Planning and Control of Cooperative Multi-Agent Manipulator-Endowed Systems

CHRISTOS K. VERGINIS

Licentiate Thesis
Stockholm, Sweden, 2018

KTH Royal Institute of Technology
School of Electrical Engineering and Computer Science
Automatic Control Lab
SE-100 44 Stockholm
SWEDEN

TRITA-EECS-AVL-2018:5
ISBN: 978-91-7729-665-2

Akademisk avhandling som med tillstånd av Kungliga Tekniska högskolan framlägges till offentlig granskning för avläggande av teknologie licentiatexamen i Elektro- och systemteknik fredagen den 16 mars 2018 klockan 10.00 i sal Q2 Kungliga Tekniska högskolan, Osquldas väg 10, KTH, Stockholm.

Tryck: Universitetsservice US AB

Abstract

Multi-agent planning and control is an active and increasingly studied topic of research, with many practical applications, such as rescue missions, security, surveillance, and transportation. More specifically, cases that involve complex manipulator-endowed systems deserve extra attention due to potential complex cooperative manipulation tasks and their interaction with the environment. This thesis addresses the problem of cooperative motion- and task-planning of multi-agent and multi-agent-object systems under complex specifications expressed as temporal logic formulas. We consider manipulator-endowed robotic agents that can coordinate in order to perform, among other tasks, cooperative object manipulation/transportation. Our approach is based on the integration of tools from the following areas: multi-agent systems, cooperative object manipulation, discrete abstraction design of multi-agent-object systems, and formal verification. More specifically, we divide the main problem into three different parts. The first part is devoted to the control design for the formation control of a team of rigid-bodies, motivated by its application to cooperative manipulation schemes. We propose decentralized control protocols such that desired position and orientation-based formation between neighboring agents is achieved. Moreover, inter-agent collisions and connectivity breaks are guaranteed to be avoided. In the second part, we design continuous control laws explicitly for the cooperative manipulation/transportation of an object by a team of robotic agents. Firstly, we propose robust decentralized controllers for the trajectory tracking of the object's center of mass. Secondly, we design model predictive control-based controllers for the transportation of the object with collision and singularity constraints. In the third part, we design discrete representations of multi-agent continuous systems and synthesize hybrid controllers for the satisfaction of complex tasks expressed as temporal logic formulas. We achieve this by combining the results of the previous parts and by proposing appropriate trajectory tracking- and potential field-based continuous control laws for the transitions of the agents among the discrete states. We consider teams of unmanned aerial vehicles and mobile manipulators as well as multi-agent-object systems where the specifications of the objects are also taken into account. Numerical simulations and experimental results verify the claimed results.

Sammanfattning

Planering och reglering av multiagent-system är ett aktivt och växande forskningsfält med en rad praktiska tillämpningar såsom räddningsuppdrag, transport, övervakning och säkerhet. De fall där komplexa manipulatorbaserade system ingår förtjänar extra uppmärksamhet eftersom de utför potentiellt komplexa och samarbetskrävande manipulationsuppgifter och kräver interaktion med omgivningen. Denna avhandling behandlar problem med rörelse- och uppgifts-planering av samarbetande multiagenter och multiagent-objekt-system under komplexa specifikationer uttryckta med temporallogiska formler. Vi betraktar manipulatorbaserade robotagenter som kan koordineras för att utföra bland annat samarbetande manipulation/transport av objekt. Vår ansats är baserad på integration av verktyg från följande områden: multiagent-system, samarbetsbaserad objektsmanipulation, diskret abstraktionsdesign av multiagent-objekt-system samt formell verifikation. Mer specifikt delar vi in huvudproblemet i tre olika delar. Den första delen tillägnas reglerdesign för formationsreglering av en grupp stelkroppsagenter, vilket kan motiveras av dess tillämpning till samarbetskrävande manipulationsuppgifter. Vi föreslår decentraliserade reglerprotokoll så att önskade positions- och orienterings-formationer uppnås mellan närliggande agenter. Dessutom garanteras att kollisioner och förlorad anslutning mellan agenter undviks. I den andra delen designar vi kontinuerliga styrlagar explicit för manipulation/transport av ett objekt utfört av en grupp robotagenter. Först föreslår vi robusta decentraliserade regulatorer för trajektoria-spårning av ett objekts masscentrum. Sedan utvecklar vi modell-prediktiva regulatorer för transport av objektet med bivillkor för kollisioner och singulariteter. I den tredje delen designar vi diskreta representationer av kontinuerliga multiagentsystem, och syntetiserar hybrida regulatorer som uppfyller komplexa uppgifter uttryckta med temporallogik. Vi uppnår detta genom att kombinera resultat från tidigare delar och genom att föreslå passande kontinuerliga reglerprotokoll, baserade på positionsspårning och potentialfält, för agenternas övergångar mellan de diskreta tillstånden. Vi betraktar grupper av obemannade flygfordon och mobila manipulatorer, samt multiagent-objekt-system där även objektens specifikationer tas i beaktning. Numeriska simuleringar och experimentella resultat verifierar de hävdade resultaten.

Acknowledgements

I would like first of all to express my gratitude to my supervisor Prof. Dimos Dimarogonas for his invaluable support and guidance in my work; his continuous feedback and inspiration has made this thesis possible, and it has been an excellent experience to work with him. I would also like to give special thanks to my co-supervisor Prof. Danica Kragic for her great insight, knowledge and our collaboration on the Knut och Alice Wallenberg Foundation project IPSYS.

I thank all my colleagues (current and former) at the Automatic Control department of KTH, for creating a nice working atmosphere, and for always keeping the door open to inspiring conversations. Special thanks go to my office roommates Pedro Pereira, Goncalo Pereira, Yulong Gao, Manne Held, and Rong Du for creating a friendly atmosphere as well as to my colleagues and good friends Alexandros Nikou and Lars Lindemann for the continuous offer for assistance and inspiration to me.

Many thanks also to Pedro Roque, Matteo Mastellaro, Ziwei Xu, and Imran Khan for the great collaboration and work during their master thesis and all the professors and the administrative staff of the Department.

I am grateful to the Knut och Alice Wallenberg Foundation, the European Union through the project AEROWORKS and Co4Robots, and the Swedish Foundation for Strategic Research, for the financial support that made this work possible.

Finally, I would like to express my gratitude to my girlfriend, Stella Tsiona, my family and my friends for their constant moral support.

Christos Verginis
March 16, 2018.

Contents

Abbreviations

CTL	Computational Tree Logic
LTL	Linear Temporal Logic
MILP	Mixed-Integer Linear Programming
MTL	Metric Temporal Logic
MITL	Metric Interval Temporal Logic
NBA	Nondeterministic Büchi Automaton
NF	Navigation Function
NMPC	Nonlinear Model Predictive Control
PPC	Prescribed Performance Control
ROCP	Robust Optimal Control Problem
TBA	Timed Büchi Automaton
TS	Transition System
UAV	Unmanned Aerial Vehicle
WTS	Weighted Transition System

List of Figures

Chapter 1

Introduction

1.1 Motivation

The technological developments have been increasing exponentially during the last century, with an evident peak in the last few decades. The recent need for development of smart cities (including autonomy in industrial buildings, houses, highways, as well as automated rescue missions) calls for wider deployment of robots that must coordinate with each other to achieve a specific task. Additionally, noteworthy is the increasing evolution of wireless communication technology that results in the low-cost massive development of (internal and external) sensor devices. Along with the incapability of the corresponding computing units to process very large amounts of data in small amounts of time, this has given rise to a special case of systems that consist of multiple robots, namely multi-agent systems. Multi-agent systems consist of agents/robots that rely solely on local sensor information with respect to their neighboring robots to determine their actions, which is often called *decentralized control*.

During the last decade, decentralized control of multi-agent systems has gained a significant amount of attention due to the great variety of its applications, including multi-robot systems, transportation, multi-point surveillance and biological systems. The main focus of multi-agent systems is the design of distributed control protocols in order to achieve global tasks, such as *consensus* [1–5], in which all the agents are required to converge to a specific point and *formation* [6, 7], in which all the agents aim to form a predefined geometric shape. At the same time, the agents might need to fulfill certain transient properties, such as *network connectivity* [8–10] and/or *collision avoidance* [11].

A special case of multi-agent systems is cooperative robotic manipulators. In particular, when it comes to object manipulation/transportation, large/heavy payloads as well as complex maneuvers necessitate the deployment of more than one robot. The most common tasks consist of pick-and-place tasks and cooperative object transportation, while satisfying certain properties, such as collision- and singularity-avoidance.

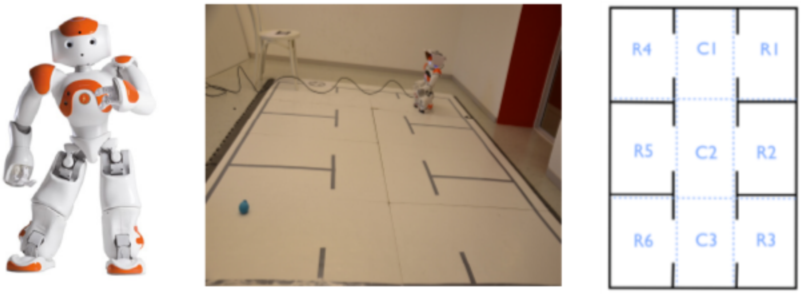


Figure 1.1: A humanoid robot moving to an environment consisting of 6 rooms and 3 corridor regions. In room $R6$ there exists a ball that the robot can grab.

Another topic that has troubled researchers the last decades is the control of multiple systems such that each agent/robot fulfills desired tasks given by high-level specifications expressed as temporal logic formulas. Temporal-logic based motion planning has gained a significant amount of attention over the last decade, since it provides a fully automated correct-by-design controller synthesis approach for autonomous robots. Temporal logics, such as linear temporal logic (LTL), provide formal high-level languages that can describe planning objectives more complex than the well-studied navigation algorithms, and have been used extensively both in single- as well as in multi-agent setups. The objectives are given as a temporal logic formula with respect to a discretized abstraction of the system (usually a finite transition system), and then, a high-level discrete path is found by off-the-shelf model-checking algorithms, given the abstracted system and the task specification. Consider, for instance, the robot in Figure 1.1 operating in a workspace which is partitioned into 6 rooms and a corridor consisting of three regions. A high-level task for the robot might have the following form: "Periodically visit rooms R_1, R_4, R_6, in this order, while avoiding rooms R_2, R_3 and R_5", or "Grab the ball that lies in room R_6 and deliver it in room R_3 between 10 and 20 seconds". The aforementioned specifications include complex tasks where *time* might play an important role.

One of the main problems that arise when dealing with high-level tasks based on temporal-logic formulas is the construction of a discrete abstracted representation of the continuous system. More specifically, given a temporal-logic formula over a continuous workspace/state space, how does one partition this space into discrete states? Moreover, given a predefined partition, what are the control inputs of the agents that guarantee well-defined transitions among the discrete states? When multi-agent systems are concerned, the aforementioned specifications must also incorporate collision-avoidance as well as connectivity-maintenance properties among the robots, which brings the problem of abstraction to a new level of complexity.

Furthermore, consider a case where some unactuated objects must undergo a series of processes in a workspace with autonomous agents (e.g., car factories), expressed as temporal-logic high-level specifications. In such cases, the agents, except for satisfying their own motion specifications, are also responsible for coordinating with each other in order to transport the objects around the workspace. When the unactuated objects' specifications are expressed using temporal logics, then the motion- and task- planning of the agents' behavior becomes much more complex, since the discrete system abstraction has to also take into account the objects' goals.

Motivated by the above discussion, this thesis aims at solving the problem of decentralized motion- and task-planning of multi-agent and multi-agent-object systems under complex task specifications by integrating tools from the computer science and automatic control fields. The main contributions lie in the abstraction of the continuous coupled object-agents dynamics into a discrete representation of the system (transition systems) and the application of formal verification methodologies towards the satisfaction of temporal logic formulas. More specifically, we break down the problem into three main subproblems. Motivated by the need of transition design for unactuated objects, we consider the problem of cooperative object manipulation in the first two parts. In the first part, we address the problem of multi-agent formation control, since one of its many applications is cooperative object manipulation [12]. In the second part, we model explicitly the coupled system that consists of an object grasped by multiple robotic agents, and we tackle the problem of constrained pose and time trajectory tracking of the object's center of mass. The third part addresses the discrete abstractions of multi-agent systems and the control synthesis for the satisfaction of high level specifications. We consider the multi-agent navigation problem as a means for designing multi-agent transition systems and synthesizing control plans that satisfy the agents' specifications. Finally, we combine the results from the previous parts to build multi-agent-object coupled transition systems and synthesize controllers that incorporate the task specifications of the unactuated objects. In the following, we list the problems we address, by further subdividing the third part:

1. Consider a multi-agent system modeled by 2nd order Lagrangian dynamics. The goal is to design decentralized controllers that use only local information with respect to the neighboring agents such that a predefined geometric formation is achieved, while guaranteeing inter-agent collision avoidance and connectivity maintenance. Among the numerous applications of the formation control problem, an important one is the cooperative manipulation case [12]. For instance, the center of mass of the object can be considered as a virtual leader that desires to track a predefined desired trajectory and the robots' end-effectors need to keep fixed distances with each other while complying with grasping constraints.

2. Consider an object rigidly grasped by a team of robotic agents (robotic manipulators). Given a prespecified pose/time trajectory, the goal is to design communication-free decentralized control laws for the agents to achieve tracking/regulation for the object's center of mass, robust to modeling uncertainties and external disturbances. The solution of the aforementioned problem can

be applied as the building block for transitions in a potential workspace/state space partition, in order to define discrete abstractions of the system (e.g., transition systems).

3. Consider a team of robotic agents operating in a bounded 3D workspace that contains predefined points of interest. The goal is to design well defined decentralized abstractions for the agents over the points of interest in order to synthesize controllers for the satisfaction of high-level temporal logic tasks. The problem in hand is equivalent to a) designing decentralized control laws that guarantee the navigation of the multi-agent team among the predefined points of interest, while guaranteeing inter-agent collision avoidance and connectivity maintenance, and b) applying formal-verification techniques to synthesize hybrid control protocols that guarantee satisfaction of the high-level tasks.

4. Consider a system comprised of multiple robotic agents and one or more objects. Similarly to problem 3, given a predefined partition of the 3D workspace, the goal is to design abstractions for the overall system that incorporate the motion/tasks of the agents as well as of the objects. This allows the synthesis of controllers that take into account the agents' as well as the objects' task specifications, which are modeled through high-level temporal-logic formulas.

Taking the aforementioned problems into consideration, this thesis is divided into three main parts. The first two parts deal with methodologies and control algorithms for solving Problems 1 and 2, whereas the third part considers Problems 3 and 4. The work developed in this thesis was supported by the research projects "H2020 Research and Innovation Programme" under the Grant Agreements No. 644128 (AEROWORKS) and No. 731869 (Co4Robots), the the H2020 ERC Starting Grant BUCOPHSYS, the Knut and Alice Wallenberg Foundation, the Swedish Research Council (VR), and the Swedish Foundation for Strategic Research. The next section presents the outline of this thesis.

1.2 Thesis Outline and Contributions

In this Section, we provide the outline of the thesis and indicate the contributions of each chapter. Chapter 2 is devoted to notation that will be adopted in this thesis and preliminary background knowledge. The thesis is divided into *four main parts* which aim to solve the Problems that were previously mentioned. In particular,

- The *first part* consists of Chapter 3. In this part, we propose a novel **decentralized control protocol for formation control** of a multi-agent system in $SE(3)$. The proposed control scheme guarantees position and orientation based formation, inter-agent collision avoidance, as well as connectivity maintenance among the agents of the initially connected graph.

- The *second part* consists of Chapters 4 and 5. In the second part we address the **cooperative manipulation/transportation of an object by a team of robotic agents**. Motivated by the need of designing transition relations to define discrete transition systems for unactuated objects, we propose novel continuous-time control methodologies that guarantee trajectory tracking as well as pose stabilization of an object rigidly grasped by a team of robotic agents.

- The *third part* consists of Chapters 6 and 7. This part addresses the problem of **defining abstractions for multi-agent robotic systems**. In Chapter 6 we propose continuous-time control laws for the navigation of multi-agent teams among predefined regions of interest, thus establishing well-defined transition systems for the agents. In Chapter 7, we proceed similarly and we incorporate unactuated objects in the designed transition systems, allowing the incorporation of high-level goals for the objects.

Chapter 3

This chapter presents a novel control protocol for the formation control of tree graphs in $SE(3)$. The control laws are decentralized (in the sense that each agent uses only local relative information from its neighbors to calculate its control signal) as well as robust to modeling uncertainties (parametric and structural) and external disturbances. The proposed methodology guarantees collision avoidance and connectivity maintenance among the initially connected agents. Moreover, certain predefined functions characterize the transient and steady state performance of the closed loop system. Finally, simulation results verify the validity and efficiency of the proposed approach. The covered material is based on the following contributions [13]:

- A. Nikou, C. K. Verginis and D. V. Dimarogonas, “Robust distance-based formation control of multiple rigid bodies with orientation alignment”, IFAC Proceedings Volumes, Toulouse, France, 2017.

- C. K. Verginis, A. Nikou and D. V. Dimarogonas, “Robust Formation Control of Tree Graphs in $\mathbb{SE}(3)$ with Prescribed Transient and Steady State Performance”, under preparation.

Chapter 4

This chapter addresses the problem of cooperative manipulation of a single object by multiple robotic agents. More specifically, we present two novel control methodologies for the trajectory tracking of the object’s center of mass. Firstly, we design an adaptive control protocol which employs quaternion feedback for the object orientation to avoid potential representation singularities. Secondly, we propose a control protocol that guarantees predefined transient and steady-state performance for the object trajectory. Both methodologies are decentralized, since the agents calculate their own signals without communicating with each other, as well as robust to external disturbances and model uncertainties. Moreover, we consider that the grasping points are rigid, and avoid the need for force/torque measurements. Load sharing coefficients are also introduced to account for potential differences in the agents' power capabilities. Finally, simulation and experimental results with two robotic arms verify the theoretical findings. The covered material is based on the following contribution [14, 15]:

- C. K. Verginis, M. Mastellaro and D. V. Dimarogonas, "Robust quaternion-based cooperative manipulation without force/torque information", IFAC Proceedings Volumes, Toulouse, France, 2017.

- C. K. Verginis, M. Mastellaro and D. V. Dimarogonas, "Cooperative manipulation without force/torque measurements: Control design and experiments", submitted to the IEEE Transactions on Control Systems Technology, 2018.

