clearly seen that the reference controller is able to limit the occurring maximum accelerations to the physical quadrotor constraints. During the whole simulation time, the required accelerations stay within the corridor spanned by (3.65) as the upper limit and (3.66) as the lower limit. Furthermore, the results for the controlled reference state $\boldsymbol{\xi}^r_{\text{contr}}$ are shown in Figure 5.15 for the translation and in Figure 5.16 for the rotational components and compared with the uncontrolled reference state $\boldsymbol{\xi}^r$. It becomes clear that the controller only limits the responsible components, namely $\ddot{z}^r$ and $\ddot{\alpha}$ in order to achieve the limited acceleration. The other states of $\boldsymbol{\xi}^r_{\text{contr}}$ remain unaffected. The reason is that the correction vector $\boldsymbol{c}_a$ is only non-zero for the $\ddot{\alpha}$ and $\ddot{z}^r$ correction, as displayed in Figure 5.17.

Table 5.3: Reference controller parameters.

Parameter	Value
h_1	1
h_2	1
h_3	1
h_4	1
h_5	1
h_6	1
k_d^r	15
k_p^r	100



(a) $t = 2\,\mathrm{s}$ (b) $t = 5\,\mathrm{s}$

(c) $t = 8\,\mathrm{s}$ (d) $t = 12\,\mathrm{s}$

Figure 5.12: Vehicle positions for four instants of time with reference correction.

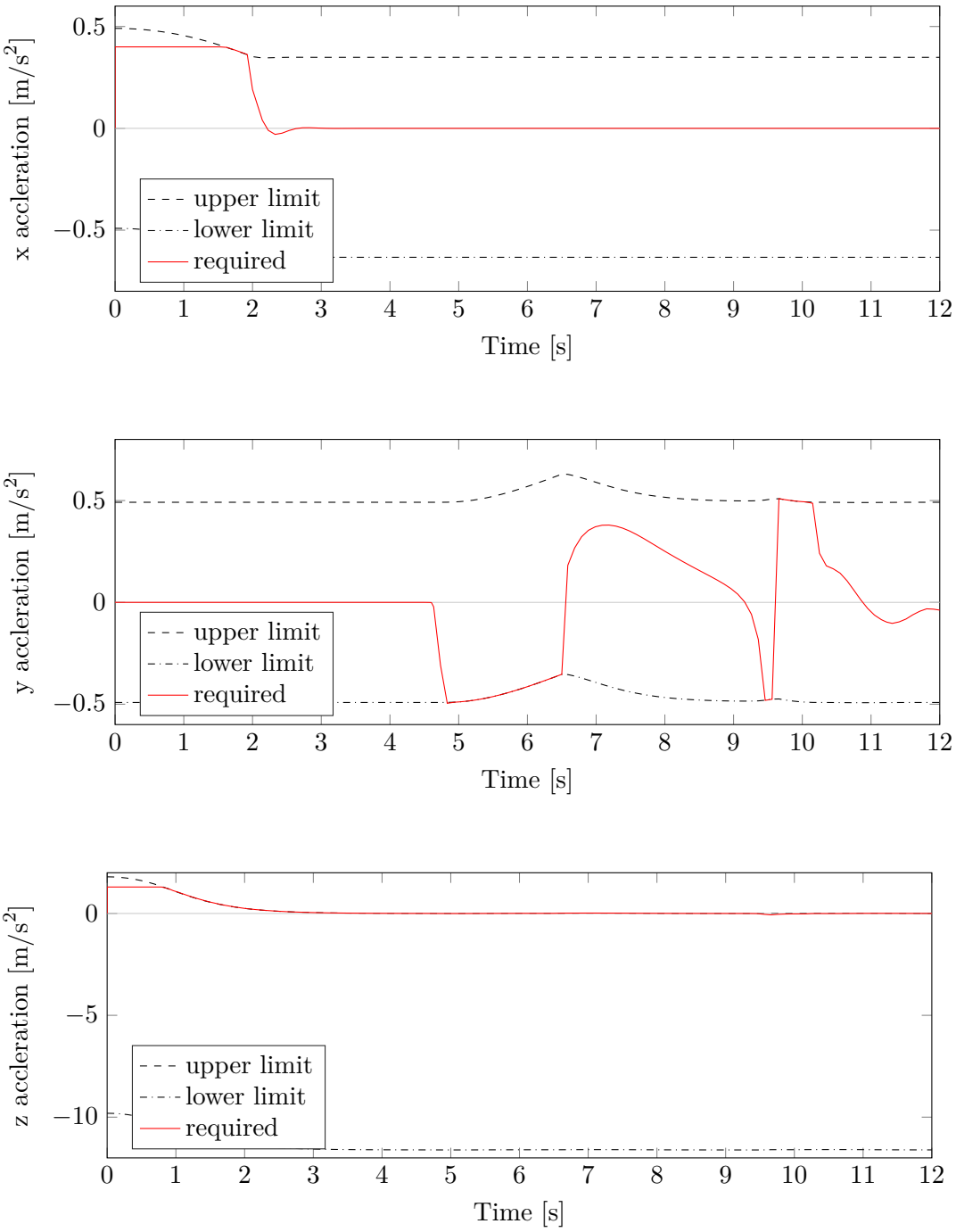


Figure 5.13: Required acceleration $\boldsymbol{a}_{\max_1}$ and quadrotor limits with reference correction.