Chapter 5

This chapter addresses the problem of cooperative transportation of an object rigidly grasped by N robotic agents. In particular, we propose two Nonlinear Model Predictive Control (NMPC) schemes that guarantee the navigation of the object to a desired pose in a bounded workspace with obstacles, while complying with certain input saturations of the agents. The first control scheme is centralized, in the sense that a central unit calculates the control inputs for each of the robotic agents, whereas the second control scheme is based on inter-agent communication and is decentralized, since each agent calculates its own control signal. Moreover, the proposed methodologies ensure that the agents do not collide with each other or with the workspace obstacles as well as that they do not pass through singular configurations. The feasibility and convergence analysis of the NMPC are explicitly provided. Finally, simulation results illustrate the validity and efficiency of the proposed methods. The results presented in this chapter are based on [16, 17]:

- A. Nikou, C. K. Verginis and D. V. Dimarogonas, "A nonlinear model predictive control scheme for cooperative manipulation with singularity and collision avoidance", Proceedings of the IEEE Mediterranean Conference on Control and Automation (MED), Valletta, Malta, 2017.

- C. K. Verginis, A. Nikou and D. V. Dimarogonas, "Communication-based decentralized cooperative object transportation using nonlinear model predictive control", submitted to the IEEE European Control Conference (ECC), Limassol, Cyprus, 2018.

Chapter 6

This chapter addresses the motion planning problem for a team of manipulator-endowed systems under high level goals. We propose a hybrid control strategy that guarantees the accomplishment of each agent's local goal specification, which is given as a temporal logic formula, while guaranteeing inter-agent collision avoidance and connectivity maintenance. The overall approach is based on an abstraction of the continuous systems into discrete transition systems, which we accomplish by designing suitable decentralized continuous controllers based on previous work on navigation functions. Next, given specific high-level tasks encoded by temporal logic formulas, we employ standard formal verification techniques and we derive high-level control algorithms that satisfy the agents' specifications. Simulation and experimental results verify the validity of the proposed methods. These results are based on [18, 19]:

- C. K. Verginis, Z. Xu and D. V. Dimarogonas, "Decentralized motion planning with collision avoidance for a team of UAVs under high level goals", Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), Singapore, 2017.
- C. K. Verginis and D. V. Dimarogonas, "Robust decentralized abstractions for multiple mobile manipulators", Proceedings of the IEEE Conference on Decision and Control (CDC), Melbourne, Australia, 2017.

Chapter 7

This chapter addresses the problem of deriving well-defined abstractions for motion planning of a team of robotic agents and objects. In particular, we propose two methodologies for the discrete abstraction of such systems. Firstly, we propose a distributed model-free control protocol for the trajectory tracking of a cooperatively manipulated object without necessitating feedback of the contact forces/torques or inter-agent communication. By employing the prescribed performance control methodology, we pre-determine the transient and steady state of the coupled object-agents system. Along with a region partition of the workspace that depends on the physical volume of the object and the agents, this allows us to define timed transitions for the coupled system among the derived workspace regions. Therefore, we abstract its motion as a finite transition system and, by employing standard automata-based methodologies, we define high level complex tasks for the object that can be encoded by timed temporal logics. Secondly, we present a hybrid control framework for the motion planning of a multi-agent system including N robotic agents and M objects, under high level goals expressed as Linear Temporal Logic (LTL) formulas. We design control protocols that allow the transition of the agents as well as the cooperative transportation of the objects by the agents, among predefined regions of interest in the workspace. This allows to abstract the coupled behavior of the agents and the objects as a finite transition system and to design a high-level multi-agent plan that satisfies the agents' and the objects' specifications, given as temporal logic formulas. Simulation results verify the validity of the proposed frameworks. These results are based on [20–23]:

- C. K. Verginis and D. V. Dimarogonas, "Distributed cooperative manipulation under timed temporal specifications", Proceedings of the American Control Conference (ACC), Seattle, USA, 2017.
- C. K. Verginis and D. V. Dimarogonas, "Timed abstractions for distributed cooperative manipulation", Autonomous Robots, 2017.
- C. K. Verginis and D. V. Dimarogonas, "Multi-agent motion planning and object transportation under high level goals", IFAC Proceedings Volumes, Toulouse, France, 2017.
- C. K. Verginis, and D. V. Dimarogonas, "Motion and cooperative transportation planning for multi-agent systems under temporal logic formulas", submitted to the IEEE Transactions on Automation Science and Engineering, 2017.

Finally, in Chapter 8, conclusions of this thesis as well as future research directions are discussed.

Contributions not included in this thesis

The following publications are not covered in this thesis, but are related to the work presented here [24, 25]:

- C. K. Verginis, A. Nikou and D. V. Dimarogonas, "Position and orientation based formation control of multiple rigid bodies with collision and avoidance and connectivity maintenance", Proceedings of the IEEE International Conference on Decision and Control (CDC), 2017, Melbourne, Australia.
- L. Lindemann, C. K. Verginis and D. V. Dimarogonas, "Prescribed performance control for signal temporal logic specifications", Proceedings of the IEEE Conference on Decision and Control (CDC), Melbourne, Australia, 2017.
- A. Nikou, C. K. Verginis, S. Heshmati-alamdari and D. V. Dimarogonas, "Decentralized abstractions and timed constrained planning of a general class of coupled multi-agent systems", Proceedings of the IEEE Conference on Decision and Control (CDC), Melbourne, Australia, 2017.

Chapter 2

Notation and Preliminaries

In this chapter, the notation that will be used hereafter as well as the necessary background, are provided.

The set of positive integers is denoted by $\mathbb{N}$ and the real n-coordinate space, with $n \in \mathbb{N}$, by $\mathbb{R}^n$; $\mathbb{R}^n_{\geq 0}$ and $\mathbb{R}^n_{>0}$ are the sets of real n-vectors with all elements nonnegative and positive, respectively. The complex n-coordinate space is denoted as $\mathbb{C}^n$. The $n \times n$ identity matrix is denoted by I_n, the n-dimensional zero vector by 0_n and the $n \times m$ matrix with zero entries by $0_{n\times m}$. The n-dimensional vector of ones is denoted by $\mathbb{1}_n$. Given a matrix $A \in \mathbb{R}^{n\times m}$, we use $\|A\| := \sqrt{\lambda_{\max}(A^\top A)}$, where $\lambda_{\max}(\cdot)$ here denotes the maximum eigenvalue of a matrix; $\mathrm{rank}(A)$ is its rank; $\|A\|_{\mathrm{F}} := \mathrm{tr}(A^\top A)$ is the Frobenius norm of A, and $\mathrm{tr}(\cdot)$ is its trace; $\det(A)$ denotes its determinant. Given $a \in \mathbb{R}^3$, $S(a)$ is the skew-symmetric matrix defined according to $S(a)b = a \times b$. Given a nonempty and bounded set of natural numbers $\mathcal{X}$ and a set of vectors (matrices) $x_i, i \in \mathbb{N}$, we denote by $[x_i^\top]^\top_{i\in\mathcal{X}}$ the stack column-vector form with the vectors (matrices) whose indices belong to $\mathcal{X}$. Unless otherwise stated, $p^C_{B/A} \in \mathbb{R}^3$ denotes the vector that connects the origins of coordinate frames $\{A\}$ and $\{B\}$ expressed in frame $\{C\}$ coordinates in 3D space. Moreover, for notational brevity, when a coordinate frame corresponds to an inertial frame of reference $\{I\}$, we will omit its explicit notation (e.g., $p_B = p^I_{B/I}, \omega_B = \omega^I_{B/I}$), unless otherwise stated. Moreover, unless otherwise stated, $\eta_{A/B} = [\phi_{A/B}, \theta_{A/B}, \psi_{A/B}]^T \in \mathbb{T} \subset \mathbb{R}^3$ are the Euler angles representing the orientation of frame $\{A\}$ with respect to frame $\{B\}$, with $\phi_{A/B}, \theta_{A/B} \in (-\pi, \pi)$ and $\theta_{A/B} \in (-\frac{\pi}{2}, \frac{\pi}{2})$, and $\mathbb{T} = (-\pi, \pi) \times (-\frac{\pi}{2}, \frac{\pi}{2}) \times (-\pi, \pi)$; We also denote $\mathbb{M} = \mathbb{R}^3 \times \mathbb{T}$. All vector and matrix differentiations will be with respect to an inertial frame $\{I\}$, unless otherwise stated. The angular velocity of frame $\{B\}$ with respect to $\{A\}$ is denoted as $\omega_{B/A} \in \mathbb{R}^3$ and it holds that [26] $\dot{R}_{B/A} = S(\omega_{B/A})R_{B/A}$, where $R_{B/A} \in SO(3)$ is the corresponding rotation matrix, and $SO(3)$ is the 3-D rotation group. The values of a Boolean variable are $\top$ (True) and $\bot$ (False). Given a set S, denote by $|S|$ its cardinality, by $S^n = S \times \cdots \times S$ its n-fold Cartesian product, and by 2^S the set of all its subsets; ∂S stands for the boundary of the set S and $\mathring{S}$ for its interior. Given a finite sequence $s_1, \ldots, s_n$ of elements in S, with $n \in \mathbb{N}$, we denote by $(s_1, \ldots, s_n)^\omega$ the infinite sequence

$s_1, \dots, s_n s_1, \dots, s_n \dots$ created by repeating $s_1, \dots, s_n$. The notation $\|x\|$ is used for the Euclidean norm of a vector $x \in \mathbb{R}^n$; $A \otimes B$ denotes the Kronecker product of the matrices $A, B \in \mathbb{R}^{m \times n}$ (see [27]). The set-valued function $\mathcal{B} : \mathbb{R}^3 \times \mathbb{R}_{>0} \rightrightarrows \mathbb{R}^3$, given as $\mathcal{B}(c, r) = \{x \in \mathbb{R}^3 : \|x - c\| \leq r\}$, represents the 3D sphere with center $c \in \mathbb{R}^3$ and radius $r \in \mathbb{R}_{>0}$. Given a scalar function $y : \mathbb{R}^n \to \mathbb{R}$ and a vector $x \in \mathbb{R}^n$, denote by $\nabla_x y(x) = \left[\frac{\partial y(x)}{\partial x_1}, \dots, \frac{\partial y(x)}{\partial x_n}\right]^\top \in \mathbb{R}^n$ the gradient of y. Similarly, given a vector-valued function $y : \mathbb{R}^n \to \mathbb{R}^m$, denote by $\nabla_x y(x) = [\nabla_x y_1(x), \dots, \nabla_x y_m(x)]^\top \in \mathbb{R}^{m \times n}$. The special Euclidean group is denoted by $SE(3) \coloneqq \{(c, R) \in \mathbb{R}^3 \times SO(3)\}$. We define the induced norm in $\mathbb{SO}(3)^N$ as $\|R\|_T \coloneqq \sum_{i \in \{1, \dots, N\}} \|R_i\|_{\mathrm{F}}$ for any $R = (R_1, \dots, R_N) \in \mathbb{SO}(3)^N$. Moreover, the tangent space to $SO(3)$ at R is denoted by $T_R SO(3)$ and we also use $\mathbb{T}_R \coloneqq \mathbb{R}^3 \times T_R SO(3)$.

Definition 2.1. Given the sets S_1, S_2, their *Minkowski addition* is defined by:

$$S_1 \oplus S_2 = \{s_1 + s_2 : s_1 \in S_1, s_2 \in S_2\}.$$

Definition 2.2. Consider two sets $S_1, S_2 \subseteq \mathbb{R}^n$. Then, the *Pontryagin difference* is defined by:

$$S_1 \sim S_2 = \{x \in \mathbb{R}^n : s_1 + s_2 \in S_1, \forall\ s_2 \in S_2\}.$$

Lemma 2.1. *(Grönwall-Bellman Inequality) ([28, Appendix A]) Let $\bar{y} : [a, b] \to \mathbb{R}$ be continuous and $\widetilde{y} : [a, b] \to \mathbb{R}$ be continuous and nonnegative. If a continuous function $y : [a, b] \to \mathbb{R}$ satisfies*

$$y(t) \leq \bar{y}(t) + \int_a^t \widetilde{y}(s) y(s) ds,$$

for $t \in [a, b]$, then on the same interval it holds that:

$$y(t) \leq \bar{y}(t) + \int_a^t \bar{y}(s) \widetilde{y}(s) \exp\left[\int_s^t \widetilde{y}(\tau) d\tau\right] ds.$$

2.1 Prescribed Performance

Prescribed performance control, recently proposed in [29], describes the behavior where a tracking error $e : \mathbb{R}_{\geq 0} \to \mathbb{R}$ evolves strictly within a predefined region that is bounded by certain functions of time, achieving prescribed transient and steady state performance. The mathematical expression of prescribed performance is given by the following inequalities:

$$-\rho_L(t) < e(t) < \rho_U(t), \quad \forall t \in \mathbb{R}_{\geq 0},$$

where $\rho_L(t), \rho_U(t)$ are smooth and bounded decaying functions of time satisfying $\lim\limits_{t \to \infty} \rho_L(t) > 0$ and $\lim\limits_{t \to \infty} \rho_U(t) > 0$, called performance functions. Specifically, for the exponential performance functions $\rho_i(t) \coloneqq (\rho_{i,0} - \rho_{i,\infty}) \exp(-l_i t) + \rho_{i,\infty}$, with

$\rho_{i,0}, \rho_{i,\infty}, l_i \in \mathbb{R}_{>0}, i \in \{U, L\}$, appropriately chosen constants, the terms $\rho_{L,0} := \rho_L(0), \rho_{U,0} := \rho_U(0)$ are selected such that $\rho_{U,0} > e(0) > \rho_{L,0}$ and the terms $\rho_{L,\infty} := \lim\limits_{t\to\infty} \rho_L(t), \rho_{U,\infty} := \lim\limits_{t\to\infty} \rho_U(t)$ represent the maximum allowable size of the tracking error $e(t)$ at steady state, which may be set arbitrarily small to a value reflecting the resolution of the measurement device, thus achieving practical convergence of $e(t)$ to zero. Moreover, the decreasing rate of $\rho_L(t), \rho_U(t)$, which is affected by the constants l_L, l_U in this case, introduces a lower bound on the required speed of convergence of $e(t)$. Therefore, the appropriate selection of the performance functions $\rho_L(t), \rho_U(t)$ imposes performance characteristics on the tracking error $e(t)$.

2.2 Dynamical Systems

Definition 2.3. ([28]) A continuous function $\alpha : [0, a) \to \mathbb{R}_{\geq 0}$ is said to belong to *class $\mathcal{K}$*, if it is strictly increasing and $\alpha(0) = 0$. It is said to belong to class $\mathcal{K}_\infty$ if $a = \infty$ and $\alpha(r) \to \infty$, as $r \to \infty$.

Definition 2.4. ([28]) A continuous function $\beta : [0, a) \times \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ is said to belong to *class $\mathcal{KL}$*, if:

- For each fixed s, $\beta(r, s) \in \mathcal{K}$ with respect to r.
- For each fixed r, $\beta(r, s)$ is decreasing with respect to s and $\beta(r, s) \to 0$, at $s \to \infty$.

Lemma 2.2. *([30]) Let γ be a continuous, positive definite function and x be an absolutely continuous function on $\mathbb{R}$. If the following holds:*

- $\|x(\cdot)\| < \infty, \|\dot{x}(\cdot)\| < \infty,$
- $\lim\limits_{t\to\infty} \int_0^t \gamma(x(s))ds < \infty.$

Then, $\lim_{t\to\infty} \|x(t)\| = 0$.

Definition 2.5. ([31]) A nonlinear system $\dot{x} = f(x, u)$ with initial condition $x(t_0)$ is said to be *Input to State Stable (ISS)* if there exist functions $\beta \in \mathcal{KL}$ and $\sigma \in \mathcal{K}_\infty$ such that:

$$\|x(t)\| \leq \beta(\|x(t_0)\|, t) + \sigma(\|u\|).$$

Definition 2.6. ([31]) A Lyapunov function $V(x, u)$ for the nonlinear system $\dot{x} = f(x, u)$ with initial condition $x(t_0)$ is said to be *ISS-Lyapunov function* if there exist functions $\alpha, \sigma \in \mathcal{K}_\infty$ such that:

$$\dot{V}(x, u) \leq -\alpha(\|x\|) + \sigma(\|u\|), \forall x, u. \tag{2.1}$$

Theorem 2.1. *([32]) A nonlinear system $\dot{x} = f(x, u)$ with initial condition $x(t_0)$ is said to be ISS if and only if it admits a ISS-Lyapunov function.*

Theorem 2.2. *([28, Appendix C]) Consider the system $\dot{x} = f(x)$ where $f : D \to \mathbb{R}^n$ is piecewise continuous and locally Lipschitz on $D \subseteq \mathbb{R}^n$; D is a domain that contains the origin. Let $V : D \to \mathbb{R}$ be a continuously differentiable function such that $\alpha_1(\|x\|) \leq V(x) \leq \alpha_2(\|x\|)$ and $\dot{V} \leq -w(x)$, $\forall \|x\| \geq \mu > 0$ for every $t \geq 0$ and $x \in \mathcal{D}$, where α_1, α_2 are class $\mathcal{K}$ functions and w_3 is a continuous positive definite function. Take $r > 0$ such that $\mathcal{B}(0, r) \subseteq D$ and suppose that $\mu < \alpha_2^{-1}(\alpha_1(r))$. Then, there exist a class $\mathcal{K}_\infty$ function α_3 and for every initial state $x(t_0)$ satisfying $\|x(t_0)\| \leq \alpha_2^{-1}(\alpha_1(r))$, there exists $T \geq 0$ such that*

$$\begin{aligned} &\|x(t)\| \leq \alpha_3(\|x(t_0)\|), \forall\ t_0 \leq t \leq T, \\ &\|x(t)\| \leq \alpha_1^{-1}(\alpha_2(\mu)), \forall t > T. \end{aligned}$$

Consider the initial value problem:

$$\dot{x} = h(x, t), x(0) \in \Omega, \tag{2.2}$$

with $h : \Omega \times \mathbb{R}_{\geq 0} \to \mathbb{R}^n$ where $\Omega \subset \mathbb{R}^n$ is a non-empty open set.

Definition 2.7. [33] A solution $x(t)$ of the initial value problem (2.2) is maximal if it has no proper right extension that is also a solution of (2.2).

Theorem 2.3. *[33] Consider problem* (2.2)*. Assume that $h(x, t)$ is: a) locally Lipschitz on x for almost all $t \in \mathbb{R}_{\geq 0}$, b) piecewise continuous on t for each fixed $x \in \Omega$ and c) locally integrable on t for each fixed $x \in \Omega$. Then, there exists a maximal solution $x(t)$ of* (2.2) *on $[0, t_{\max})$ with $t_{\max} > 0$ such that $x(t) \in \Omega, \forall t \in [0, t_{\max})$.*

Proposition 2.1. [33] Assume that the hypotheses of Theorem 2.3 hold. For a maximal solution $x(t)$ on the time interval $[0, t_{\max})$ with $t_{\max} < \infty$ and for any compact set $\Omega' \subset \Omega$ there exists a time instant $t' \in [0, t_{\max})$ such that $x(t') \notin \Omega'$.

Theorem 2.4. *[34] Let $\bar{\Omega}$ be an open set in $\mathbb{R}^n \times \mathbb{R}_{\geq 0}$. Consider a function $g : \bar{\Omega} \to \mathbb{R}^n$ that satisfies the following conditions:*

1. *For every $x \in \mathbb{R}^n$, the function $t \to h(x, t)$ defined on $\Omega_x := \{t : (x, t) \in \bar{\Omega}\}$ is measurable. For every $t \in \mathbb{R}_{\geq 0}$, the function $x \to h(x, t)$ defined on $\Omega_t := \{x : (x, t) \in \bar{\Omega}\}$ is continuous.*

2. *For every compact $K \subset \bar{\Omega}$, there exist constants C_K, L_K such that*

$$\|h(x, t)\| \leq C_K, \|h(x, t) - h(y, t)\| \leq L_K \|x - y\|,$$

$\forall (x, t), (y, t) \in K$.

Then the initial value problem (2.2) *with $h : \bar{\Omega} \to \mathbb{R}^n$ and some $(x_0, t_0) \in \bar{\Omega}$, has a unique and maximal solution $x : [t_0, t_{\max}) \to \mathbb{R}^n$, with $t_{\max} > t_0$ and $(x(t), t) \in \bar{\Omega}, \forall t \in [t_0, t_{\max})$.*

Theorem 2.5. *[34] Let the conditions of Theorem 2.4 hold in $\bar{\Omega}$ and let $x(t), t \in [t_0, t_{\max})$ be a maximal solution of the initial value problem (2.2). Then, either $t_{\max} = \infty$ or*

$$\lim_{t \to t_{\max}^-} \left(\|x(t)\| + \frac{1}{d_{\mathcal{S}}((x(t),t), \partial\bar{\Omega})} \right) = \infty,$$

where $d_{\mathcal{S}} : \mathbb{R}^n \times 2^{\mathbb{R}^n}$ is the distance of a point $x \in \mathbb{R}^n$ to a set A, defined as $d_{\mathcal{S}}(x, A) := \inf\limits_{y \in A} \{\|x - y\|\}$.

2.3 Navigation Functions

Navigation functions, initially proposed in [35] for single-point-sized robot navigation, are real-valued maps realized through cost functions, whose negated gradient field is attractive towards the goal configuration (referred to as the good or desirable set) and repulsive with respect to the obstacles set (referred to as the bad set which we want to avoid). We provide here a brief overview of the multi-agent version introduced in [36] and [37], respectively.

2.3.1 Multirobot Navigation Functions (MRNFs)

Consider $N \in \mathbb{N}$ spherical robots, with center $q_i \in \mathbb{R}^n$, $n \in \mathbb{N}$, and radius $r_i \in \mathbb{R}_{>0}$, i.e., $\mathcal{B}_n(q_i, r_i)$, $i \in \mathcal{N}$, operating in an open spherical workspace $\mathcal{W} := \mathring{\mathcal{B}}_n(0, r_0)$ of radius $r_0 \in \mathbb{R}_{>0}$. Each robot has a destination point $q_{\mathrm{d}_i} \in \mathbb{R}^n, i \in \mathcal{N}$, and $q_{\mathrm{d}} := [q_{\mathrm{d}_1}^\top, \dots, q_{\mathrm{d}_N}^\top]^\top$. Let $\mathcal{F} \subset \mathbb{R}^n$ be a compact connected analytic manifold with boundary. A map $\varphi : \mathcal{F} \to [0, 1]$ is a *MRNF* if

1. It is analytic on $\mathcal{F}$,
2. It has only one minimum at $q_{\mathrm{d}} \in \mathring{F}$,
3. Its Hessian at all critical points is full rank,
4. $\lim\limits_{q \to \partial\mathcal{F}} = 1 > \varphi(q'), \forall q' \in \mathring{F}$,

where $q := [q_1^\top, \dots, q_N^\top]^\top \in \mathbb{R}^{Nn}$. The class of MRNFs has the form

$$\varphi(q) = \frac{\gamma(q)}{\left([\gamma(q)]^\kappa + G(q) \right)^{\frac{1}{\kappa}}},$$

where $\gamma(q) := \|q - q_{\mathrm{d}}\|^2$ is the goal function, $G(q)$ is the obstacle function, and κ is a tunable gain; $\gamma^{-1}(0)$ denotes the desirable set and $G^{-1}(0)$ the set we want to avoid. Next we provide the procedure for the construction of the function G. A robot proximity function, a measure for the distance between two robots $i, l \in \mathcal{N}$, is defined as $\beta_{i,l}(q_i, q_l) := \|q_i - q_l\|^2 - (r_i + r_l)^2$, $\forall i, l \in \mathcal{N}, i \neq l$. The term *relation* is used to describe the possible collision schemes that can be defined in a multirobot team, possibly including obstacles. The *set of relations* between the members of the team can be defined as the set of all possible collision schemes between the members of the team. A binary relation is a relation between two robots. Any relation can be expressed as a set of binary relations. A *relation tree* is the set of robot/obstacles that form a linked team. Each relation may consist of more than one relation tree. The number of binary relations in a relation is called *relation level.* Illustrative examples can be found in [36]. A *relation proximity function* (RPF) provides a measure of the distance between the robots involved in a relation. Each relation has its own RPF. A RPF is the sum of the robot proximity functions of a relation. It assumes the value of zero whenever the related robots collide (since the involved robot proximity functions will be zero) and increases with respect to the distance of the related robots. The RPF of relation j at level k is given by $(b_{R_j})_k := \sum\limits_{(i,m)\in(R_j)_k} \beta_{i,m}$, where we omit the arguments q_i, q_k for notational brevity. A *relation verification function* (RVF) is defined as

$$g_{R_j} := (b_{R_j})_k + \lambda \frac{(b_{R_j})_k}{(b_{R_j})_k + (B_{(R_j^C)_k})^{\frac{1}{h}}},$$

where $\lambda, h > 0$, and R_j^C is the complementary to R_j set of relations in the same level k, j is an index number defining the relation in level k, and $B_{R_j^C} := \prod\limits_{m\in R_j^C} b_m$. The RVF serves as an analytic switch, which goes to zero only when the relation it represents is realized. By further introducing the workspace boundary obstacle functions as $G_0 := \prod_{i\in\mathcal{N}} \left\{(r_0 - r_i)^2 - \|q_i\|^2\right\}$, we can define $G := G_0 \prod_{L=1}^{n_L} \prod_{j=1}^{n_{R,L}} (g_{R_j})_L$, where n_L is the number of levels and $n_{R,L}$ the number of relations in level L. It has been proved that, by choosing the parameter κ large enough, the negated gradient field $-\nabla_q \varphi(q)$ leads to the destination configuration q_d, from almost all initial conditions [36].

2.3.2 Decentralized Navigation Functions (DNFs)

Consider now the class of *decentralized navigation functions*, which has the form $\varphi_i : \mathbb{R}^{3N} \to [0,1]$, with $\varphi_i(q) = \dfrac{\gamma_i(q_i) + f_i(G_i)}{(\gamma_i(q_i)^{\lambda_i} + G_i(q))^{1/\kappa_i}}$. The key difference in this case is the term $G_i : \mathbb{R}^{3N} \to \mathbb{R}$ that is associated with the collision avoidance property of agent i with the rest of the team and is based on the inter-agent *decentralized* distance function [37]: $\beta_{ij} : \mathbb{R}^3 \times \mathbb{R}^3 \to \mathbb{R}$ with

$$\beta_{ij}(p_i, p_j) = \begin{cases} \|p_i - p_j\|^2 - (r_i + r_j)^2, & \text{if } j \in \mathcal{N}_i \\ d_{s_i}^2 - (r_i + r_j)^2, & \text{if } j \notin \mathcal{N}_i, \end{cases}$$

that represents the distance between agents i and $j \in \mathcal{N}_i$. The term $f_i : \mathbb{R} \to \mathbb{R}$ is used in order to avoid inter-agent collisions in case one or more agents that take part in a collision scheme are very close to their goals. Note that in that case, the classical form of φ_i would yield values very close to 0, since agent i is very close to its goal, without actively taking part in avoiding potential collisions. The term f_i, therefore, forces agent i to avoid potential collisions. Analytic expressions for G_i and f_i can be found in [37]. With the aforementioned tools, the control law for agent i is $u_i = -k_i \dfrac{\partial \varphi_i(p)}{\partial p_i}$, which, as shown in [37], drives all agents to their goal positions and guarantees inter-agent collision-avoidance.

2.4 Task Specification in LTL

Definition 2.8. A transition system ($\mathcal{TS}$) is a tuple $\mathcal{T} \coloneqq (\Pi, \Pi_0, \rightarrow, \Psi, \mathcal{L})$, where Π is a discrete finite set of states, Π_0 is a discrete finite set of initial states, $\rightarrow \subseteq \Pi \times \Pi$ is a transition relation, Ψ is a discrete set of atomic propositions[1], and $\mathcal{L} : \Pi \rightarrow 2^{\Psi}$ is a labeling function that assigns to each state the atomic propositions that are true in that state.

Definition 2.9. A *run* of a $\mathcal{TS}$ is an infinite sequence

$$r_{\mathcal{TS}} \coloneqq \pi_0, \pi_1, \pi_2, \ldots, \tag{2.3}$$

with $\pi_0 \in \Pi_0$, $\pi_i \in \Pi$, $\forall i \in \mathbb{N}$.

Definition 2.10. A *word* $w_{\mathcal{TS}}$ of a run $r_{\mathcal{TS}}$ is the infinite sequence

$$w_{\mathcal{TS}}(r_{\mathcal{TS}}) = w_0, w_1, w_2, \ldots, \tag{2.4}$$

where $w_i \in 2^{\Psi}, w_i = \mathcal{L}(\pi_i), \forall i \in \mathbb{N} \cup \{0\}$.

We focus on the task specification ϕ given as a Linear Temporal Logic (LTL) formula. The basic ingredients of a LTL formula are a set of atomic propositions Ψ and several boolean and temporal operators. LTL formulas are formed according to the following grammar [38]: $\phi ::= \mathsf{true} \mid a \mid \phi_1 \wedge \phi_2 \mid \neg\phi \mid \bigcirc \phi \mid \phi_1 \cup \phi_2$, where $a \in \Psi$, ϕ_1 and ϕ_2 are LTL formulas and $\bigcirc$, $\cup$ are the *next* and *until* operators, respectively. Definitions of other useful operators like $\square$ (*always*), $\Diamond$ (*eventually*) and $\Rightarrow$ (*implication*) are omitted and can be found at [38]. The semantics of LTL are defined over infinite words over 2^{Ψ}. Intuitively, an atomic proposition $\psi \in \Psi$ is satisfied on a word $w = w_1 w_2 \ldots$ if it holds at its first position w_1, i.e. $\psi \in w_1$. Formula $\bigcirc\phi$ holds true if ϕ is satisfied on the word suffix that begins in the next position w_2, whereas $\phi_1 \cup \phi_2$ states that ϕ_1 has to be true until ϕ_2 becomes true. Finally, $\Diamond\phi$ and $\square\phi$ holds on w eventually and always, respectively. For a full definition of the LTL semantics, the reader is referred to [38].

A LTL formula ϕ over a set of atomic propositions Ψ can be translated to a Büchi Automaton $\mathcal{A}_\phi$ [38]. Then, by calculating the product of the transition system

[1] boolean variables that are either true or false in a given state

$\mathcal{TS} = (\Pi, \Pi_0, \rightarrow, \Psi, \mathcal{L})$ with $\mathcal{A}_\phi$ as $\widetilde{\mathcal{TS}} := \mathcal{TS} \otimes \mathcal{A}_\phi$, we can find the runs of $\mathcal{TS}$ that satisfy the formula ϕ. These runs can then be projected back to $\mathcal{TS}$, providing paths over Π that satisfy ϕ. More details regarding the technique can be found in [38].

2.5 Task Specification in MITL

Definition 2.11. A *Weighted Transition System (WTS)* is a tuple

$$\mathcal{TS} := (\Pi, \Pi_0, \rightarrow, \Psi, \mathcal{L}, \gamma), \tag{2.5}$$

where Π is a discrete finite set of states, $S_0 \subseteq S$ is a discrete finite set of initial states, $\rightarrow \subseteq \Pi \times \Pi$ is a transition relation, Ψ is a finite set of atomic propositions, $\mathcal{L} : \Pi \rightarrow 2^\Psi$ is a labeling function and $\gamma : (\rightarrow) \rightarrow \mathbb{R}_{\geq 0}$ is a map that assigns a positive weight to each transition.

Definition 2.12. [39] The time sequence $t_0 t_1 t_2 \dots$ is an infinite sequence of time values $t_j \in \mathbb{R}_{\geq 0}, \forall j \in \mathbb{N} \cup \{0\}$, satisfying the following constraints:

- Monotonicity: $t_j < t_{j+1}, \forall j \in \mathbb{N}$.
- Progress: $\forall t' \in \mathbb{R}_{\geq 0}, \exists j \geq 1$ such that $t_j \geq t'$.

Definition 2.13. Let Ψ be a finite set of atomic propositions. A timed word w over Ψ is an infinite sequence $w = (w_0, t_0)(w_1, t_1)(w_2, t_2), \dots$, where $w_0 w_1 w_2 \dots$ is an infinite word over 2^Ψ and $t_0 t_1 t_2 \dots$ is a time sequence according to Definition 2.12.

Definition 2.14. A *timed run* of a WTS is an infinite sequence $r_t = (r_0, t_0)(r_1, t_1) \dots$ such that $r_0 \in \Pi_0$, and $r_j \in \Pi, (r_j, r_{j+1}) \in \rightarrow, \forall j \in \mathbb{N}$. The time stamps t_j are inductively defined as

1. $t_1 = 0$,
2. $t_{j+1} = t_j + \gamma(r_j, r_{j+1}), \forall j \in \mathbb{N}$.

The timed run r_t generates a timed word

$$w(r_t) = w_0(r_0), w_1(r_1) \dots = (\mathcal{L}(r_0), t_0)(\mathcal{L}(r_1), t_1) \dots$$

over the set 2^Ψ, where $\mathcal{L}(r_j)$ is the subset of atomic propositions that are true at state r_j at time t_j, $\forall j \in \mathbb{N} \cup \{0\}$.

The syntax of *Metric Interval Temporal Logic (MITL)* over a set of atomic propositions Ψ is defined by the grammar

$$\phi := p \mid \neg\phi \mid \phi_1 \wedge \phi_2 \mid \bigcirc_I \phi \mid \Diamond_I \phi \mid \Box_I \phi \mid \phi_1 \mathcal{U}_I \phi_2, \tag{2.6}$$

where $p \in \Psi$, and $\bigcirc, \Diamond, \Box$ and $\mathcal{U}$ are the next, future, always and until operators, respectively; I is a nonempty time interval in one of the following forms:

$[i_1, i_2], [i_1, i_2), (i_1, i_2],\ (i_1, i_2), [i_1, \infty), (i_1, \infty)$ with $i_1, i_2 \in \mathbb{R}_{\geq 0}, i_2 > i_1$. MITL can be interpreted either in continuous or point-wise semantics. We utilize the latter and interpret MITL formulas over timed runs such as the ones produced by a WTS.

Definition 2.15. [40, 41] Given a run $r_t = (r_0, t_0)(r_1, t_1)\ldots$ of a WTS and a MITL formula ϕ, we define $(r_t, j) \models \phi, j \in \mathbb{N}$ (r_t satisfies ϕ at j) as follows:

$$
\begin{aligned}
(r_t, j) \models & p \Leftrightarrow p \in \mathcal{L}(r_j),\\
(r_t, j) \models & \neg\phi \Leftrightarrow (r_t, j) \not\models \phi\\
(r_t, j) \models & \phi_1 \wedge \phi_2 \Leftrightarrow (r_t, j) \models \phi_1 \text{ and } (r_t, j) \models \phi_2\\
(r_t, j) \models & \bigcirc_I \phi \Leftrightarrow (r_t, j+1) \models \phi \text{ and } t_{j+1} - t_j \in I\\
(r_t, j) \models & \phi_1 \mathcal{U}_I \phi_2 \Leftrightarrow \exists k, j, \text{ with } j \leq k, \text{s.t. } (r_t, k) \models \phi_2, t_k - t_j \in I \text{ and } (r_t, m) \models \phi_1,\\
& \forall m \in \{j, \ldots, k\}
\end{aligned}
$$

Also, $\Diamond_I \phi = \top \mathcal{U}_I \phi$ and $\Box_I \phi = \neg \Diamond_I \neg \phi$. The sequence r_t satisfies ϕ, denoted as $r_t \models \phi$, if and only if $(r_t, 1) \models \phi$.

Chapter 3

Formation Control

The chapter presents a novel control protocol for the formation control of tree graphs in $SE(3)$. The control laws are decentralized, in the sense that each agent uses only local relative information from its neighbors to calculate its control signal as well as robust to modeling (parametric and structural) uncertainties and external disturbances. The proposed methodology guarantees collision avoidance and connectivity maintenance among the initially connected agents. Moreover, certain predefined functions characterize the transient and steady state performance of the closed loop system. Finally, simulation results verify the validity and efficiency of the proposed approach.

3.1 Introduction

During the last decades, decentralized control of networked multi-agent systems has gained a significant amount of attention due to the great variety of its applications, including multi-robot systems, transportation, multi-point surveillance and biological systems. The main focus of multi-agent systems is the design of distributed control protocols in order to achieve global tasks, such as consensus [2–4, 42], and at the same time fulfill certain properties, e.g., network connectivity [6, 10].

A particular multi-agent problem that has been considered in the literature is the formation control problem, where the agents represent robots that aim to form a prescribed geometrical shape, specified by a certain set of desired relative configurations between the agents. The main categories of formation control that have been studied in the related literature are ([7]) position-based control, displacement-based control, distance-based control and orientation-based control. One of the several applications of formation control is the cooperative manipulation of an object [12], where the formation occurs either for the inter-agent distances and angles or the forces that arise at the grasping points. Motivated by this, we address in this chapter the distance- and orientation-based multi-agent formation control problem.

In distance-based formation control, inter-agent distances are actively controlled to achieve a desired formation, dictated by desired inter-agent distances. Each agent is assumed to be able to sense the relative positions of its neighboring agents, without the need of orientation alignment of the local coordinate systems. When orientation alignment is considered as a control design goal, the problem is known as orientation-based (or bearing-based) formation control. The desired formation is then defined by relative inter-agent orientations. The orientation-based control steers the agents to configurations that achieve desired relative orientation angles. In this work, we aim to design a decentralized control protocol such that both distance- and orientation-based formation is achieved.