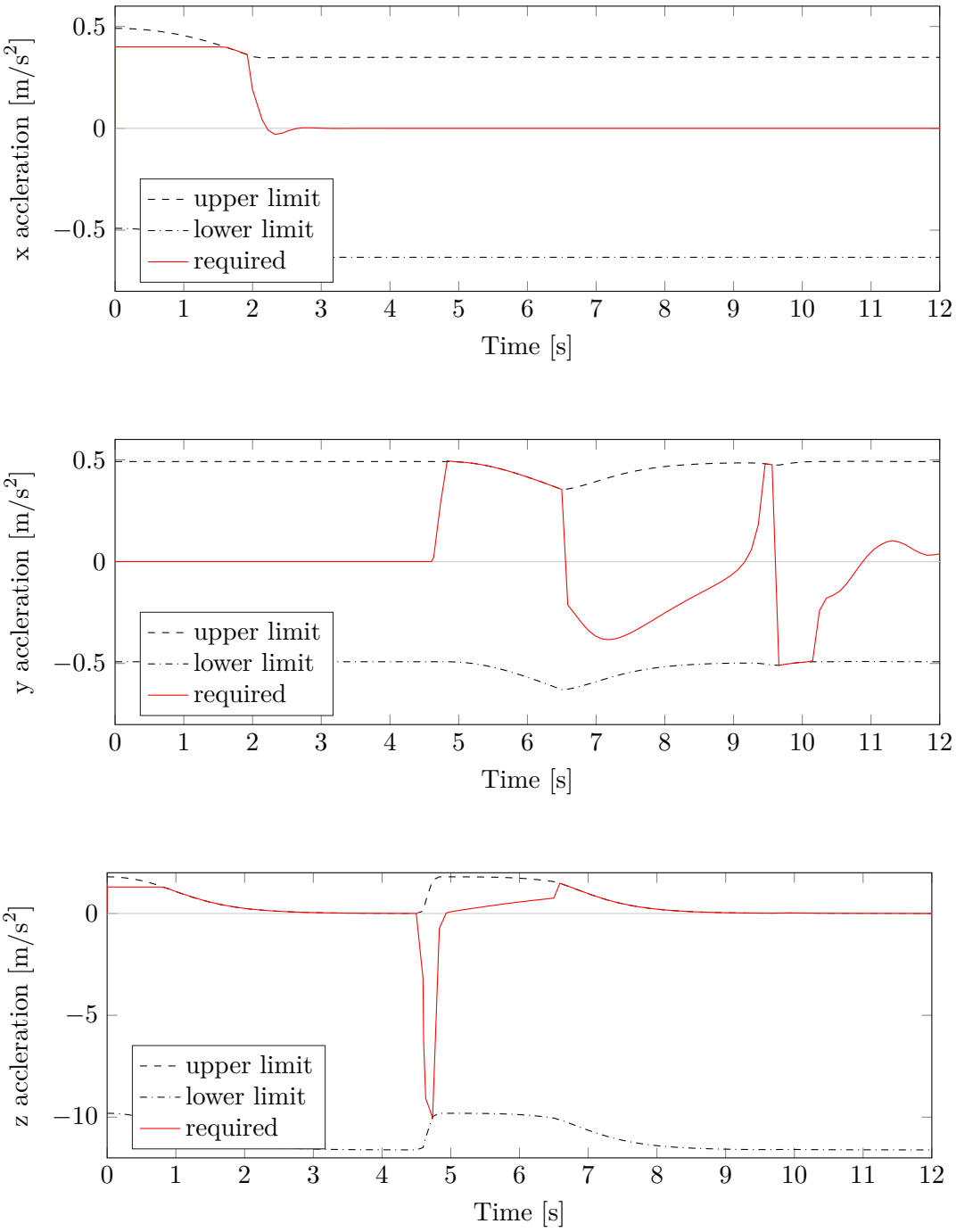


Figure 5.14: Required acceleration $\boldsymbol{a}_{\max_2}$ and quadrotor limits with reference correction.