The literature in distance-based formation control is rich, and is traditionally categorized in single or double integrator agent dynamics and directed or undirected communication topologies (see e.g. [43–58]). Orientation-based formation control has been addressed in [59–62], whereas the authors in [61, 63, 64] have considered the combination of distance- and orientation-based formation.

In most of the aforementioned works in formation control, the two-dimensional case with simple dynamics and point-mass agents has been dominantly considered. In real applications, however, the engineering systems have nonlinear second order dynamics and are usually subject to exogenous disturbances and modeling errors. Another important issue concerns the connectivity maintenance, the collision avoidance between the neighboring agents and the transient and steady state response of the closed loop system, which have not been taken into account in the majority of related woks. Thus, taking all the above into consideration, the design of robust distributed control schemes for the multi-agent formation control problem becomes a challenging task.

In this chapter, we aim at addressing the distance-based formation control problem with orientation alignment for a team of tree-graph communication-based rigid bodies operating in $SE(3)$, with unknown second-order nonlinear dynamics and external disturbances. We propose a purely decentralized control protocol that guarantees distance formation, orientation alignment as well as collision avoidance and connectivity maintenance between initially neighboring agents and in parallel ensures prescribed transient and steady state performance. The prescribed performance control framework has been incorporated in multi-agent systems in [65, 66], where first order dynamics have been considered. Furthermore, the first one only addresses the consensus problem, whereas the latter solves the position based formation control problem, instead of the distance- and orientation-based problem treated here. More specifically, the proposed methodology exhibits the following attributes:

1. It is decentralized, in the sense that each agent computes its own control signal based on its local sensing capabilities, without needing to communicate with the rest of the agents or know the pose of a global coordinate frame.

2. It is robust to bounded external disturbances and uncertainties of the dynamic model, since these are not employed in the control design.

3. It guarantees collision avoidance and connectivity maintenance among the initially connected agents.

4. It guarantees convergence to a feasible formation configuration with predefined transient and steady-state performance from almost all initial conditions. Moreover, in contrast to standard continuous control methodologies on $SO(3)$, it guarantees convergence to the desired formation configuration *arbitrarily fast*, regardless of the distance of the initial system configuration to the undesired unstable equilibrium.

3.2 Preliminaries

3.2.1 Graph Theory

An *undirected graph* $\mathcal{G}$ is a pair $(\mathcal{N}, \mathcal{E})$, where $\mathcal{N}$ is a finite set of $N \in \mathbb{N}$ nodes, representing a team of agents, and $\mathcal{E} \subseteq \{\{i,j\} : i,j \in \mathcal{N}, i \neq j\}$, with $K = |\mathcal{E}|$, is the set of edges that model the communication capabilities between neighboring agents. For each agent, its neighboring set $\mathcal{N}_i$ is defined as $\mathcal{N}_i := \{j \in \mathcal{V} \text{ s.t. } \{i,j\} \in \mathcal{E}\}$.

If there is an edge $\{i,j\} \in \mathcal{E}$, then i, j are called *adjacent*. A *path* of length r from vertex i to vertex j is a sequence of $r+1$ distinct vertices, starting with i and ending with j, such that consecutive vertices are adjacent. For $i = j$, the path is called a cycle. If there is a path between any two vertices of the graph $\mathcal{G}$, then $\mathcal{G}$ is called *connected*. A connected graph is called a tree if it contains no cycles.

Consider an arbitrary orientation of $\mathcal{G}$, which assigns to each edge $\{i,j\} \in \mathcal{E}$ precisely one of the ordered pairs (i,j) or (j,i). When selecting the pair (i,j), we say that i is the tail and j is the head of the edge $\{i,j\}$. By considering a numbering $k \in \mathcal{K} := \{1, \ldots, K\}$ of the graph's edge set, we define the $N \times M$ *incidence matrix* $D(G)$ as it was given in [67].

Lemma 3.1. *[48, Section III] Assume that the graph $\mathcal{G}$ is a connected tree. Then, $D(\mathcal{G})^\top \Delta D(\mathcal{G})$ is positive definite for any positive definite matrix $\Delta \in \mathbb{R}^{N \times N}$.*

Proposition 3.1. Let $f : \mathbb{R}_{\geq 0} \to \mathbb{R}$, with $f(x) := \exp(x)(\exp(x) - 1) - x^2$. Then it holds that $f(x) \geq 0$, $\forall x \in \mathbb{R}_{\geq 0}$.

Proof. It holds that $\frac{\partial f(x)}{\partial x} = 2\exp(2x) - \exp(x) - 2x > 0, \forall x \in \mathbb{R}_{\geq 0}$. Hence, $f(x) \geq f(0) = 0$, $\forall x \in \mathbb{R}_{\geq 0}$. □

Proposition 3.2. [68] Let $R_1, R_2 \in SO(3)$, and $e_R := S^{-1}(R_1^\top R_2 - R_2^\top R_1)$. Then $\|e_R\|^2 := \|R_1 - R_2\|_{\text{F}}^2 \Big(1 - \frac{1}{8}\|R_1 - R_2\|_{\text{F}}^2\Big)$.

Proposition 3.3. Let $R_1, R_2 \in SO(3)$. Then, for the rotation matrix $R_2^\top R_1 \in SO(3)$ it holds that $-1 \leq \text{tr}[R_2^\top R_1] \leq 3$; $\text{tr}[R_2^\top R_1] = 3$ if and only if $R_2^\top R_1 = I_3 \Leftrightarrow R_1 = R_2$; $\text{tr}[R_2^\top R_1] = -1$ when $R_1 = R_2 \exp(\pi \hat{s})$, for every $\hat{s}$ in the unit sphere.

Useful properties of skew symmetric matrices [69]:

- $x^\top S(y) x = 0$;

- $S(Rx) = RS(x)R^\top$;
- $-\frac{1}{2}\text{tr}\left[S(x)S(y)\right] = x^\top y$;
- $\text{tr}\left[AS(x)\right] = \frac{1}{2}\text{tr}\left[S(x)(A - A^\top)\right] = -x^\top S^{-1}(A - A^\top)$,

for every x, $y \in \mathbb{R}^3$, $A \in \mathbb{R}^{3\times 3}$ and $R \in SO(3)$.

3.3 Problem Formulation

3.3.1 System Model

Consider a set of N rigid bodies, with $\mathcal{N} = \{1, 2, \dots, N\}$, $N \geq 2$, operating in a workspace $W \subseteq \mathbb{R}^3$. We consider that each agent occupies a ball $\mathcal{B}(p_i, r_i)$, where $p_i \in \mathbb{R}^3$ is the position of the agent's center of mass with respect to an inertial frame $\mathcal{F}_o$ and $r_i \in \mathbb{R}_{>0}$ is the agent's radius (see Fig. 3.1). We also denote as $R_i \in SO(3)$ the rotation matrix associated with the orientation of the ith rigid body. Moreover, we denote by $v_{i,L} \in \mathbb{R}^3$ and $\omega_i \in \mathbb{R}^3$ the linear and angular velocity of agent i with respect to frame $\mathcal{F}_o$. The vectors p_i are expressed in $\mathcal{F}_o$ coordinates, whereas $v_{i,L}$ and ω_i are expressed with respect to a local frame $\mathcal{F}_i$ centered at each agent's center of mass. The position, though, of $\mathcal{F}_o$, is not required to be known by the agents, as will be shown later. By defining $x_i := (p_i, R_i) \in SE(3)$ and $v_i := [v_{i,L}^\top, \omega_i^\top]^\top \in \mathbb{R}^6$, we model each agent's motion with the 2nd order Newton-Euler dynamics:

$$\dot{x}_i = (R_i v_{i,L}, R_i S(\omega_i)) \in \mathbb{T}_{R_i}, \tag{3.1a}$$

$$u_i = M_i \dot{v}_i + C_i(v_i)v_i + g_i(x_i) + w_i(x_i, v_i, t), \tag{3.1b}$$

where the matrix $M_i \in \mathbb{R}^{6\times 6}$ is the constant positive definite inertia matrix, $C_i : \mathbb{R}^6 \to \mathbb{R}^{6\times 6}$ is the Coriolis matrix, $g_i : SE(3) \to \mathbb{R}^6$ is the body-frame gravity vector, $w_i : \mathbb{R}^6 \times \mathbb{R}_{\geq 0} \to \mathbb{R}^6$ is a bounded vector representing model uncertainties and external disturbances, and $\mathbb{T}_{R_i} = \mathbb{R}^3 \times T_R SO(3)$, as defined in the previous chapter. Finally, $u_i \in \mathbb{R}^6$ is the control input vector representing the 6D body-frame generalized force acting on agent i. The following properties hold for the aforementioned terms:

- The terms $M_i, C_i(\cdot), g_i(\cdot)$ are *unknown*, $C_i(\cdot), g_i(\cdot)$ are continuous and it holds that

  $$0 < \underline{m}_i < \bar{m}_i < \infty \tag{3.2a}$$

  $$\|g_i(x_i)\| \leq \bar{g}_i, \forall x_i \in SE(3), \tag{3.2b}$$

  $\forall i \in \mathcal{N}$, where $\bar{g}_i$ is a finite *unknown* positive constant and $\underline{m}_i := \lambda_{\min}(M_i)$, and $\bar{m}_i := \lambda_{\max}(M_i)$, which are also *uknown*, $\forall i \in \mathcal{N}$.

- The functions $w_i(x_i, v_i, t)$ are assumed to be continuous in $v_i \in \mathbb{R}^6$ and for each fixed $v_i \in \mathbb{R}^6$, the functions $(x_i, t) \to w_i(x_i, v_i, t)$ are assumed to be

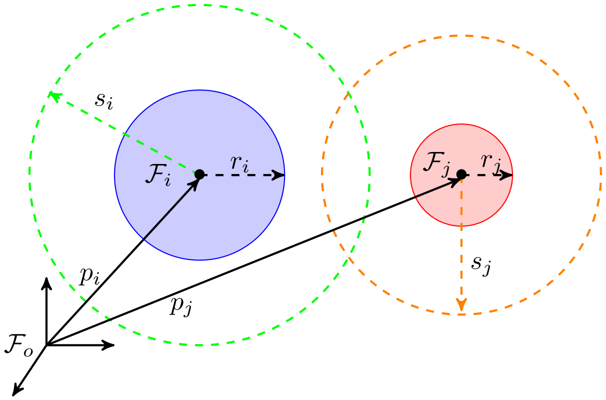


Figure 3.1: Illustration of two agents i, $j \in \mathcal{N}$ in the workspace; $\mathcal{F}_o$ is the inertial frame, $\mathcal{F}_i$, $\mathcal{F}_j$ are the frames attached to the agents' center of mass, p_i, $p_j \in \mathbb{R}^3$ are the positions of the center of mass with respect to $\mathcal{F}_o$; r_i, r_j are the radii of the agents and $s_i > s_j$ are their sensing ranges.

bounded by *unknown* positive finite constants $\bar{w}_i$, i.e., $\|w_i(x_i, v_i, t)\| \le \bar{w}_i < \infty$, $\forall x_i \in SE(3), t \in \mathbb{R}_{\geq 0}$, $i \in \mathcal{N}$.

The dynamics (3.1b) can be written in a vector form representation as:

$$\dot{x} = h_x(x, v), \tag{3.3a}$$
$$u = M\dot{v} + C(v)v + g(x) + w(x, v, t), \tag{3.3b}$$

where $x \coloneqq (x_1, \dots, x_N) \in SE(3)^N$, $v \coloneqq [v_1^\top, \dots, v_N^\top]^\top \in \mathbb{R}^{6N}$, $u \coloneqq [u_1^\top, \dots, u_N^\top]^\top \in \mathbb{R}^{6N}$, and

$$\begin{aligned}
h_x(x, v) &\coloneqq (h_{x_1}(x_1, v_1), \dots, h_{x_N}(x_N, v_N)) \\
&\coloneqq ((R_1 v_{1,L}, R_1 S(\omega_1)), \dots, (R_N v_{N,L}, R_N S(\omega_N))) \\
&\in \mathbb{T}_{R_1} \times \cdots \times \mathbb{T}_{R_N}, \\
M &\coloneqq \text{diag}\{[M_i]_{i \in \mathcal{V}}\} \in \mathbb{R}^{6N \times 6N}, \\
C(v) &\coloneqq \text{diag}\{[C_i(v_i)]_{i \in \mathcal{V}}\} \in \mathbb{R}^{6N \times 6N}, \\
g(x) &\coloneqq [g_1(x_1)^\top, \dots, g_N(x_N)^\top]^\top \in \mathbb{R}^{6N}, \\
w(x, v, t) &\coloneqq [w_1(x_1, v_1, t)^\top, \dots, w(x_N, v_N, t)^\top]^\top \in \mathbb{R}^{6N}.
\end{aligned}$$

It is also further assumed that each agent has a limited sensing range of $s_i > \max_{i,j \in \mathcal{N}}\{r_i + r_j\}$. Therefore, by defining the set-valued neighboring function $\mathcal{N}_i : \mathbb{R}^{3N} \rightrightarrows \mathbb{N}$, with $\mathcal{N}_i(p) \coloneqq \{j \in \mathcal{N} : p_j \in \mathcal{B}(p_i, s_i)\}$, and $p \coloneqq [p_1^\top, \dots, p_N^\top]^\top \in \mathbb{R}^{3N}$, agent i can measure the relative offset $R_i^\top(p_i - p_j)$ (i.e., expressed in i's local frame), the distance $\|p_i - p_j\|$, as well as the relative orientation $R_j^\top R_i$ with respect to its neighbors $j \in \mathcal{N}_i(p)$. In addition, we consider that each agent can measure its own velocity subject to time- and state-varying bounded noise, i.e., agent i has continuous feedback of $\widetilde{v}_i := [\widetilde{v}_{i,L}^\top, \widetilde{\omega}_i{}^\top]^\top := v_i + n_i(x_i, t)$, $\forall i \in \mathcal{N}$. The vector fields $n_i : SE(3) \times \mathbb{R}_{\geq 0} \to \mathbb{R}^6$ are assumed to be bounded by *unknown* positive finite constants $\bar{n}_i$, i.e., $\|n_i(x_i,t)\| \leq \bar{n}_i$, $\forall x_i \in SE(3), t \in \mathbb{R}_{\geq 0}$, $i \in \mathcal{N}$. Moreover, the vector fields $n_{i,d} : SE(3) \times \mathbb{T}_{R_i} \times \mathbb{R}_{\geq 0} \to \mathbb{R}^6$ with $n_{i,d}(x_i, \dot{x}_i, t) := \dot{n}_i(x_i, \dot{x}_i) = \frac{\partial n_i(x_i,t)}{\partial x_i}\dot{x}_i + \frac{\partial n_i(x_i,t)}{\partial t}$ are assumed to be continuous in $\dot{x}_i \in \mathbb{T}_{R_i}$ and for each fixed $\dot{x}_i \in \mathbb{T}_{R_i}$, the functions $(x_i, t) \to n_{i,d}(x_i, \dot{x}_i, t)$ are assumed to be bounded by *unknown* positive finite constants $\bar{n}_{i,d}$, i.e., $\|n_{i,d}(x_i, \dot{x}_i, t)\| \leq \bar{n}_{i,d}$, $\forall x_i \in SE(3), t \in \mathbb{R}_{\geq 0}$, $i \in \mathcal{N}$.

Remark 3.1. (**Local relative feedback**) Note that the agents do not need to have information of any common global inertial frame. The feedback they obtain is relative with respect to their neighboring agents (expressed in their local frames) and they are not required to perform transformations in order to obtain absolute positions/orientations. In the same vein, note also that the velocities v_i are vectors expressed in the agents' local frames.

The topology of the multi-agent network is modeled through the *undirected* graph $\mathcal{G} := (\mathcal{N}, \mathcal{E})$, with $\mathcal{E} = \{(i,j) \in \mathcal{N}^2 : j \in \mathcal{N}_i(p(0)) \text{ and } i \in \mathcal{N}_j(p(0))\}$ (i.e., the initially connected agents), which is assumed to be nonempty and *connected*. We further denote $\mathcal{K} := \{1, \dots, K\}$ where $K := |\mathcal{E}|$. Given the k-th edge, we use the simplified notation (k_1, k_2) for the function that assigns to edge k the respective agents, with $k_1, k_2 \in \mathcal{N}$, $\forall k \in \mathcal{K}$. Since the agents are heterogeneous with respect to their sensing capabilities (different sensing radii s_i), the fact that the initial graph is nonempty, connected and undirected implies that

$$\|p_{k_2}(0) - p_{k_1}(0)\| < d_{k,\text{con}}, \tag{3.4}$$

with $d_{k,\text{con}} := \min\{s_{k_1}, s_{k_2}\}, \forall k \in \mathcal{K}$. We also consider that $\mathcal{G}$ is static in the sense that no edges are added to the graph. We do not exclude, however, edge removal through connectivity losses between initially neighboring agents, which we guarantee to avoid. That is, the proposed methodology guarantees that $\|p_{k_2}(t) - p_{k_1}(t)\| < d_{k,\text{con}}$, $\forall k \in \mathcal{K}$, $\forall t \in \mathbb{R}_{\geq 0}$. It is also assumed that at $t = 0$ the neighboring agents are at a collision-free configuration, i.e., $d_{k,\text{col}} < \|p_{k_2}(0) - p_{k_1}(0)\|, \forall k \in \mathcal{K}$, with $d_{k,\text{col}} := r_{k_1} + r_{k_2}$. Hence, we conclude that

$$d_{k,\text{col}} < \|p_{k_2}(0) - p_{k_1}(0)\| < d_{k,\text{con}}, \forall k \in \mathcal{K}. \tag{3.5}$$

The desired formation is specified by the constants $d_{k,\text{des}} \in \mathbb{R}_{\geq 0}, R_{k,\text{des}} \in SO(3), \forall k \in \mathcal{K}$, for which, the formation configuration is called *feasible* if the set $\Phi := \{x \in SE(3)^N : \|p_{k_2} - p_{k_1}\| = d_{k,\text{des}},\ R_{k_2}^\top R_{k_1} = R_{k,\text{des}}, \forall k \in \mathcal{K}\}$ is nonempty.

3.3.2 Problem Statement

Due to the fact that the agents are not dimensionless and their communication capabilities are limited, the control protocol, except from achieving a desired inter-agent formation and maintaining connectivity, should also guarantee for all $t \in \mathbb{R}_{\geq 0}$

that the initially neighboring agents avoid collision with each other. Therefore, all pairs $(k_1, k_2) \in \mathcal{N}^2$ of agents that initially form an edge must remain within distance greater than $d_{k,\text{col}}$ and less than $d_{k,\text{con}}$. We also make the following assumptions that on the graph topology:

Assumption 3.1. The communication graph $\mathcal{G}$ is a tree.

Formally, the robust formation control problem under the aforementioned constraints is formulated as follows:

Problem 3.1. Given N agents governed by the dynamics (3.1), under Assumption 3.1 and given the desired inter-agent configuration constants $d_{k,\text{des}} \in \mathbb{R}_{\geq 0}$, $R_{k,\text{des}} \in SO(3)$, with $d_{k,\text{col}} < d_{k,\text{des}} < d_{k,\text{con}}$, $\forall k \in \mathcal{K}$, design decentralized control laws $u_i \in \mathbb{R}^6, i \in \mathcal{N}$ such that, $\forall\ k \in \mathcal{K}$, the following hold:

1. $\lim\limits_{t\to\infty} \|p_{k_2}(t) - p_{k_1}(t)\| = d_{k,\text{des}}$;
2. $\lim\limits_{t\to\infty} [R_{k_2}(t)]^\top R_{k_1}(t) = R_{k,\text{des}}$;
3. $d_{k,\text{col}} < \|p_{k_2}(t) - p_{k_1}(t)\| < d_{k,\text{con}}, \forall\ t \in \mathbb{R}_{\geq 0}$.

The term "robust" here refers to robustness of the proposed methodology with respect to the unknown dynamics and external disturbances in (3.1) as well as the unknown noise $n_i(\cdot)$ in the velocity feedback.

3.4 Main Results

3.4.1 Error Derivation

Let us first introduce the distance and orientation errors:

$$e_k := \|p_{k_2} - p_{k_1}\|^2 - d_{k,\text{des}}^2 \ \in \mathbb{R}, \tag{3.6a}$$

$$\psi_k := \frac{1}{2}\text{tr}\Big[I_3 - R_{k,\text{des}}^\top R_{k_2}^\top R_{k_1}\Big] \ \in [0, 2], \tag{3.6b}$$

$\forall k \in \mathcal{K}$, where we have used Proposition 3.3. Regarding e_k, our goal is to guarantee $\lim_{t\to\infty} e_k(t) \to 0$ from all initial conditions satisfying (3.5), while avoiding inter-agent collisions and connectivity losses among the initially connected agents specified by $\mathcal{E}$. Regarding ψ_k, we aim to guarantee the following:

1. $\lim_{t\to\infty} \psi_k(t) \to 0$, which, according to Proposition 3.3 implies that
$$\lim_{t\to\infty} R_{k_2}(t)^\top R_{k_1}(t) = R_{k,\text{des}}$$
2. $\psi_k(t) < 2, \forall t \in \mathbb{R}_{\geq 0}$, since the configuration $\psi_k = 2$ is an undesired equilibrium, as will be clarified later[1].

[1] It has been proved that topological obstructions do not allow global stabilization on $SO(3)$ with a continuous feedback control law (see [68–70])

By using properties of skew-symmetric matrices, we derive the following dynamics of the errors (3.6):

$$\dot{e}_k = 2(p_{k_2} - p_{k_1})^\top (R_{k_2} v_{k_2,L} - R_{k_1} v_{k_1,L}) = 2(R_{k_1}^\top \widetilde{p}_{k_2,k_1})^\top (R_{k_1}^\top R_{k_2} v_{k_2,L} - v_{k_1,L}), \tag{3.7a}$$

$$\dot{\psi}_k = \frac{1}{2} e_{R_k}^\top (R_{k_1}^\top R_{k_2} \omega_{k_2} - \omega_{k_1}), \tag{3.7b}$$

where $\widetilde{p}_{k_2,k_1} := p_{k_2} - p_{k_1}$ and $e_{R_k} := S^{-1}(R_{k_1}^\top R_{k_2} R_{k,\text{des}} - R_{k,\text{des}}^\top R_{k_2}^\top R_{k_1})$, $\forall k \in \mathcal{K}$.

By employing Proposition 3.2, we obtain $\|e_{R_k}\|^2 = \|R_{k_2}^\top R_{k_1} - R_{k,\text{des}}\|_{\text{F}}^2 (1 - \frac{1}{8}\|R_{k_2}^\top R_{k_1} - R_{k,\text{des}}\|_{\text{F}}^2)$ as well as

$$\begin{aligned} \|R_{k_2}^\top R_{k_1} - R_{k,\text{des}}\|_{\text{F}}^2 &= \text{tr}\Big[(R_{k_2}^\top R_{k_1} - R_{k,\text{des}})^\top (R_{k_2}^\top R_{k_1} - R_{k,\text{des}})\Big] \\ &= \text{tr}\left[2I_3 - 2R_{k,\text{des}}^\top R_{k_2}^\top R_{k_1}\right] = 4\psi_k. \end{aligned}$$

Hence, it holds that:

$$\|e_{R_k}\|^2 = 2\psi_k(2 - \psi_k), \tag{3.8}$$

which implies that: $\|e_{R_k}\| = 0 \Rightarrow \psi_k = 0 \ or \ \psi_k = 2$, $\forall k \in \mathcal{M}$. The two configurations $\psi_k = 0$ and $\psi_k = 2$ correspond to the desired and undesired equilibrium, respectively.

3.4.2 Performance Functions

The concepts and techniques of prescribed performance control (see Section 2.1) are adapted in this work in order to: a) achieve predefined transient and steady state response for the distance and orientation errors e_k, ψ_k, $\forall k \in \mathcal{K}$, as well as ii) avoid the violation of the collision and connectivity constraints between initially neighboring agents, as presented in Section 3.3. The mathematical expressions of prescribed performance are given by the inequality objectives:

$$-C_{k,\text{col}} \rho_{e_k}(t) < e_k(t) < C_{k,\text{con}} \rho_{e_k}(t), \tag{3.9a}$$

$$0 \leq \psi_k(t) < \rho_{\psi_k}(t) < 2, \tag{3.9b}$$

$\forall k \in \mathcal{K}$, where $\rho_{e_k} : \mathbb{R}_{\geq 0} \to \left[\frac{\rho_{e_k,\infty}}{\max\{C_{k,\text{con}}, C_{k,\text{col}}\}}, 1\right]$, $\rho_{\psi_k} : \mathbb{R}_{\geq 0} \to [\rho_{\psi_k,\infty}, \rho_{\psi_k,0}]$, with

$$\begin{aligned} \rho_{e_k}(t) &:= \left[1 - \frac{\rho_{e_k,\infty}}{\max\{C_{k,\text{con}}, C_{k,\text{col}}\}}\right] e^{-l_{e_k} t} + \frac{\rho_{e_k,\infty}}{\max\{C_{k,\text{con}}, C_{k,\text{col}}\}}, \\ \rho_{\psi_k}(t) &:= (\rho_{\psi_k,0} - \rho_{\psi_k,\infty}) e^{-l_{\psi_k} t} + \rho_{\psi_k,\infty}, \end{aligned}$$

are designer-specified, smooth, bounded, and decreasing functions of time; the constants l_{e_k}, $l_{\psi_k} \in \mathbb{R}_{\geq 0}$, and $\rho_{e_k,\infty} \in (0, \max\{C_{k,\text{con}}, C_{k,\text{col}}\})$, $\rho_{\psi_k,\infty} \in (0, \rho_{\psi_k,0})$, $\forall k \in \mathcal{K}$, incorporate the desired transient and steady state performance specifications respectively, as presented in Section 2.1, and $C_{k,\text{col}}$, $C_{k,\text{con}} \in \mathbb{R}_{>0}, \forall k \in \mathcal{K}$, are associated with the collision and connectivity constraints. In particular, we select

$$C_{k,\text{col}} := d^2_{k,\text{des}} - d^2_{k,\text{col}}, \tag{3.10a}$$

$$C_{k,\text{con}} := d^2_{k,\text{con}} - d^2_{k,\text{des}}, \tag{3.10b}$$

$\forall k \in \mathcal{K}$, which, since the desired formation is compatible with the collision and connectivity constraints (i.e., $d_{k,\text{col}} < d_{k,\text{des}} < d_{k,\text{con}}, \forall k \in \mathcal{K}$), ensures that $C_{k,\text{col}}, C_{k,\text{con}} \in \mathbb{R}_{>0}, \forall k \in \mathcal{K}$, and consequently, in view of (3.5), that:

$$-C_{k,\text{col}}\rho_{e_k}(0) < e_k(0) < \rho_{e_k}(0)C_{k,\text{con}}, \tag{3.11a}$$

$\forall k \in \mathcal{K}$. Moreover, assuming that $\psi_k(0) < 2$, $\forall k \in \mathcal{K}$, by choosing

$$\rho_{\psi_k,0} := \rho_{\psi_k}(0) \in \Big(\psi_k(0), 2\Big), \tag{3.11b}$$

it is also guaranteed that:

$$0 \leq \psi_k(0) < \rho_{\psi_k}(0) < 2, \tag{3.11c}$$

$\forall k \in \mathcal{K}$. Hence, if we guarantee prescribed performance via (3.9), by setting the steady state constants $\rho_{e_k,\infty}, \rho_{\psi_k,\infty}$ arbitrarily close to zero and by employing the decreasing property of $\rho_{e_k}(t), \rho_{\psi_k}(t), \forall k \in \mathcal{K}$, we guarantee practical convergence of the errors $e_k(t), \psi_k(t)$ to zero and we further obtain:

$$-C_{k,\text{col}} < e_k(t) < C_{k,\text{con}}, \tag{3.12a}$$

$$0 \leq \psi_k(t) < \rho_{\psi_k}(t), \tag{3.12b}$$

$\forall t \in \mathbb{R}_{\geq 0}$, which, owing to (3.10), implies:

$$d_{k,\text{col}} < \|p_{k_2}(t) - p_{k_1}(t)\| < d_{k,\text{con}},$$

$\forall k \in \mathcal{K}, t \in \mathbb{R}_{\geq 0}$, providing, therefore, a solution to problem 3.1. Moreover, note that the choice of $\rho_{\psi_k,0}$ along with (3.12) guarantee that $\psi_k(t) < 2$, $\forall t \in \mathbb{R}_{\geq 0}$ and the avoidance of the unstable singularity equilibrium.

In the sequel, we propose a decentralized control protocol that does not incorporate any information on the agents' dynamic model and guarantees (3.9) for all $t \in \mathbb{R}_{\geq 0}$.

3.4.3 Control Design

Given the errors e_k, ψ_k defined in Section 3.4.1, we perform the following steps:

Step I-a: Select the corresponding functions $\rho_{e_k}(\cdot), \rho_{\psi_k}(\cdot)$ and positive parameters $C_{k,\text{con}}$, $C_{k,\text{col}}$, $k \in \mathcal{K}$, following (3.9), (3.11b), and (3.10), respectively, in order to incorporate the desired transient and steady state performance specifications as well as the collision and connectivity constraints, and define the normalized errors, $\forall k \in \mathcal{K}$,

$$\xi_{e_k} := \frac{e_k}{\rho_{e_k}(t)}, \xi_{\psi_k} := \frac{\psi_k}{\rho_{\psi_k}(t)}. \tag{3.13}$$

Step I-b: Define the transformations $T_{e_k} : (-C_{k,\text{col}}, C_{k,\text{con}}) \to \mathbb{R}$, $k \in \mathcal{K}$, and $T_\psi : [0,1) \to [0,\infty)$ by

$$T_{e_k}(x) := \ln\left(\frac{1 + \frac{x}{C_{k,\text{col}}}}{1 - \frac{x}{C_{k,\text{con}}}}\right), T_\psi(x) := \ln\left(\frac{1}{1-x}\right),$$

$\forall k \in \mathcal{K}$, and the transformed error states, $\forall k \in \mathcal{K}$,

$$\varepsilon_{e_k} := T_{e_k}(\xi_{e_k}), \tag{3.14a}$$

$$\varepsilon_{\psi_k} := T_\psi(\xi_{\psi_k}). \tag{3.14b}$$

Next, we design the decentralized reference velocity vector for each agent $v_{i,\text{des}} := [v_{i,L\text{des}}^\top, \omega_{i,\text{des}}^\top]^\top$ as

$$\begin{aligned} v_{i,\text{des}} &= \begin{bmatrix} v_{i,L\text{des}} \\ \omega_{i,\text{des}} \end{bmatrix} \\ &= -\delta_i \begin{bmatrix} 2 \sum\limits_{k \in \mathcal{M}} \alpha(i,k,R_{k_1},R_{k_2}) \frac{r_{e_k}(\xi_{e_k})}{\rho_{e_k}(t)} \varepsilon_{e_k} R_{k_1}^\top \widetilde{p}_{k_2,k_1} \\ \sum\limits_{k \in \mathcal{K}} \alpha(i,k,R_{k_1},R_{k_2}) \frac{r_\psi(\xi_{\psi_k})}{\rho_{\psi_k}(t)} e_{R_k} \end{bmatrix}, \end{aligned} \tag{3.15}$$

where $\delta_i \in \mathbb{R}_{>0}$ are positive gains, $\forall i \in \mathcal{N}$, $r_{e_k} : (-C_{k,\text{col}}, C_{k,\text{con}}) \to [1,\infty), r_\psi : [0,1) \to [1,\infty)$, with $r_{e_k}(x) := \frac{\partial T_{e_k}(x)}{\partial x}$, $r_\psi(x) := \frac{\partial T_\psi(x)}{\partial x}$, and the function α is defined as $\alpha(i,k,R_{k_1},R_{k_2}) = -I_3$, if i is the tail of the kth edge ($i = k_1$), $\alpha(i,k,R_{k_1},R_{k_2}) = R_{k_2}^\top R_{k_1}$ if i is the head of the kth edge ($i = k_2$), and 0 otherwise. The assignment of the head and tail in each edge can be done off-line according to the specified orientation of the graph, as mentioned in Section 3.2.1.

Step II-a: Define for each agent the velocity errors $e_{v_i} := [e_{v_i,1}^\top, \dots, e_{v_i,6}^\top]^\top := \widetilde{v}_i - v_{i,\text{des}}$, $\forall i \in \mathcal{N}$, and design the decreasing performance functions as $\rho_{v_{i,\ell}} : \mathbb{R}_{\geq 0} \to [\rho_{v_{i,\ell}^0}, \rho_{v_{i,\ell}^\infty}]$, with $\rho_{v_{i,\ell}}(t) := (\rho_{v_{i,\ell}^0} - \rho_{v_{i,\ell}^\infty})\exp(-l_{v_{i,\ell}} t) + \rho_{v_{i,\ell}^\infty}$, where the constants $\rho_{v_{i,\ell}^0}, \rho_{v_{i,\ell}^\infty}, l_{v_{i,\ell}}$ incorporate the desired transient and steady state specifications, with the design constraints $\rho_{v_{i,\ell}^0} > |e_{v_{i,\ell}}(0)|$, $\rho_{v_{i,\ell}^\infty} \in (0, \rho_{v_{i,\ell}^0})$, $\forall \ell \in \{1,\dots,6\}$, $i \in \mathcal{N}$. The term $e_{v_{i,\ell}}(0)$ can be measured be each agent at $t = 0$ directly after the calculation of $v_{i,\text{des}}(0)$.

Moreover, define the normalized velocity errors

$$\xi_{v_i} := \begin{bmatrix} \xi_{v_i,1} \\ \vdots \\ \xi_{v_i,6} \end{bmatrix} := \rho_{v_i}(t)^{-1} e_{v_i}, \tag{3.16}$$

where $\rho_{v_i}(\cdot) := \mathrm{diag}\{[\rho_{v_{i,\ell}}(\cdot)]_{\ell\in\{1,\ldots,6\}}\}$, $\forall i \in \mathcal{N}$.

Step II-b: Define the transformation $T_v : (-1,1) \to \mathbb{R}$ as

$$T_v(x) := \ln\Big(\frac{1+x}{1-x}\Big),$$

and the transformed error states

$$\varepsilon_{v_i} := \begin{bmatrix} \varepsilon_{v_i,1} \\ \vdots \\ \varepsilon_{v_i,6} \end{bmatrix} := \begin{bmatrix} T_v(\xi_{v_i,1}) \\ \vdots \\ T_v(\xi_{v_i,6}) \end{bmatrix}. \tag{3.17}$$

Finally, design the decentralized control protocol for each agent $i \in \mathcal{N}$ as

$$u_i := -\gamma_i \, [\rho_{v_i}(t)]^{-1} \, \bar{r}_v(\xi_{v_i})\varepsilon_{v_i}, \tag{3.18}$$

where $\bar{r}_v(\xi_{v_i}) := \mathrm{diag}\{[r_v(\xi_{v_i,\ell})]_{\ell\in\{1,\ldots,6\}}\}$ with $r_v : (-1,1) \to [1,\infty)$, $r_v(x) := \frac{\partial T_v(x)}{\partial x}$, and $\gamma_i \in \mathbb{R}_{>0}$ are positive gains, $\forall i \in \mathcal{N}$.

Remark 3.2. (**Control protocol intuition**) Note that the selection of $C_{k,\mathrm{col}}, C_{k,\mathrm{con}}$ according to (3.10) and of $\rho_{\psi_k}(t), \rho_{v_i,\ell}(t)$ such that $\rho_{\psi_k,0} = \rho_{\psi_k}(0) \in (\psi_k(0), 2), \rho_{v^0_{i,\ell}} = \rho_{v_{i,\ell}}(0) > |e_{v_{i,\ell}}(0)|$ along with (3.5), guarantee that $\xi_{e_k}(0) \in (C_{k,\mathrm{col}}, C_{k,\mathrm{con}})$, $\psi_k(0) \in [0,2)$, $\xi_{v_{i,\ell}}(0) \in (-1,1)$, $\forall k \in \mathcal{K}$, $\ell \in \{1,\ldots,6\}$, $i \in \mathcal{N}$. The prescribed performance control technique enforces these normalized errors $\xi_{e_k}(t), \xi_{\psi_k}(t)$ and $\xi_{v_i,\ell}(t)$ to remain strictly within the sets $(-C_{k,\mathrm{col}}, C_{k,\mathrm{con}}), [0,2)$, and $(-1,1)$, respectively, $\forall k \in \mathcal{K}, \ell \in \{1,\ldots,6\}, i \in \mathcal{N}, t \geq 0$, guaranteeing thus a solution to Problem 3.1. It can be verified that this can be achieved by maintaining the boundedness of the modulated errors $\varepsilon_{e_k}(t), \varepsilon_{\psi_k}(t)$ and $\varepsilon_{v_i}(t)$ in a compact set, $\forall t \geq 0$.