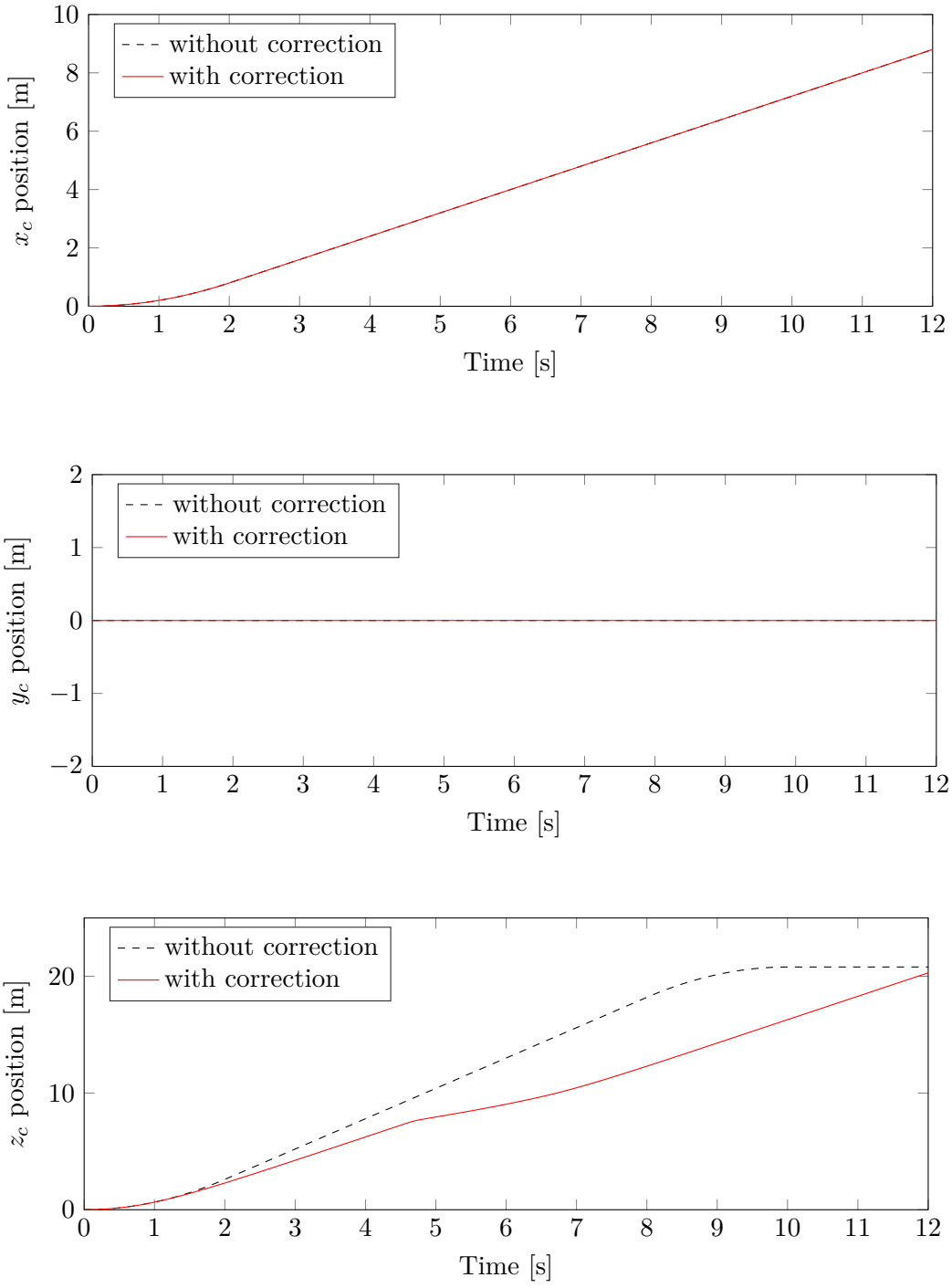


Figure 5.15: Controlled reference state $\boldsymbol{\xi}^r_{\text{contr}}$ versus $\boldsymbol{\xi}^r$.

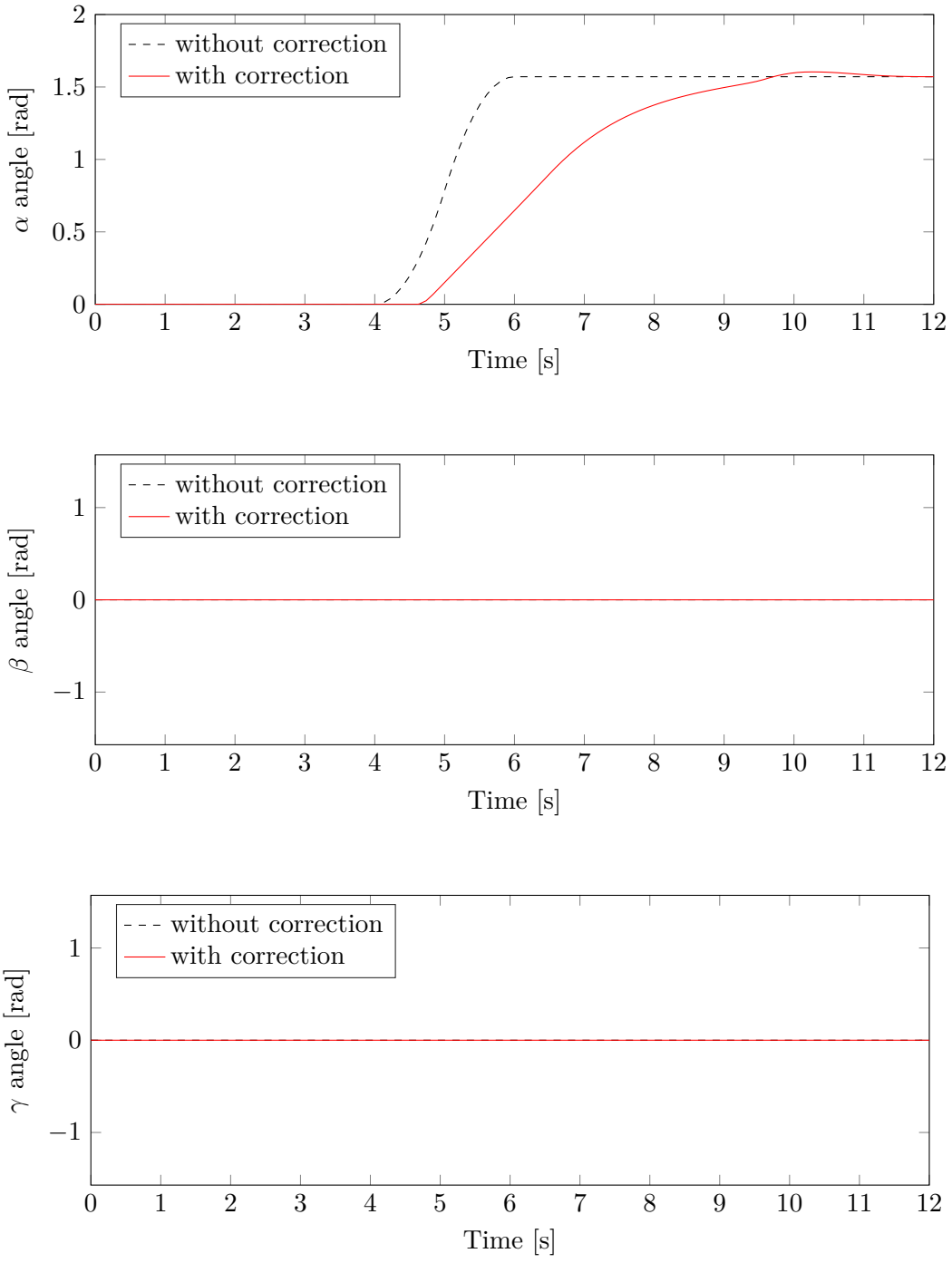


Figure 5.16: Controlled reference state $\boldsymbol{\xi}^r_{\text{contr}}$ versus $\boldsymbol{\xi}^r$.