Remark 3.3. (**Arbitrarily fast convergence to $\psi_k = 0$**) The configurations where $\|e_{R_k}\| = 0 \Leftrightarrow \psi_k = 0$ or $\psi_k = 2$ are equilibrium configurations that result in $\omega_{k_1,\mathrm{des}} = \omega_{k_2,\mathrm{des}} = 0$, $\forall k \in \mathcal{K}$. If $\psi_k(0) = 2$, which is a local minima, the orientation formation specification for edge k cannot be met, since the system becomes uncontrollable. This is an inherent property of stabilization in $SO(3)$, and cannot be resolved with a purely continuous controller [70]. Moreover, initial configurations $\psi_k(0)$ starting arbitrarily close to 2 might take infinitely long to be stabilized at $\psi_k = 0$ with common continuous methodologies [71]. Note however, that the proposed control law guarantees convergence to $\psi_k = 0$ arbitrarily fast, given that $\psi_k(0) < 2$. More specifically, given the initial configuration $\psi_k(0) < 2$, we can always choose $\rho_{\psi_k,0}$ such that $\psi_k(0) < \rho_{\psi_k,0} < 2$, regardless of how close $\psi_k(0)$ is to 2. Then, as proved in the next section, the proposed control algorithm guarantees (3.9b) and the transient and steady state performance of the evolution $\psi_k(t)$ is determined solely by $\rho_{\psi_k}(t)$ and more specifically, the rate of convergence is determined by the term l_{ψ_k}. It can be observed from the desired angular velocities designed $\omega_{i,\mathrm{des}}$ in (3.15) that close to the configuration $\psi_k(0) = 2$, the term $e_{R_k}(0)$, which is close to zero (since $\psi_k(0) = 2 \Rightarrow \|e_{R_k}(0)\| = 0$), is compensated by the term $r_\psi(\xi_{\psi_k}(0)) = \frac{1}{1-\xi_{\psi_k}(0)}$, which attains large values (since $\xi_{\psi_k}(0) = \frac{\psi_k(0)}{\rho_{\psi_k,0}}$ is close to 1). Moreover, potentially large values (but always bounded, as proved in the next section) for $\omega_{i,\text{des}}$ and hence u_i due to the term $r_\psi(\xi_{\psi_k}(0))$ can be compensated by tuning the control gains δ_i and γ_i.

Remark 3.4. (**Decentralized manner, relative feedback, and robustness**) Notice by (3.15) and (3.18) that the proposed control protocols are distributed in the sense that each agent uses only local *relative* information to calculate its own signal. In that respect, regarding every edge k, the parameters $\rho_{e_k,\infty}, \rho_{\psi_k,\infty}, l_{e_k}, l_{\psi_k}$, as well as the sensing radii $s_j, \forall j \in \mathcal{N}_i(p(0))$, which are needed for the calculation of the performance functions $\rho_{e_k}(t), \rho_{\psi_k}(t)$, can be transmitted off-line to the agents $k_1, k_2 \in \mathcal{N}$. In the same vein, regarding $\rho_{v_{i,\ell}}(t)$, i.e., the constants $\rho_{v^\infty_{i,\ell}}, l_{v_{i,\ell}}$ can be transmitted off-line to each agent i, which can also compute $\rho_{v^0_{i,\ell}}$, given the initial velocity errors $e_{v_i}(0)$. Notice also from (3.15) that each agent i uses only relative feedback with respect to its neighbors. In particular, for the calculation of $v_{i,L\text{des}}$, the tail of edge k, i.e., agent k_1, uses feedback of $R^\top_{k_1}(p_{k_2} - p_{k_1})$, and the head of edge k, i.e., agent k_2, uses feedback of $R^\top_{k_2} R_{k_1} R^\top_{k_1}(p_{k_2} - p_{k_1}) = R^\top_{k_2}(p_{k_2} - p_{k_1})$. Both of these terms are the relative inter-agent position difference expressed in the agents' local frames. For the calculation of $\omega_{i,\text{des}}$, agents k_1 and k_2 require feedback of the relative orientation $R^\top_{k_2} R_{k_1}$, as well as the signal $S^{-1}(R^\top_{k_1} R_{k_2} R_{k,\text{des}} - R^\top_{k,\text{des}} R^\top_{k_2} R_{k_1})$, which is a function of $R^\top_{k_2} R_{k_1}$. The aforementioned signals encode information related to the relative pose of each agent with respect to its neighbors, without the need for knowledge of a common global inertial frame. It should also be noted that the proposed control protocol (3.18) depends exclusively on the velocity of each agent and not on the velocity (expressed in a local frame) of its neighbors. Moreover, the proposed control law does not incorporate any prior knowledge of the model nonlinearities/disturbances, enhancing thus its robustness. Finally, the proposed methodology results in a low complexity. Notice that no hard calculations (neither analytic nor numerical) are required to output the proposed control signal.

Remark 3.5. (**Construction of performance functions and gain tuning**) Regarding the construction of the performance functions, we stress that the desired performance specifications concerning the transient and steady state response as well as the collision and connectivity constraints are introduced in the proposed control schemes via $\rho_{e_k}(t), \rho_{\psi_k}(t)$ and $C_{k,\text{col}}, C_{k,\text{con}}$, $k \in \mathcal{K}$. In addition, the velocity performance functions $\rho_{v_{i,\ell}}(t)$, impose prescribed performance on the velocity errors $e_{v_i} = v_i - v_{i,\text{des}}$, $i \in \mathcal{N}$. In this respect, notice that $v_{i,\text{des}}$ acts as a reference signal for the corresponding velocities v_i, $i \in \mathcal{N}$. However, it should be stressed that although such performance specifications are not required (only the neighborhood position and orientation errors need to satisfy predefined transient and steady state performance specifications), their selection affects both the evolution of the errors within the corresponding performance envelopes as well as the control input characteristics (magnitude and rate). More specifically, relaxing the convergence rate and the steady state limit of the velocity performance functions leads to increased oscillatory behavior within the prescribed performance region, which is improved when considering tighter performance functions, enlarging, however, the control effort both in magnitude and rate. Nevertheless, the only hard constraint attached to their definition is related to their initial values. Specifically, $\rho_{\psi_k,0} = \rho_{\psi_k}(0) \in (\psi_k(0), 2), \rho_{v^0_{i,\ell}} = \rho_{v_{i,\ell}}(0) > |e_{v_{i,\ell}}(0)|$, $\forall k \in \mathcal{K}$, $\ell \in \{1, \dots, 6\}$, $i \in \mathcal{N}$. In the same vein, as will be verified by the proof of Theorem 3.1, the actual transient- and steady-state performance of the closed loop system is solely determined by the performance functions $\rho_{e_k}(t)$, $\rho_{\psi_k}(t)$, $\rho_{v_{i,l}}(t)$, and the constants $C_{k,\text{col}}$, $C_{k,\text{con}}$, $k \in \mathcal{K}, \ell \in \{1, \dots, 6\}, i \in \mathcal{N}$, without requiring any tuning of the gains δ_i, γ_i, $i \in \mathcal{N}$. It should be noted, however, that their selection affects the control input characteristics and the state trajectory in the prescribed performance area. In particular, decreasing the gain values leads to increased oscillatory behavior within the prescribed performance area, which is improved when adopting higher values, enlarging, however, the control effort both in magnitude and rate. Fine gain tuning is also needed in cases where the control input needs to be bounded by a pre-specified saturation value, since, although the proposed methodology yields bounded control inputs, it does not guarantee explicit bounds. A detailed analysis regarding the acquirement of such bounds is found in the next chapter.

3.4.4 Stability Analysis

In this section we provide the main result of this paper, which is summarized in the following theorem.

Theorem 3.1. *Consider the multi-agent system described by the dynamics* (3.3)*, under a static tree communication graph $\mathcal{G}$, aiming at establishing a formation described by the desired offsets $d_{k,des} \in (d_{k,col}, d_{k,con})$ and $R_{k,des}$, $\forall k \in \mathcal{K}$, $\forall k \in \mathcal{K}$. Then, the control protocol* (3.13)-(3.18) *guarantees the prescribed transient and steady-state performance*

$$-C_{k,col}\rho_{e_k}(t) < e_k(t) < C_{k,con}\rho_{e_k}(t), \tag{3.19a}$$

$$0 \leq \psi_k(t) < \rho_{\psi_k}(t), \tag{3.19b}$$

$\forall k \in \mathcal{K}$, $t \in \mathbb{R}_{\geq 0}$, under all initial conditions satisfying $\psi_k(0) < 2$, $\forall k \in \mathcal{K}$ and (3.5)*, providing thus a solution to Problem 3.1.*

Proof. We start by defining some vector and matrix forms of the introduced signals and functions:

$$
\begin{aligned}
& e := [e_1, \dots, e_K]^\top, \psi := [\psi_1, \dots, \psi_K]^\top \\
& e_R := [e_{R_1}^\top, \dots, e_{R_K}^\top]^\top, e_v := [e_{v_1}^\top, \dots, e_{v_N}^\top]^\top \\
& \xi_a := [\xi_{a_1}, \dots, \xi_{a_K}]^\top, \xi_v := [\xi_{v_1}^\top, \dots, \xi_{v_N}^\top]^\top \\
& \varepsilon_e := [\varepsilon_{e_1}, \dots, \varepsilon_{e_K}]^\top, \varepsilon_\psi := [\varepsilon_{\psi_1}, \dots, \varepsilon_{\psi_K}]^\top \\
& \varepsilon_v := [\varepsilon_{v_1}^\top, \dots, \varepsilon_{v_N}^\top]^\top, \widetilde{p} := [\widetilde{p}_{1_2,1_1}^\top, \dots, \widetilde{p}_{K_2,K_1}^\top]^\top \\
& v_L := [v_{1,L}^\top, \dots, v_{N,L}^\top]^\top, v_{L\text{des}} := [v_{1,L\text{des}}^\top, \dots, v_{N,L\text{des}}^\top]^\top \\
& \omega := [\omega_1^\top, \dots, \omega_N^\top]^\top, \omega_{\text{des}} := [\omega_{1,\text{des}}^\top, \dots, \omega_{N,\text{des}}^\top]^\top \\
& v_{\text{des}} := [v_{1,\text{des}}^\top, \dots, v_{N,\text{des}}^\top]^\top, \rho_a(t) := \text{diag}\{[\rho_{a_k}(t)]_{k\in\mathcal{K}}\} \\
& \rho_v(t) := \text{diag}\{[\rho_{v_i}(t)]_{i\in\mathcal{N}}\} \\
& r_e(\xi_e) := \text{diag}\{[r_{e_k}(\xi_{e_k})]_{k\in\mathcal{K}}\}, \Sigma_e(\xi_e, t) := r_e(\xi_e)\rho_e(t)^{-1} \\
& \widetilde{r}_\psi(\xi_\psi) := \text{diag}\{[r_\psi(\xi_{\psi_k})]_{k\in\mathcal{K}}\}, \Sigma_\psi(\xi_\psi, t) := \widetilde{r}_\psi(\xi_\psi)\rho_\psi(t)^{-1}, \\
& \widetilde{r}_v(\xi_v) := \text{diag}\{[\bar{r}_v(\xi_{v_i})]_{i\in\mathcal{N}}\}, \Sigma_v(\xi_v, t) := \widetilde{r}_v(\xi_v)\rho_v(t)^{-1}
\end{aligned}
$$

where $a \in \{e, \psi\}$.

With the introduced notation, (3.7) can be written in vector form as:

$$
\begin{aligned}
\dot{e} = \begin{bmatrix} \dot{e}_1 \\ \vdots \\ \dot{e}_K \end{bmatrix} &= \begin{bmatrix} 2(R_{1_1}^\top \widetilde{p}_{1_2,1_1})^\top (R_{1_1}^\top R_{1_2} v_{1_2,L} - v_{1_1,L}) \\ \vdots \\ 2(R_{K_1}^\top \widetilde{p}_{K_2,K_1})^\top (R_{K_1}^\top R_{K_2} v_{K_2,L} - v_{K_1,L}) \end{bmatrix} \\
&= 2 \begin{bmatrix} \widetilde{p}_{1_2,1_1}^\top & \cdots & 0_{1\times 3} \\ \vdots & \ddots & \vdots \\ 0_{1\times 3} & \cdots & \widetilde{p}_{K_2,K_1}^\top \end{bmatrix} \hat{R} D_R(R, \mathcal{G})^\top v_L \\
&= \mathbb{F}_p(\widetilde{p})^\top \hat{R} D_R(R, \mathcal{G})^\top v_L, \qquad (3.20a) \\
\dot{\psi} = \begin{bmatrix} \dot{\psi}_1 \\ \vdots \\ \dot{\psi}_K \end{bmatrix} &= \frac{1}{2} \begin{bmatrix} e_{R_1}^\top (R_{1_1}^\top R_{1_2} \omega_{1_2} - \omega_{1_1}) \\ \vdots \\ e_{R_K}^\top (R_{K_1}^\top R_{K_2} \omega_{1_2} - \omega_{K_1}) \end{bmatrix} \\
&= \frac{1}{2} \begin{bmatrix} e_{R_1}^\top & \cdots & 0_{1\times 3} \\ \vdots & \ddots & \vdots \\ 0_{1\times 3} & \cdots & e_{R_K}^\top \end{bmatrix} D_R(R, \mathcal{G})^\top \omega \\
&= \mathbb{F}_R(e_R)^\top D_R(R, \mathcal{G})^\top \omega, \qquad (3.20b)
\end{aligned}
$$

where $\hat{R} := \mathrm{diag}\{[R_{k_1}]_{k\in\mathcal{K}}\} \in \mathbb{R}^{3K\times 3K}$,

$$\mathbb{F}_p(\widetilde{p}) := 2\begin{bmatrix} \widetilde{p}_{1_2,1_1} & \dots & 0_{3\times 1} \\ \vdots & \ddots & \vdots \\ 0_{3\times 1} & \dots & \widetilde{p}_{K_2,K_1} \end{bmatrix} \in \mathbb{R}^{3K\times K},$$

$$\mathbb{F}_R(e_R) := \frac{1}{2}\begin{bmatrix} e_{R_1} & \dots & 0_{3\times 1} \\ \vdots & \ddots & \vdots \\ 0_{3\times 1} & \dots & e_{R_K} \end{bmatrix} \in \mathbb{R}^{3K\times K},$$

$D_R \in \mathbb{R}^{3N} \times \mathbb{R}^{3K}$ is the *orientation incidence matrix* of the graph:

$$D_R(R,\mathcal{G}) := \bar{R}^\top \left[D(\mathcal{G}) \otimes I_3\right] \hat{R}, \tag{3.21}$$

with $\bar{R} := \mathrm{diag}\{[R_i]_{i\in\mathcal{N}}\} \in \mathbb{R}^{3N\times 3N}$, and $D(\mathcal{G})$ is the incidence matrix of the graph. The terms $\bar{R}$ and $\hat{R}$ in $D_R(R,\mathcal{G})$ correspond to the block diagonal matrix with the agents' rotation matrices along the main block diagonal, and the block diagonal matrix with the rotation matrix of each edge's tail along the main block diagonal, respectively. These two terms have motivated the incorporation of the terms $\alpha(\cdot)$ in the desired velocities $v_{i,\mathrm{des}}$ designed in (3.15), since, as shown next, the vector form v_{des} yields the orientation incidence matrix $D_R(R,\mathcal{G})$.

The desired velocities (3.15) and control inputs (3.18) can be written in vector form as

$$v_{L\mathrm{des}} = -\Delta D_R(R,\mathcal{G})\hat{R}^\top \mathbb{F}_p(\widetilde{p})\Sigma_e(\xi_e,t)\varepsilon_e, \tag{3.22a}$$
$$\omega_{\mathrm{des}} = -\Delta D_R(R,\mathcal{G})\left[\Sigma_\psi(\xi_\psi,t)\otimes I_3\right] e_R, \tag{3.22b}$$
$$u = -\Gamma\, \Sigma_v(\xi_v,t)\varepsilon_v, \tag{3.22c}$$

where $\Delta := \mathrm{diag}\{[\delta_i I_3]_{i\in\mathcal{N}}\} \in \mathbb{R}^{3N\times 3N}$ and $\Gamma := \mathrm{diag}\{[\gamma_i I_6]_{i\in\mathcal{N}}\} \in \mathbb{R}^{6N\times 6N}$. Note from (3.22c) and (3.13), (3.16), (3.14), (3.17) that u can be expressed as a function of the states $u(x,v,t)$. Hence, the closed loop system can be written as

$$\begin{aligned} \dot{x} &= h_x(x,v) \\ \dot{v} &= -M^{-1}\Big\{C(v)v + g(x) + w(x,v,t) - u(x,v,t)\Big\} =: h_v(x,v,t). \end{aligned}$$

By defining $z := (x,v) \in SE(3)^N \times \mathbb{R}^{6N}$, we can write the closed loop system in vector form as

$$\dot{z} = h(z,t) := (h_x(z), h_v(z,t)). \tag{3.23}$$

Next, define the set

$$\begin{aligned}\Omega \coloneqq \{(x,v,t) \in SE(3)^N \times \mathbb{R}^{6N} \times \mathbb{R}_{\geq 0} :\\ \xi_{e_k}(p_{k_1}, p_{k_2}, t) \in (-C_{k,\mathrm{col}}, C_{k,\mathrm{con}}),\\ \xi_{\psi_k}(R_{k_1}, R_{k_2}, t) < 1,\\ \xi_{v_i}(x, v_i, t) \in (-1,1)^6, \forall k \in \mathcal{K}\},\end{aligned}$$

where we have expressed ξ_{e_k}, ξ_{ψ_k}, ξ_{v_i} from (3.13), (3.16) as a function of the states. It can be verified that the set Ω is open due to the continuity of the operators $\xi_{e_k}(\cdot), \xi_{\psi_k}(\cdot), \xi_{v_i}(\cdot)$ and nonempty, due to (3.10). Our goal here is to prove firstly that (3.23) has a unique and maximal solution $(z(t), t)$ in Ω and then that this solution stays in a compact subset of Ω.

It can be verified that the function $h : \Omega \to \mathbb{T}_{R_1} \times \cdots \times \mathbb{T}_{R_N} \times \mathbb{R}^{6N}$ is (a) continuous in t for each fixed $(x,v) \in \{(x,v) \in SE(3)^N \times \mathbb{R}^{6N} : (x,v,t) \in \Omega\}$, and (b) continuous and locally lipschitz in (x,v) for each fixed $t \in \mathbb{R}_{\geq 0}$. Therefore, the conditions of Theorem 2.4 are satisfied and hence, we conclude the existence of a unique and maximal solution of (3.23) for a timed interval $[0, t_{\max})$, with $t_{\max} > 0$, such that $(z(t), t) \in \Omega$, $\forall t \in [0, t_{\max})$. This implies that

$$\xi_{e_k}(t) = \frac{e_k(t)}{\rho_{e_k}(t)} \in (-1,1), \tag{3.24a}$$

$$\xi_{\psi_k}(t) = \frac{\psi_k(t)}{\rho_{\psi_k}(t)} < 1, \tag{3.24b}$$

$$\xi_{v_i}(t) = \rho_{v_i}(t)^{-1} e_{v_i}(t) \in (-1,1)^6, \tag{3.24c}$$

$\forall k \in \mathcal{K}$, $i \in \mathcal{N}$, $t \in [0, t_{\max})$. Therefore, the signals $e_k(t), \psi_k(t), e_{v_i}(t)$ are bounded for all $t \in [0, t_{\max})$. In the following, we aim to show that the solution $(z(t), t)$ is bounded in a compact subset of Ω and hence, by employing Theorem 2.5, that $t_{\max} = \infty$.