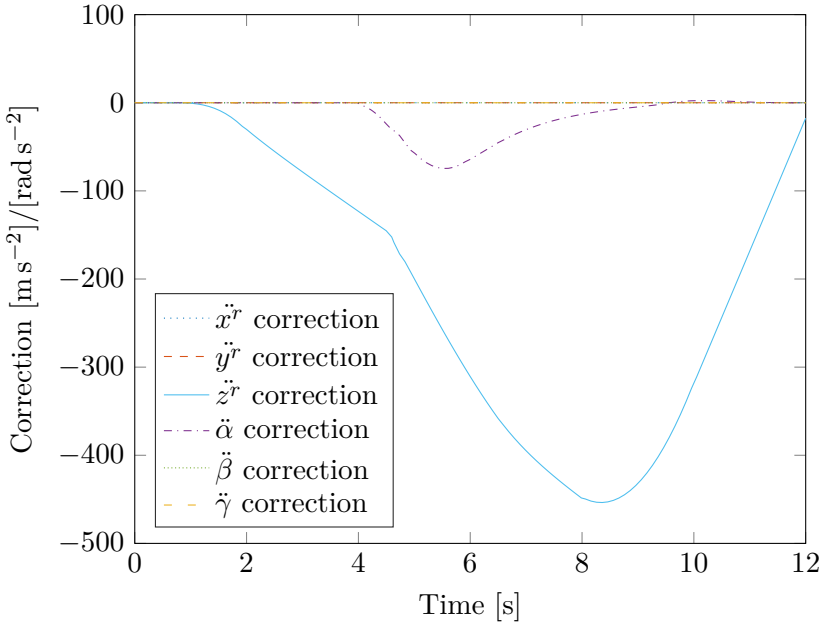


Figure 5.17: Correction vector $\boldsymbol{c}_a$.

5.5.3 Comparison

The absolute differences between desired vehicle positions and actual locations

$$\boldsymbol{e}_{r_i} = |\boldsymbol{r}_i^d - \boldsymbol{r}_i|, \quad i = \{1, \ldots, n\} \tag{5.12}$$

can be a measure for formation keeping accuracy. Let

$$J_{r_i} = \int_{0\,\mathrm{s}}^{12\,\mathrm{s}} ||\boldsymbol{e}_{r_i}||_1 \mathrm{d}t, \tag{5.13}$$

be a performance criteria for each vehicle and

$$J_{\text{total}} = \sum_{i=1}^{n} J_{r_i} \tag{5.14}$$

for the formation as a whole. In Figure 5.18, the values J_{total} of simulation results with and without reference correction are compared. It can be clearly seen that without reference correction the performance is far worse. Figure 5.19 gives an even better understanding as it shows the individual performance values J_{r_i} for each vehicle without reference correction. When the rotation sets in at $t = 4\,\mathrm{s}$, the J_{r_i} values start to increase faster as high position errors occur. This backs the observations form Chapter 5.5.2 where the formation shape gets lost and the vehicles almost converge to a single point in Figure 5.12. Contrary, the individual performance values resemble straight lines which are not greatly influenced by the rotation of the reference frame in Figure 5.20 as the reference correction was active.

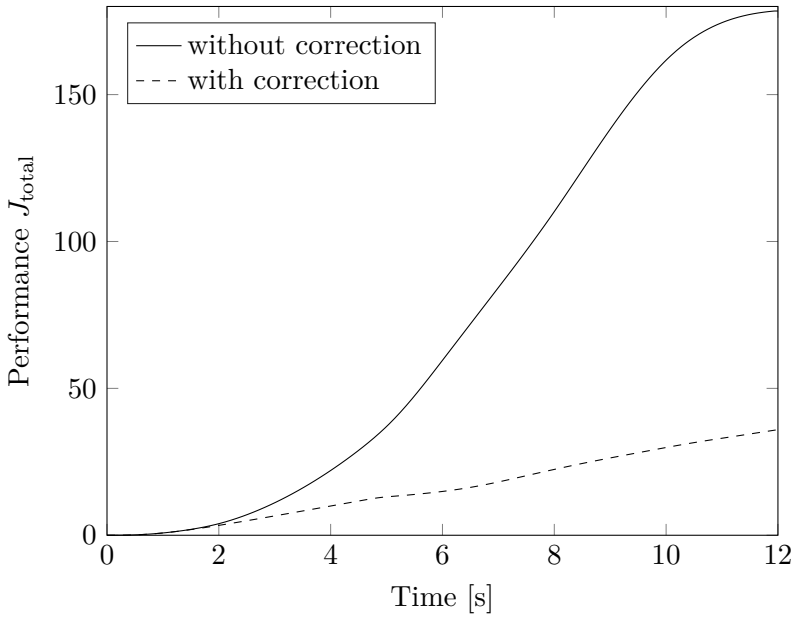


Figure 5.18: Total formation performance.

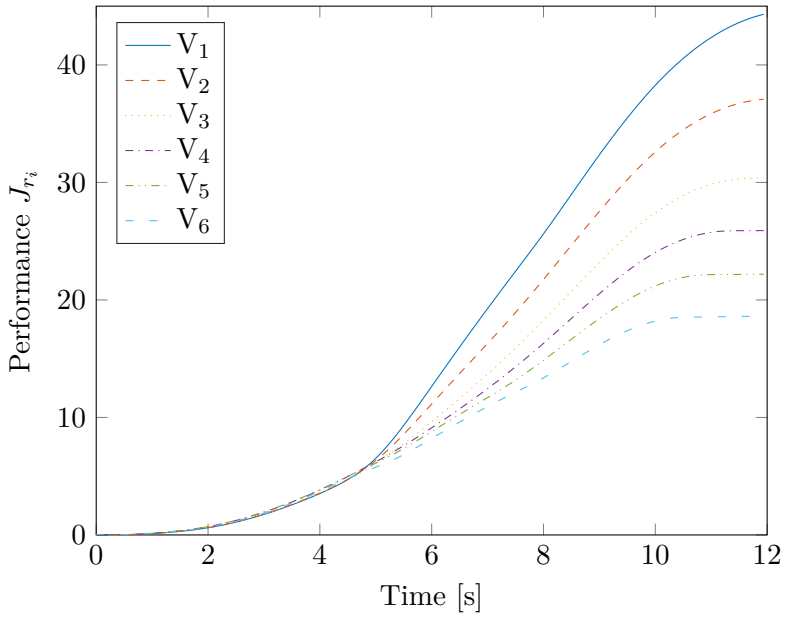


Figure 5.19: Vehicle performance without reference correction.

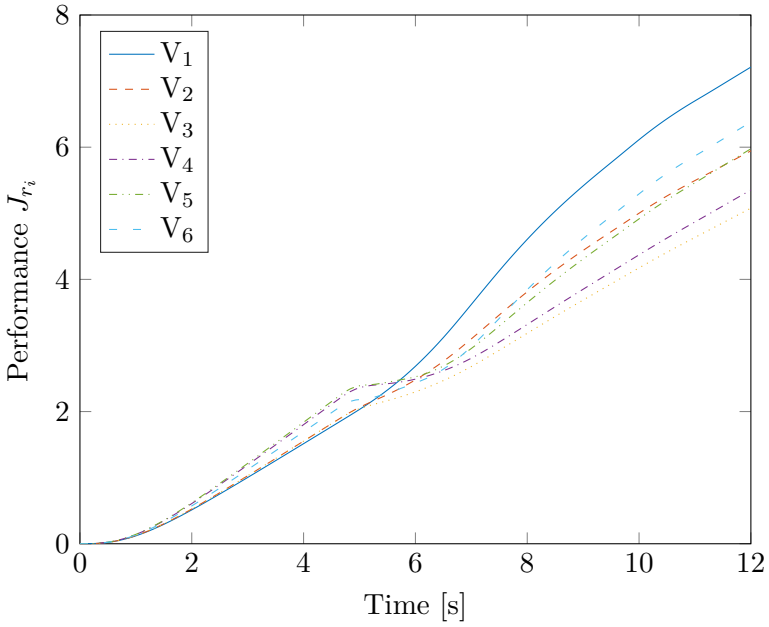


Figure 5.20: Vehicle performance with reference correction.

Chapter 6

Conclusion

In the present thesis, a complete formation control framework is presented. It allows the control of large swarms of vehicles in geometric configurations. As a novel extension to traditional formation control strategies, formation feedback is added. Thereby, the additional available maximum distance information is used to influence the reference trajectory which ultimately leads to better position keeping accuracy. To validate the effectiveness of the proposed approach, a quadrotor model is introduced and its associated local control. Simulation results with a group of these vehicles show that the solution with a controlled reference trajectory substantially improves formation keeping accuracy. The main contribution of this work lies in the mathematical formulation of the QP problem which allows the optimization of the reference trajectory in real-time and uses the formation feedback. With the distributed approach, the robustness to agent failures is given compared to a centralized solution. Only local neighbor-to-neighbor communication is necessary which allows to easily change the number of vehicles, formation shape or communication structure.

One limitation of the proposed framework is the necessity of a global exact position measuring system for the individual vehicles. This might be given in closed indoor research environments, however real world outdoor situations pose additional challenges. Furthermore, the communication network is assumed to be free of delays, without data losses and unlimited in bandwidth. For large numbers of vehicles these assumptions might not be reasonable. The influence of communication problems is well studied for consensus algorithms in particular but not for complete formation control frameworks [OFM07].

Combined, these limitations are part of the reason why experimental validations for quadrotors in formation flight with a large number of vehicles are rare. Together with the technical complexity, the costs are another factor. Yet, the proposed framework could also be realized for a group of ground robots. Lastly, the proposed approach assumes the same vehicle type for all agents in the formation because the reference correction is calculated based on the dynamic limitations of one individual vehicle. This contradicts real world applications where a swarm of heterogeneous vehicles might be participating. Future research could examine the adaption of the formation control framework for this scenario.