Consider the positive definite Lyapunov candidate $V_e : \mathbb{R} \to \mathbb{R}_{\geq 0}$, with $V_e(\varepsilon_e) \coloneqq \frac{1}{2}\|\varepsilon_e\|^2$, which is well defined for $t \in [0, t_{\max})$, due to (3.24a). By differentiating $V_e(\varepsilon_e)$ and taking into account the dynamics $\dot{\xi}_e = \rho_e(t)^{-1}\left[\dot{e} - \dot{\rho}_e(t)\xi_e\right]$, we obtain $\dot{V}_e(\varepsilon_e) = \left[\frac{\partial V_e(\varepsilon_e)}{\partial \varepsilon}\right]^\top \dot{\varepsilon}_e = \varepsilon_e^\top \Sigma_e(\xi_e, t)\left\{\mathbb{F}_p(\widetilde{p})^\top \hat{R} D_R(R, \mathcal{G})^\top v_L - \dot{\rho}_e(t)\xi_e\right\}$, which, by substituting $v_L = \widetilde{v}_L - n_p(x,t) = e_{v_p} + v_{L\mathrm{des}} - n_p(x,t)$ and (3.20), becomes

$$\begin{aligned}\dot{V}_e(\varepsilon_e) =& -\varepsilon_e^\top \Sigma_e(\xi_e, t)\mathbb{F}_p(\widetilde{p})^\top \widetilde{D}(\mathcal{G})\mathbb{F}_p(\widetilde{p})\Sigma_e(\xi_e, t)\varepsilon_e\\ &+ \varepsilon_e^\top \Sigma_e(\xi_e, t)\Big[\mathbb{F}_p(\widetilde{p})^\top \hat{R} D_R(R, \mathcal{G})^\top (e_{v_p} - n_p(x,t)) - \dot{\rho}_e(t)\xi_e\Big],\end{aligned} \tag{3.25}$$

where $\widetilde{D}(\mathcal{G}) \coloneqq \hat{R} D_R(R,\mathcal{G})^\top D_R(R,\mathcal{G})\hat{R}^\top = D(\mathcal{G})^\top \otimes I_3 \; \Delta \; D(\mathcal{G}) \otimes I_3 \in \mathbb{R}^{3K \times 3K}$ (by employing (3.21)), and e_{v_p}, $n_p(x,t)$ are the linear parts of e_v and $n(x,t)$ (i.e., the stack vector of the first three components of every e_{v_i}, $n_i(x_i, t)$), respectively. Note first that, due to (3.24c), the function $e_{v_p}(t)$ is bounded for all $t \in [0, t_{\max})$. Moreover, note that (3.24a) implies that $0 < d_{k,\text{col}} < \|p_{k_1}(t) - p_{k_2}(t)\| < d_{k,\text{con}}$, $\forall t \in [0, t_{\max})$. Therefore, it holds that $\text{rank}(\mathbb{F}_p(\widetilde{p}(t))) = K$, $\forall t \in [0, t_{\max})$. In addition, since G is a connected tree graph and $\delta_i \in \mathbb{R}_{>0}$, $\forall i \in \mathcal{N}$, $\widetilde{D}(\mathcal{G})$ is positive definite and $\text{rank}(\widetilde{D}(\mathcal{G})) = 3K$. Hence, we conclude that $\text{rank}\Big([\mathbb{F}_p(\widetilde{p}(t))]^\top \widetilde{D}(\mathcal{G}) \mathbb{F}_p(\widetilde{p}(t))\Big) = K$ and the positive definiteness of $[\mathbb{F}_p(\widetilde{p}(t))]^\top \widetilde{D}(\mathcal{G}) \mathbb{F}_p(p(t))$, $\forall t \in [0, t_{\max})$. In addition, since $\|p_{k_2}(t) - p_{k_1}(t)\| < d_{k,\text{con}}$, we also conclude that the term $\mathbb{F}_p(\widetilde{p})^\top \hat{R} D_R(R, \mathcal{G})^\top$ is upper bounded, $\forall t \in [0, t_{\max})$. Finally, $\dot{\rho}_e(t)$ and $n_p(x,t)$ are bounded by definition and assumption, respectively, $\forall x \in SE(3)^N, t \in \mathbb{R}_{\geq 0}$. Note that all the aforementioned bounds are independent of $t_{\max}$. We obtain now from (3.25):

$$\begin{aligned} \dot{V}_e(\varepsilon_e) &\leq -\underline{\lambda}_{\widetilde{D}} \|\Sigma_e(\xi_e, t)\varepsilon_e\|^2 + \|\Sigma_e(\xi_e, t)\varepsilon_e\| \bar{B}_e \\ &= -\underline{\lambda}_{\widetilde{D}} \|\Sigma_e(\xi_e, t)\varepsilon_e\| \left[\|\Sigma_e(\xi_e, t)\varepsilon_e\| - \frac{\bar{B}_e}{\underline{\lambda}_{\widetilde{D}}} \right], \end{aligned}$$

$\forall t \in [0, t_{\max})$ where

$$\begin{aligned} \underline{\lambda}_{\widetilde{D}} &:= \inf_{p(t), t \in [t_0, t_{\max})} \left\{ \lambda_{\min}\Big(\mathbb{F}_p(\widetilde{p}(t))^\top \widetilde{D}(\mathcal{G}) \mathbb{F}_p(\widetilde{p}(t))\Big) \right\} \\ &\geq d_{k,\text{col}}^2 \lambda_{\min}(\widetilde{D}(\mathcal{G})) > 0, \end{aligned}$$

and $\bar{B}_e$ is a positive constant, independent of $t_{\max}$, satisfying the following inequality: $\bar{B}_e \geq \|\mathbb{F}_p(\widetilde{p})^\top \hat{R} D_R(R, \mathcal{G})^\top (e_{v_p}(t) - n_p(x,t)) - \dot{\rho}_e(t)\xi_e(t)\|, \forall t \in [0, t_{\max})$. Note that, in view of the aforementioned discussion, $\bar{B}_e$ is finite.

Hence, we conclude that $\dot{V}(\varepsilon_e) < 0 \Leftrightarrow \|\Sigma_e(\xi_e, t)\varepsilon_e\| > \frac{\bar{B}_e}{\underline{\lambda}_{\widetilde{D}}}$. By noting that

$$r_{e_k}(x) = \frac{\partial T_{e_k}(x)}{\partial x} = \frac{\frac{1}{C_{k,\text{col}}} + \frac{1}{C_{k,\text{con}}}}{\left(1 + \frac{x}{C_{k,\text{col}}}\right)\left(1 - \frac{x}{C_{k,\text{con}}}\right)} > \frac{1}{C_{k,\text{col}}} + \frac{1}{C_{k,\text{con}}},$$

$\forall x \in (-C_{k,\text{col}}, C_{k,\text{con}})$, as well as $\rho_{e_k}(t) \leq 1, \forall t \in \mathbb{R}_{\geq 0}$, $k \in \mathcal{K}$, we conclude that $\|\Sigma_e(\xi_e(t), t)\varepsilon_e(t)\| = \sqrt{\sum_{k \in \mathcal{K}} \frac{[r_{e_k}(\xi_{e_k}(t))]^2}{[\rho_{e_k}(t)]^2} [\varepsilon_{e_k}(t)]^2} \geq \bar{C}\|\varepsilon_e(t)\|$, $\forall t \in [0, t_{\max})$, where $\bar{C} := \max\left\{\frac{1}{C_{k,\text{col}}} + \frac{1}{C_{k,\text{con}}}\right\}$. Hence, we conlude that $\dot{V}_e(\varepsilon_e) < 0$, $\forall \|\varepsilon_e\| \geq \frac{\bar{B}_e}{\underline{\lambda}_{\widetilde{D}}\bar{C}}$, $\forall t \in [0, t_{\max})$. Therefore, by invoking Theorem 4.8 in [28] we conclude that

$$\|\varepsilon_e(t)\| \leq \bar{\varepsilon}_e := \max\left\{\varepsilon_e(0), \frac{\bar{B}_e}{\underline{\lambda}_{\widetilde{D}}\bar{C}}\right\}, \tag{3.26}$$

$t \in [0, t_{\max})$, and by taking the inverse logarithm function:

$$-C_{k,\text{col}} < -\underline{\xi}_e \leq \xi_{e_k}(t) \leq \bar{\xi}_e < C_{k,\text{con}}, \tag{3.27}$$

$\forall t \in [0, t_{\max})$, where $\bar{\xi}_e := \frac{\exp(\bar{\varepsilon}_e) - 1}{\exp(\bar{\varepsilon}_e) + 1} C_{k,\text{con}}$, and $\underline{\xi}_e := \frac{\exp(-\bar{\varepsilon}_e) - 1}{\exp(-\bar{\varepsilon}_e) + 1} C_{k,\text{con}}$. Note that $\varepsilon_e(0)$ is finite due to the assumption $d_{k,\text{col}} < \|p_{k_2}(0) - p_{k_1}(0)\| < d_{k,\text{con}}$. Therefore, since $\underline{\lambda}_{\widetilde{D}}$ is strictly positive and $\bar{B}_e$ is also finite, $\bar{\varepsilon}_e$ is well defined. Hence, (3.26) and (3.27) imply the boundedness of $\varepsilon_{e_k}(t)$, $r_{e_k}(\xi_{e_k}(t)$, $\widetilde{p}(t)$, and $p(t)$ in compact sets, $\forall k \in \mathcal{K}$, and therefore, through (3.15), the boundedness of $v_{i,L\text{des}}(t)$, $\forall i \in \mathcal{N}$, $t \in [0, t_{\max})$.

Similarly, consider the positive definite Lyapunov candidate $V_\psi : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$, with $V_\psi(\varepsilon_\psi) = 2\sum_{k\in\mathcal{K}} \varepsilon_{\psi_k}$. By differentiating $V_\psi(\varepsilon_\psi)$ and taking into account the dynamics $\dot{\xi}_{\psi_k}(t) = \rho_{\psi_k}(t)^{-1}\left[\dot{\psi}_k(t) - \dot{\rho}_{\psi_k}(t)\xi_{\psi_k}\right]$, we obtain $\dot{V}_\psi(\varepsilon_\psi) := \left[\frac{\partial V_\psi(\varepsilon_\psi)}{\partial \varepsilon_e}\right]^\top \dot{\varepsilon}_\psi$ $= 2\sum_{k\in\mathcal{K}} \frac{r_\psi(\xi_{\psi_k})}{\rho_{\psi_k}(t)}(\dot{\psi}_k - \dot{\rho}_{\psi_k}\xi_{\psi_k})$, which, after substituting (3.7b), (3.20), becomes

$$\begin{aligned}\dot{V}_\psi(\varepsilon_\psi) &= e_R^\top \left[\Sigma_\psi(\xi_\psi, t) \otimes I_3\right] D_R(R, \mathcal{G})^\top \omega - 2\sum_{k\in\mathcal{K}} \frac{r_\psi(\xi_{\psi_k})}{\rho_{\psi_k}(t)} \dot{\rho}_{\psi_k}(t)\xi_{\psi_k} \\ &= e_R^\top \left[\Sigma_\psi(\xi_\psi, t) \otimes I_3\right] D_R(R, \mathcal{G})^\top \left[\omega_{\text{des}} + e_{v_R} - n_R(x,t)\right] \\ &\quad - 2\sum_{k\in\mathcal{K}} \frac{r_\psi(\xi_{\psi_k})}{\rho_{\psi_k}(t)} \dot{\rho}_{\psi_k}(t)\xi_{\psi_k},\end{aligned}$$

where e_{v_R} and $n_R(x,t)$ are the angular parts of e_v and $n(x,t)$ (i.e., the stack vector of the last three components of every e_{v_i}, $n_i(x,t)$), respectively. By substituting (3.22b) and defining $\widetilde{\Sigma}_\psi(\xi_\psi, t) := \Sigma_\psi(\xi_\psi, t) \otimes I_3 \in \mathbb{R}^{3K\times 3K}$, $\widetilde{D}_R(R, \mathcal{G}) := D_R(R, \mathcal{G})^\top \Delta D_R(R, \mathcal{G}) \in \mathbb{R}^{3K\times 3K}$, we obtain:

$$\begin{aligned}\dot{V}_\psi(\varepsilon_\psi) = &- e_R^\top \widetilde{\Sigma}_\psi(\xi_\psi, t)\widetilde{D}_R(R, \mathcal{G})\widetilde{\Sigma}_\psi(\xi_\psi, t)e_R \\ &+ e_R^\top \widetilde{\Sigma}_\psi(\xi_\psi, t)D_R(R, \mathcal{G})^\top \left[e_{v_R} - n_R(x,t)\right] \\ &- 2\sum_{k\in\mathcal{K}} \frac{r_\psi(\xi_{\psi_k})}{\rho_{\psi_k}(t)} \dot{\rho}_{\psi_k}(t)\xi_{\psi_k}.\end{aligned} \tag{3.28}$$

According to (3.21), $D_R(R, \mathcal{G}) = \bar{R}^\top \left[D(\mathcal{G}) \otimes I_3\right] \hat{R}$. Since $\bar{R}$ and $\hat{R}$ are rotation (and thus unitary) matrices, the singular values of $D_R(R, \mathcal{G})$ are identical to the ones of $D(\mathcal{G})$, and hence $\lambda_{\min}(\widetilde{D}_R(R, \mathcal{G})) = \lambda_{\min}(\widetilde{D}(\mathcal{G})) > 0$. Indeed, let $D(\mathcal{G}) \otimes I_3 = U\Sigma_D V^\top$ be a singular value decomposition of $D(\mathcal{G}) \otimes I_3$, where U, V are unitary matrices, and Σ_D is a diagonal matrix containing the singular values of $D(\mathcal{G}) \otimes I_3$. Then $D_R(R, \mathcal{G}) = \bar{R}^\top U\Sigma_D V^\top \hat{R} = \widetilde{U}\Sigma_D \widetilde{V}^\top$ where $\widetilde{U} := \bar{R}^\top U$, and $\widetilde{V} = \hat{R}^\top V$ are unitary matrices (being products of unitary matrices). Thus, $\widetilde{U}\Sigma_D\widetilde{V}^\top$ is the singular value decomposition of $D_R(R, \mathcal{G})$, and hence its singular values are the diagonal values of Σ_D. By further defining $\beta := [\beta_1^\top, \ldots, \beta_K^\top]^\top := D_R(R, \mathcal{G})^\top(e_{v_R} - n_R(x,t)) \in \mathbb{R}^{3M}$, with $\beta_k \in \mathbb{R}^3$, $\forall k \in \mathcal{K}$, (3.28) becomes

$$\begin{aligned}\dot{V}_\psi(\varepsilon_\psi) \leq & -\lambda_{\min}(\widetilde{D}(\mathcal{G}))\|\widetilde{\Sigma}_\psi(\xi_\psi,t)e_R\|^2 + \sum_{k\in\mathcal{K}} \frac{r_\psi(\xi_{\psi_k})}{\rho_{\psi_k}(t)}(e_{R_k})^\top \beta_k \\ & -2\sum_{k\in\mathcal{K}} \frac{r_\psi(\xi_{\psi_k})}{\rho_{\psi_k}(t)}\dot{\rho}_{\psi_k}(t)\xi_{\psi_k}.\end{aligned}$$

Note that, by construction, $\xi_{\psi_k} \geq 0$, $\forall k \in \mathcal{K}$, and $r_\psi(x) = \frac{\partial T_\psi(x)}{\partial x} = \frac{1}{1-x} > 1, \forall x < 1$. Hence, in view of (3.24b), we conclude that $r_\psi(\xi_{\psi_k}(t)) > 1$, $\forall t \in [0, t_{\max})$. By noting also that $\dot{\rho}_{\psi_k}(t) < 0, \forall t \in \mathbb{R}_{\geq 0}$, $\dot{V}_\psi(\varepsilon_\psi)$ becomes

$$\begin{aligned}\dot{V}_\psi(\varepsilon_\psi) \leq & -\lambda_{\min}(\widetilde{D}(\mathcal{G}))\sum_{k\in\mathcal{K}}\left[\frac{r_\psi(\xi_{\psi_k})}{\rho_{\psi_k}(t)}\right]^2\|e_{R_k}\|^2 + \bar{B}_{\psi_1}\sum_{k\in\mathcal{K}}\frac{r_\psi(\xi_{\psi_k})}{\rho_{\psi_k}(t)}\|e_{R_k}\| \\ & + 2\max_{k\in\mathcal{K}}\{l_{\psi_k}(\rho_{\psi_k,0}-\rho_{\psi_k,\infty})\}\sum_{k\in\mathcal{K}}\frac{r_\psi(\xi_{\psi_k})}{\rho_{\psi_k}(t)}\xi_{\psi_k},\end{aligned}$$

where $\bar{B}_{\psi_1}$ is a positive constant, independent of $t_{\max}$, satisfying the inequality $\bar{B}_{\psi_1} \geq \max_{k\in\mathcal{K}}\{\|\beta_k(t)\|\}$, $\forall t \in [0, t_{\max})$. Note that $\bar{B}_{\psi_1}$ is finite, $\forall t \in [0, t_{\max})$, due to (3.24b) and the boundedness of the noise signals $n(x,t)$. After substituting (3.8), we obtain

$$\begin{aligned}\dot{V}_\psi(\varepsilon_\psi) \leq & -2\lambda_{\min}(\widetilde{D}(\mathcal{G}))\sum_{k\in\mathcal{K}}\left[\frac{r_\psi(\xi_{\psi_k})}{\rho_{\psi_k}(t)}\right]^2\psi_k(2-\psi_k) \\ & + \bar{B}_{\psi_1}\sum_{k\in\mathcal{K}}\frac{r_\psi(\xi_{\psi_k})}{\rho_{\psi_k}(t)}\sqrt{2\psi_k(2-\psi_k)} + 2\max_{k\in\mathcal{K}}\{l_{\psi_k}(\rho_{\psi_k,0}-\rho_{\psi_k,\infty})\}\sum_{k\in\mathcal{K}}\frac{r_\psi(\xi_{\psi_k})}{\rho_{\psi_k}(t)}\xi_{\psi_k}.\end{aligned} \tag{3.29}$$

From (3.24b) we conclude that $0 \leq \psi_k(t) < \rho_{\psi_k}(t) \leq \rho_{\psi_k,0} < 2$, and hence $2-\psi_k(t) \geq 2-\rho_{\psi_k,0} =: \underline{\rho}_k > 0$ $\forall t \in [0, t_{\max})$, $k \in \mathcal{K}$. Moreover, by noticing that $2-\psi_k \leq 2$, $\rho_{\psi_k}(t) \leq \rho_{\psi_k,0}$, and $\psi_k = \xi_{\psi_k}\rho_{\psi_k}(t)$, $\forall k \in \mathcal{K}$, (3.29) becomes

$$\begin{aligned}\dot{V}_\psi(\varepsilon_\psi) \leq & -\widetilde{\mu}\sum_{k\in\mathcal{K}}[r_\psi(\xi_{\psi_k})]^2\xi_{\psi_k} + \frac{2\bar{B}_{\psi_1}}{\max\limits_{k\in\mathcal{K}}\{\sqrt{\rho_{\psi_k,0}}\}}\sum_{k\in\mathcal{K}} r_\psi(\xi_{\psi_k})\sqrt{\xi_{\psi_k}} \\ & + 2\max_{k\in\mathcal{K}}\left\{\frac{l_{\psi_k}(\rho_{\psi_k,0}-\rho_{\psi_k,\infty})}{\rho_{\psi_k,0}}\right\}\sum_{k\in\mathcal{K}} r_\psi(\xi_{\psi_k})\xi_{\psi_k},\end{aligned}$$

where $\widetilde{\mu} := \frac{2\lambda_{\min}(\widetilde{D}(\mathcal{G}))\min_{k\in\mathcal{K}}\{\underline{\rho}_k\}}{\max_{k\in\mathcal{K}}\{\rho_{\psi_k,0}\}}$.