Appendix A

Rotation Matrices

Rotations in the three dimensional Euclidean space can be expressed through matrices. By multiplying the matrix $\boldsymbol{R}$ with a vector $\boldsymbol{v}$, the vector coordinates are turned around the origin. For each coordinate axis, there exists a basic rotation matrix where a positive angle leads to counter-clockwise rotation as shown in Figure A.1. The dashed arrows indicate the rotated system. Concerning the first coordinate axis, a rotation is described as

$$\boldsymbol{R}(\phi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi & \cos\phi \end{bmatrix}. \tag{A.1}$$

The corresponding rotation matrix of the second coordinate matrix can be expressed as

$$\boldsymbol{R}(\theta) = \begin{bmatrix} \cos\theta & 0 & \sin\theta \\ 0 & 1 & 0 \\ -\sin\theta & 0 & \cos\theta \end{bmatrix}. \tag{A.2}$$

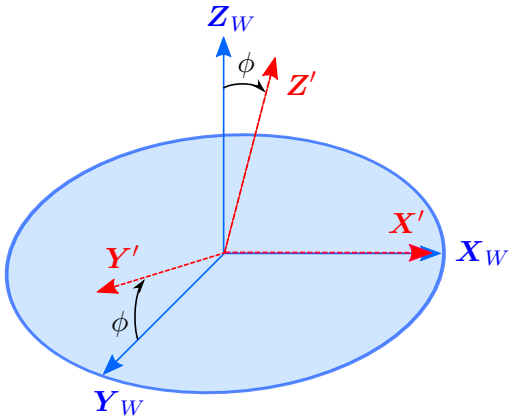
Finally, a rotation around the third coordinate axis is formulated as

$$\boldsymbol{R}(\psi) = \begin{bmatrix} \cos\psi & -\sin\psi & 0 \\ \sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{A.3}$$

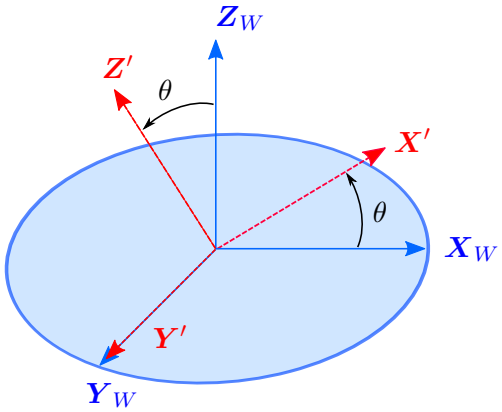
Any rotation in space can be given as a composition of these basic rotations by first rotating around the third axis, then the second, and lastly the first axis. This gives following rotational matrix

$${}^{W}\boldsymbol{R}_{B} = \boldsymbol{R}(\phi)\boldsymbol{R}(\theta)\boldsymbol{R}(\psi) = \begin{bmatrix} c_\psi c_\theta & c_\psi s_\theta s_\phi - s_\psi c_\phi & c_\psi s_\theta c_\phi + s_\psi s_\phi \\ s_\psi c_\theta & s_\psi s_\theta s_\phi + c_\psi c_\phi & s_\psi s_\theta c_\phi - c_\psi s_\phi \\ -s_\theta & c_\theta s_\phi & c_\theta c_\phi \end{bmatrix} \tag{A.4}$$

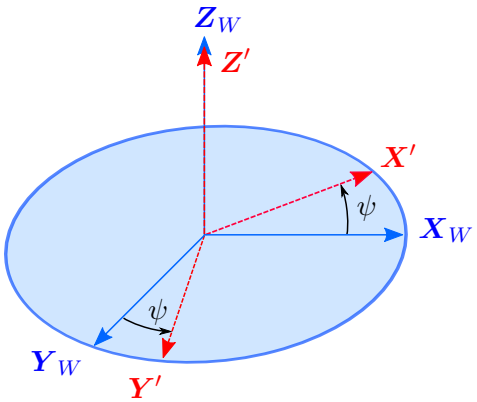
to describe a general rotation from a body coordinate to a world coordinate system [ER96]. For the reverse direction from world to body coordinates, the transformation matrix ${}^{B}\boldsymbol{R}_{W}$ is obtained by calculating the inverse of ${}^{W}\boldsymbol{R}_{B}$.



(a) Rotation around the first coordinate axis.



(b) Rotation around the second coordinate axis.



(c) Rotation around the third coordinate axis.

Figure A.1: Three basic rotations in Euclidean space.

Appendix B

Rigid Body Kinematics

The movement of a rigid body in space can been seen as an superposition of two single motions: *translation* and *rotation*. A translation moves all points of a body the same distance and therefore keeping its orientation fixed. Hence, velocities and accelerations of any body points are exactly the same [Gro+15]. If $\boldsymbol{r}$ denotes an arbitrary point of the body, then its velocity and acceleration are

$$\boldsymbol{v} = \dot{\boldsymbol{r}}, \qquad \boldsymbol{a} = \dot{\boldsymbol{v}} = \ddot{\boldsymbol{r}}. \tag{B.1}$$

Contrary, a rotational movement rotates all points of rigid body around a temporary axis with the fixed point $\boldsymbol{r}_A$. Let $\boldsymbol{\omega}$ denote the angular velocity vector of a body, then the velocity of an arbitrary point $\boldsymbol{r}_P$ is

$$\boldsymbol{v}_P = \boldsymbol{\omega} \times \boldsymbol{r}_{AP} \tag{B.2}$$

where $\boldsymbol{r}_{AP} = \boldsymbol{r}_P - \boldsymbol{r}_A$. The acceleration is obtained by taking the derivative of the velocity

$$\boldsymbol{a}_P = \frac{\mathrm{d}\boldsymbol{v}_P}{\mathrm{d}t} = \dot{\boldsymbol{\omega}} \times \boldsymbol{r}_{AP} + \boldsymbol{\omega} \times \dot{\boldsymbol{r}}_{AP} \tag{B.3}$$

As $\dot{\boldsymbol{r}}_{AP} = \dot{\boldsymbol{r}}_P - \dot{\boldsymbol{r}}_A$ and $\dot{\boldsymbol{r}}_A = 0$ due to the property of being fixed, (B.3) can be rewritten to

$$\boldsymbol{a}_P = \dot{\boldsymbol{\omega}} \times \boldsymbol{r}_{AP} + \boldsymbol{\omega} \times (\boldsymbol{\omega} \times \boldsymbol{r}_{AP}). \tag{B.4}$$

To analyze the case of general motion, a coordinate frame $\mathcal{F}_B$ is introduced which moves translational but does not rotate as illustrated in Figure B.1. For an observer situated in point A the body only rotates according to (B.2) and (B.4). On top of this observed motion, the translation of A has to be added which gives

$$\boldsymbol{v}_P = \boldsymbol{v}_A + \boldsymbol{\omega} \times \boldsymbol{r}_{AP} \tag{B.5}$$

$$\boldsymbol{a}_P = \boldsymbol{a}_A + \dot{\boldsymbol{\omega}} \times \boldsymbol{r}_{AP} + \boldsymbol{\omega} \times (\boldsymbol{\omega} \times \boldsymbol{r}_{AP}) \tag{B.6}$$

to describe the velocity and acceleration of any point of a rigid body.

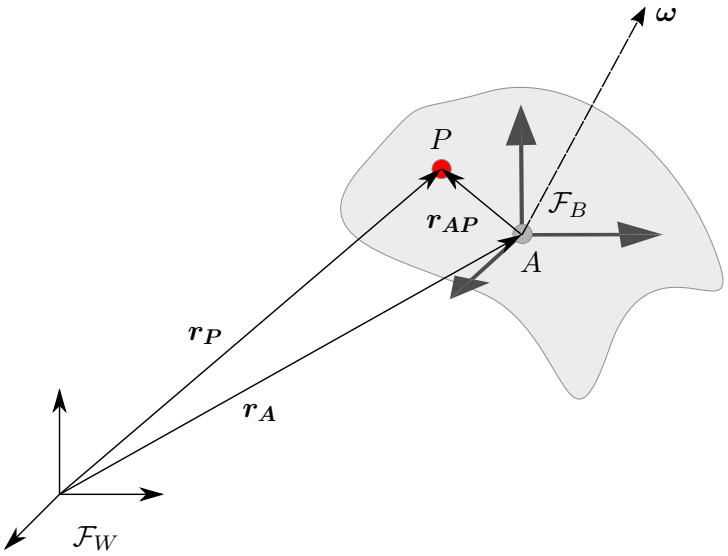


Figure B.1: Rigid Body Movement.

List of Figures

List of Tables

Bibliography

[Ard06] Mark D. Ardema. *Newton-Euler Dynamics*. 1. Aufl. s.l.: Springer Science + Business Media, 2006. ISBN: 0-387-23276-1. URL: http://gbv.eblib.com/patron/FullRecord.aspx?p=324170.

[BAW11] He Bai, Murat Arcak, and John Wen. *Cooperative control design: A systematic, passivity-based approach*. Communications and Control Engineering. New York, NY: Springer, 2011. ISBN: 9781461400141. URL: http://www.loc.gov/catdir/enhancements/fy1316/2011929229-d.html.

[BBS16] Igor Henrique Beloti Pizetta, Alexandre Santos Brandao, and Mario Sarcinelli-Filho. „Cooperative quadrotors carrying a suspended load“. In: *2016 International Conference on Unmanned Aircraft Systems (ICUAS)*. IEEE, 2016, pp. 1049–1055. ISBN: 978-1-4673-9334-8. DOI: 10.1109/ICUAS.2016.7502605.

[Bou07] Samir Bouabdallah. „Design and Control of Quadrotors with Application to autonomous Flying“. Dissertation. Lausanne: Ecole Polytechnique Federale de Lausanne, 2007.

[Boy+04] S. Boyd et al. „Analysis and optimization of randomized gossip algorithms“. In: *2004 43rd IEEE Conference on Decision and Control (CDC) (IEEE Cat. No.04CH37601)*. 2004, 5310–5315 Vol.5. DOI: 10.1109/CDC.2004.1429652.

[Cui+10] Rongxin Cui et al. „Leader–follower formation control of underactuated autonomous underwater vehicles“. In: *Ocean Engineering* 37.17-18 (2010), pp. 1491–1502. ISSN: 00298018. DOI: 10.1016/j.oceaneng.2010.07.006.

[ER96] Bernard Etkin and Lloyd Duff Reid. *Dynamics of flight: Stability and control*. 3. ed. New York: Wiley, 1996. ISBN: 0471034185. URL: http://www.loc.gov/catdir/description/wiley033/95020395.html.

[Fer+13] H. C. T. E. Fernando et al. „Modelling, simulation and implementation of a quadrotor UAV“. In: *2013 IEEE 8th International Conference on Industrial and Information Systems (ICIIS)*. 2013, pp. 207–212. DOI: 10.1109/ICIInfS.2013.6731982.

[Gar+13] Luis Rodolfo García Carrillo et al. *Quad Rotorcraft Control: Vision-Based Hovering and Navigation*. Advances in Industrial Control. London: Springer, 2013. ISBN: 978-1-4471-4399-4. DOI: 10.1007/978-1-4471-4399-4. URL: http://site.ebrary.com/lib/alltitles/docDetail.action?docID=10656388.

[Gho+10] Jawhar Ghommam et al. „Formation path following control of unicycle-type mobile robots“. In: *Robotics and Autonomous Systems* 58.5 (2010), pp. 727–736. ISSN: 09218890. DOI: 10.1016/j.robot.2009.10.007.

[GR04] Christopher David Godsil and Gordon Royle. *Algebraic graph theory*. Vol. 207. Graduate texts in mathematics. New York: Springer, 2004. ISBN: 9780387952413.

[Gro+15] Dietmar Gross et al. *Technische Mechanik 3: Kinetik*. 13., überarb. Aufl. Springer-Lehrbuch. Berlin: Springer Vieweg, 2015. ISBN: 9783642539534. DOI: 10.1007/978-3-642-53954-1. URL: http://dx.doi.org/10.1007/978-3-642-53954-1.

[Kur15] Yasuhiro Kuriki. „A Study on Cooperative Formation Control for a Multi-UAV System“. Dissertation. Tokyo: Keio University, 2015.

[Lun14] Jan Lunze. *Regelungstechnik 1: Systemtheoretische Grundlagen, Analyse und Entwurf einschleifiger Regelungen*. 10., aktual. Aufl. Springer-Lehrbuch. Berlin: Springer Vieweg, 2014. ISBN: 978-3-642-53908-4. DOI: 10.1007/978-3-642-53909-1. URL: http://dx.doi.org/10.1007/978-3-642-53909-1.

[Lyn96] Nancy A. Lynch. *Distributed algorithms*. The Morgan Kaufmann series in data management systems. 1996. ISBN: 1558603484. URL: http://search.ebscohost.com/login.aspx?direct=true&scope=site&db=nlebk&AN=207317.

[OFM07] Reza Olfati-Saber, J. Alex Fax, and Richard M. Murray. „Consensus and Cooperation in Networked Multi-Agent Systems“. In: *Proceedings of the IEEE* 95.1 (2007), pp. 215–233. ISSN: 0018-9219. DOI: 10.1109/JPROC.2006.887293.

[OM03] Reza Olfati-Saber and R. M. Murray. „Consensus protocols for networks of dynamic agents“. In: *2003 American Control Conference, 2003*. 2003, pp. 951–956. DOI: 10.1109/ACC.2003.1239709.

[OPA14] Kwang-Kyo Oh, Myoung-Chul Park, and Hyo-Sung Ahn. „A survey of multi-agent formation control“. In: *Automatica* 53 (2014), pp. 424–440. ISSN: 00051098. DOI: 10.1016/j.automatica.2014.10.022.

[RB08] Wei Ren and Randal W. Beard. *Distributed Consensus in Multi-vehicle Cooperative Control: Theory and Applications*. Communications and Control Engineering. London: Springer-Verlag London Limited, 2008. ISBN: 978-1-84800-014-8. DOI: 10.1007/978-1-84800-015-5. URL: http://dx.doi.org/10.1007/978-1-84800-015-5.

[Rey87] Craig W. Reynolds. „Flocks, herds and schools: A distributed behavioral model“. In: *the 14th annual conference*. Ed. by Maureen C. Stone. 1987, pp. 25–34. DOI: 10.1145/37401.37406.

[RS08] Wei Ren and Nathan Sorensen. „Distributed coordination architecture for multi-robot formation control“. In: *Robotics and Autonomous Systems* 56.4 (2008), pp. 324–333. ISSN: 09218890. DOI: 10.1016/j.robot.2007.08.005.

[Rud+14] Konrad Rudin et al. *Towards Estimation and Correction of Wind Effects on a Quadrotor UAV*. 2014. DOI: 10.3929/ethz-a-010286793.

[Sch13] Thomas Schütz, ed. *Hucho - Aerodynamik des Automobils*. Wiesbaden: Springer Fachmedien Wiesbaden, 2013. DOI: 10.1007/978-3-8348-2316-8.

[SKB07] P. B. Sujit, Derek Kingston, and Randy Beard. „Cooperative forest fire monitoring using multiple UAVs“. In: *46th IEEE Conference on Decision and Control, 2007*. Piscataway, NJ: IEEE Service Center, 2007, pp. 4875–4880. ISBN: 978-1-4244-1497-0. DOI: 10.1109/CDC.2007.4434345.

[Wes01] Douglas Brent West. *Introduction to graph theory*. 2. ed. Upper Saddle River, NJ: Prentice Hall, 2001. ISBN: 0130144002.

Erklärung des Autors

der Bachelorarbeit mit dem Titel

Distributed Consensus-based Formation Control of Quadrotors with Formation Feedback

Hiermit versichere ich,

1. dass ich meine Arbeit bzw. bei einer Gruppenarbeit den entsprechend gekennzeichneten Anteil der Arbeit selbständig verfasst habe,
2. dass ich keine anderen als die angegebenen Quellen benutzt und alle wörtlich oder sinngemäß aus anderen Werken übernommenen Aussagen als solche gekennzeichnet habe,
3. dass die eingereichte Arbeit weder vollständig noch in wesentlichen Teilen Gegenstand eines anderen Prüfungsverfahrens gewesen ist,
4. dass ich die Arbeit weder vollständig noch in Teilen bereits veröffentlicht habe und
5. dass das elektronische Exemplar mit den anderen Exemplaren übereinstimmt.

Stuttgart, den 15.11.2016

Jens Petit

Inofficial translation of the Declaration of Authorship („Erklärung des Autors“). Only the original German wording is legally recognized.

I hereby certify

1. that this thesis, resp. in a group project the appropriately labeled parts of this thesis, has/have been created by myself,
2. that no other sources than those referenced have been used, and that all ideas adopted from other work, literally or paraphrased, are appropriately labeled as such,
3. that this submitted thesis neither in full nor to a significant part has been part of any other examination process,
4. that I have neither published this thesis in full nor in parts, and
5. that the electronic version is identical to the hardcopy.