From (3.24b), (3.13), and the fact that $\psi_k \in [0,2]$, it holds that $\xi_{\psi_k}(t) < \sqrt{\xi_{\psi_k}(t)}, \forall k \in \mathcal{K}$. By also employing the property

$$\sum_{k\in\mathcal{K}} r_{\psi_k}(\xi_{\psi_k})\sqrt{\xi_{\psi_k}} \leq \sqrt{K}\sqrt{\sum_{k\in\mathcal{K}}(r_{\psi_k}(\xi_{\psi_k}))^2\xi_{\psi_k}},$$

we obtain

$$\dot{V}_\psi(\varepsilon_\psi) \leq -\sqrt{\sum_{k\in\mathcal{K}} \left[r_\psi(\xi_{\psi_k})\right]^2 \xi_{\psi_k}}\left\{\widetilde{\mu}\sqrt{\sum_{k\in\mathcal{K}} \left[r_{\psi_k}(\xi_{\psi_k})\right]^2 \xi_{\psi_k}} - \bar{B}_\psi\right\},$$

where:

$$\bar{B}_\psi := 2\sqrt{K}\left(\frac{\bar{B}_{\psi_1}}{\max\limits_{k\in\mathcal{K}}\{\sqrt{\rho_{\psi_k,0}}\}} + \max_{k\in\mathcal{K}}\left\{\frac{l_{\psi_k}(\rho_{\psi_k,0}-\rho_{\psi_k,\infty})}{\rho_{\psi_k,0}}\right\}\right).$$

We conclude therefore that $\dot{V}_\psi(\varepsilon_\psi) < 0 \Leftrightarrow \sqrt{\sum_{k\in\mathcal{K}}\left[r_\psi(\xi_{\psi_k})\right]^2 \xi_{\psi_k}} > \frac{\bar{B}_\psi}{\widetilde{\mu}}$. From (3.14b), given $y = T_\psi(x)$, we obtain:

$$\begin{aligned}\left[r_\psi(x)\right]^2 x &= \left[\frac{\partial T(x)}{\partial x}\right]^2 T^{-1}(y) = \frac{1}{(1-x)^2}T^{-1}(y)\\ &= \frac{1}{\left[1-T^{-1}(y)\right]^2}T^{-1}(y) = \exp(y)\left[\exp(y)-1\right],\end{aligned}$$

$\forall x \in [0,1)$. Therefore, $\left[r_\psi(\xi_{\psi_k})\right]^2 \xi_{\psi_k} = \exp(\varepsilon_{\psi_k})\ \left[\exp(\varepsilon_{\psi_k})-1\right]$, and according to Proposition 3.1,

$$\sqrt{\sum_{k\in\mathcal{K}}\left[r_\psi(\xi_{\psi_k})\right]^2 \xi_{\psi_k}} = \sqrt{\sum_{k\in\mathcal{K}}\exp(\varepsilon_{\psi_k})\left[\exp(\varepsilon_{\psi_k})-1\right]} \geq \sqrt{\sum_{k\in\mathcal{K}}\varepsilon^2_{\psi_k}} = \|\varepsilon_\psi\|.$$

Hence, we conclude that $\dot{V}_\psi(\varepsilon_\psi) < 0, \forall \|\varepsilon_\psi\| > \frac{\bar{B}_\psi}{\widetilde{\mu}}$. Therefore,

$$\|\varepsilon_\psi(t)\| \leq \bar{\varepsilon}_\psi := \max\left\{\varepsilon_\psi(0), \frac{\bar{B}_\psi}{\widetilde{\mu}}\right\}, \tag{3.30}$$

and, by taking the inverse logarithm:

$$0 \leq -\underline{\xi}_\psi \leq \xi_{\psi_k}(t) \leq \bar{\xi}_\psi < 1, \tag{3.31}$$

where $\bar{\xi}_\psi := \frac{\exp(\bar{\varepsilon}_\psi)-1}{\exp(\bar{\varepsilon}_\psi)}$ and $\underline{\xi}_\psi := \frac{\exp(-\bar{\varepsilon}_\psi)-1}{\exp(-\bar{\varepsilon}_\psi)}$, $\forall k \in \mathcal{K}$. Note that $\bar{B}_\psi$ as well as $\varepsilon_\psi(0)$ are finite, due to the choice $\psi_k(0) < \rho_{\psi_k}(0) < 2$, $\forall k \in \mathcal{K}$. Hence, since $\widetilde{\mu}$ is strictly positive, $\bar{\varepsilon}_\psi$ is also finite. Therefore, we conclude the boundedness of $\varepsilon_{\psi_k}, r_{\psi_k}(\xi_{\psi_k}(t))$, $e_v(t)$ in compact sets, $\forall k \in \mathcal{K}$, and therefore, through (3.15), the boundedness of $\omega_{i,\text{des}}(t)$, $\forall i \in \mathcal{N}, t \in [0, t_{\max})$. From the proven boundedness of $p(t)$ and $p_{i,\text{des}}(t)$, we also conclude the boundedness of $n(x(t),t)$ and invoking $\widetilde{v} = v + n(x,t) = e_v(t) - v_{\text{des}}(t)$ and (3.24c), the boundedness of $v(t)$ and $\dot{x}(t)$, $\forall t \in [0, t_{\max})$. Moreover, in view of (3.26), (3.27), (3.23), (3.15), we also conclude the boundedness of $\dot{v}_{\text{des}}(t)$.

Proceeding along similar lines, we consider the positive definite Lyapunov candidate $V_v : \mathbb{R} \to \mathbb{R}_{\geq 0}$ with $V_v(\varepsilon_v) = \frac{1}{2}\varepsilon_v^\top \Gamma \varepsilon_v$. By computing $\dot{V}_v(\varepsilon_v) = \left[\frac{\partial V_v(\varepsilon_v)}{\partial \varepsilon_v}\right]^\top \dot{\varepsilon}_v$ and using the dynamics $\dot{\xi}_v = \rho_v(t)^{-1}(\dot{e}_v(t) - \dot{\rho}_v(t)\xi_v)$, we obtain

$$\begin{aligned}\dot{V}_v(\varepsilon_v) &= \varepsilon_v^\top \Gamma \Sigma_v(\xi_v, t)\left[\dot{v} + \dot{n}(x,t)\right] - \varepsilon_v^\top \Gamma \Sigma_v(\xi_v, t)\dot{v}_{\text{des}} - \varepsilon_v^\top \Gamma \Sigma_v(\xi_v, t)\dot{\rho}_v(t)\xi_v \\ &= -\varepsilon_v^\top \Sigma_v(\xi_v, t)\Gamma M(x)^{-1}\Gamma \Sigma_v(\xi_v, t)\varepsilon_v \\ &- \varepsilon_v^\top \Sigma_v(\xi_v, t)\Big\{\Gamma M(x)^{-1}\Big[C(v)v + g(x) + w(x,v,t)\Big] - \dot{n}(x,t) + \dot{v}_{\text{des}} + \dot{\rho}_v(t)\xi_v\Big\}.\end{aligned} \tag{3.32}$$

Since we have proved the boundedness of $v(t)$ and $\dot{x}$, $\forall t \in [0, t_{\max})$ the terms $C(v)v$, $\dot{n}(x,t)$, and $w(x,v,t)$ are also bounded, $t \in [0, t_{\max})$, due to the continuities of $C(\cdot)$, $w(\cdot)$, and $\dot{n}(\cdot)$ in v, $\dot{x}$ and the boundedness of $w(\cdot)$ and $\dot{n}()$ in x, t. Moreover, $g(x)$, $\xi_v(t)$, and $\dot{\rho}_v(t)$ are also bounded due to (3.2b), (3.24c), and by construction, respectively. By also using (3.2a), we obtain from (3.32):

$$\dot{V}_v(\varepsilon_v) \leq -\underline{\lambda}_K \|\Sigma_v(\xi_v, t)\varepsilon_v\|^2 + \|\Sigma_v(\xi_v, t)\varepsilon_v\|\bar{B}_v,$$

where $\bar{B}_v$ is a positive term, independent of $t_{\max}$, satisfying $\bar{B}_v \geq \left\| \frac{\max_{i\in\mathcal{N}}\{\gamma_i\}}{\min_{i\in\mathcal{N}}\{\underline{m}_i\}} \Big[C(v)v + g(x) + w(x,v,t)\Big] - \dot{n}(x,t) + \dot{v}_{\text{des}}(t) + \dot{\rho}_v(t)\xi_v(t) \right\|$, and $\underline{\lambda}_K := \frac{\min_{i\in\mathcal{N}}\{\gamma_i\}^2}{\max_{i\in\mathcal{N}}\{\bar{m}\}} > 0$. Hence, $\dot{V}_v(\varepsilon_v) < 0 \Leftrightarrow \|\Sigma_v(\xi_v, t)\varepsilon_v\| > \frac{\bar{B}_v}{\underline{\lambda}_K}$. By noting that

$$r_v(x) = \frac{\partial T_v(x)}{\partial x} = \frac{2}{(1+x)(1-x)} > 2 > 1, \tag{3.33}$$

$\forall x \in (-1, 1)$, as well as $\rho_{v_i,\ell}(t) \leq \rho_{v_i^0,\ell}$, $\forall \ell \in \{1,\dots,6\}, t \in \mathbb{R}_{\geq 0}$, we conclude that $\|\Sigma_v(\xi_v(t), t)\varepsilon_v(t)\| = \sqrt{\sum_{i\in\mathcal{N}} \sum_{\ell\in\{1,\dots,6\}} \frac{[r_v(\xi_{v_i,\ell}(t))]^2}{[\rho_{v_{i,\ell}}(t)]^2}[\varepsilon_{v_{i,\ell}}(t)]^2} \geq \frac{1}{\widetilde{\rho}}\|\varepsilon_v(t)\|$, $\forall t \in [0, t_{\max})$, where $\widetilde{\rho} := \max\limits_{\substack{i\in\mathcal{N} \\ m\in\{1,\dots,6\}}} \{\rho_{v^0_{i,m}}\}$. Hence, we conclude that $\dot{V}_v(\varepsilon_v) < 0, \forall \|\varepsilon_v\| \geq \frac{\widetilde{\rho}\bar{B}_v}{\underline{\lambda}_K}, \forall t \in [0, t_{\max})$, and consequently that

$$\|\varepsilon_v(t)\| \leq \bar{\varepsilon}_v := \max\left\{\varepsilon_v(0), \frac{\widetilde{\rho}\bar{B}_v}{\underline{\lambda}_K}\frac{\max\limits_{i\in\mathcal{N}}\{\gamma_i\}}{\min\limits_{i\in\mathcal{N}}\{\gamma_i\}}\right\}, \tag{3.34}$$

$\forall t \in [0, t_{\max})$ and by taking the inverse logarithm function:

$$-1 < -\bar{\xi}_v \leq \xi_{v_{i,\ell}}(t) \leq \bar{\xi}_v < 1, \tag{3.35}$$

$\forall \ell \in \{1,\dots,6\}$, $t \in [0, t_{\max})$ where $\bar{\xi}_v := \frac{\exp(\varepsilon_v)-1}{\exp(\varepsilon_v)+1} = -\frac{\exp(-\varepsilon_v)-1}{\exp(-\varepsilon_v)+1}$. Note that the term $\bar{B}_v$ is finite, $\forall t \in [0, t_{\max})$. Moreover, the term $\varepsilon_v(0)$ is finite due to the choice

$\rho_{v^0_{i,\ell}} > |e_{v_i,\ell}(0)|, \forall \ell \in \{1,\ldots,6\}, i \in \mathcal{N}$. Hence, since $\underline{\lambda}_K$ is strictly positive, the term $\bar{\varepsilon}_v$ is also finite. Thus, the terms $e_v(t)$, $\widetilde{r}_v(\xi_v(t))$ and hence the control laws (3.18) are also bounded in compact sets for all $t \in [0, t_{\max})$.

What remains to be shown is that $t_{\max} = \infty$. Towards that end, suppose that $t_{\max}$ is finite, i.e., $t_{\max} < \infty$. Then, according to Theorem 2.5, it holds that $\lim\limits_{t \to t_{\max}^-} \left(\|z(t)\| + \frac{1}{d_{\mathcal{S}}((z(t),t),\partial\Omega)} \right) = \infty$. We first rewrite the condition in a more explicit form, in order to account for the matrix tuple $R \in SO(3)^N$. We define $z_{p,v} := [p^\top, v^\top]^\top \in \mathbb{R}^{3N} \times \mathbb{R}^{6N}$, the projection sets $\Omega_R := \{(R,t) \in SO(3)^N \times \mathbb{R}_{\geq 0} : (x,v,t) \in \Omega\}$ and $\Omega_{p,v} := \{(p,v,t) \in \mathbb{R}^{3N} \times \mathbb{R}^{6N} \times \mathbb{R}_{\geq 0} : (x,v,t) \in \Omega\}$ as well as the distance from a set $A \subset SO(3)^N \times \mathbb{R}_{\geq 0}$ as $d_{\mathcal{S},SO(3)} : SO(3)^N \times \mathbb{R}_{\geq 0} \times 2^{SO(3)^N \times \mathbb{R}_{\geq 0}} \to \mathbb{R}_{\geq 0}$ with $d_{\mathcal{S},SO(3)}((R,t),A) := \inf\limits_{(R_A,t_A) \in A} \{\|R - R_A\|_T + t - t_A\}$, where $\|\cdot\|_T$ is the induced norm in $SO(3)^N$ as defined in the previous chapter. Therefore, the condition of Theorem 2.5 can now be stated as follows: Since $t_{\max} < \infty$, it holds that

$$L := \lim_{t \to t_{\max}^-} \Bigg(\|p(t)\| + \|v(t)\| + \|R(t)\|_T + \\ \frac{1}{d_{\mathcal{S}}((z_{p,v}(t),t),\partial\Omega_{p,v}) + d_{\mathcal{S},SO(3)}((R(t),t),\partial\Omega_R)} \Bigg) = \infty \tag{3.36}$$

which we aim to prove that is a contradiction. Firstly, it holds that $\|R(t)\|_T = \sum_{i \in \mathcal{N}} \|R_i(t)\|_F \leq N \sup_{t \in [0,t_{\max})} \{\max_{i \in \mathcal{N}\{R_i(t)\}}\}$. However, according to Proposition 3.3, it holds that $-1 \leq \mathrm{tr}(R) \leq 3$ for any $R \in SO(3)$. Hence, $\|R(t)\|_T \leq 3N, \forall t \in [0, t_{\max}]$. Moreover, from (3.35) and (3.16) we obtain $\|e_v(t)\| \leq \sqrt{6}\bar{\xi}_v \widetilde{\rho}$, $\forall t \in [0, t_{\max})$. By invoking (3.26), (3.30), we can also conclude that there exists a finite $\bar{v}_{\mathrm{des}}$ such that $\|v_{\mathrm{des}}(t)\| \leq \bar{v}_{\mathrm{des}}$, $\forall t \in [0, t_{\max})$. Hence, since $\|n_i(x_i,t)\| \leq \bar{n}_i$, $\forall x_i \in SE(3), t \in \mathbb{R}_{\geq 0}, i \in \mathcal{N}$, $v = \widetilde{v} - n(x,t) = e_v + v_{\mathrm{des}} - n(x,t)$ implies that there exists a finite $\bar{v}$ such that $\|v(t)\| \leq \bar{v}$, $\forall t \in [0, t_{\max})$. Hence, $\|p(t)\| = \| \int_0^{t_{\max}} \bar{R}(s)v(s)ds\| \leq \int_0^{t_{\max}} \|\bar{R}(s)v(s)\|ds = \int_0^{t_{\max}} \|v(s)\|ds \leq \int_0^{t_{\max}} \bar{v}ds \Rightarrow \|p(t)\| \leq t_{\max}\bar{v}$, $\forall t \in [0, t_{\max})$, which proves the boundedness of $\|p(t)\|$, since $t_{\max} < \infty$.

Next, note that $\partial\Omega_{p,v} = \{(p,v,t) \in \mathbb{R}^{3N} \times \mathbb{R}^{6N} \times \mathbb{R}_{\geq 0} : (\exists k \in \mathcal{K} : \xi_{e_k}(p_{k_1}, p_{k_2}, t) = -C_{k,\mathrm{col}}$ or $\xi_{e_k}(p_{k_1}, p_{k_2}, t) = C_{k,\mathrm{con}})$ or $(\exists i \in \mathcal{N}, \ell \in \{1,\ldots,6\} : \xi_{v_{i,\ell}}(x, v_i, t) = -1$ or $\xi_{v_{i,\ell}}(x, v_i, t) = 1)\}$ and $\partial\Omega_R = \{(R,t) \in SO(3)^N \times \mathbb{R}_{\geq 0} : \exists k \in \mathcal{K} : \xi_{\psi_k}(R_{k_1}, R_{k_2}, t) = 1\}$. We have proved, however, from (3.27), (3.31), and (3.35) that the maximal solution satisfies the strict inequalities $-C_{k,\mathrm{col}} < -\underline{\xi}_e \leq \xi_{e_k}(p_{k_1}(t), p_{k_2}(t), t) \leq \bar{\xi}_e < C_{k,\mathrm{con}}$, $\xi_{\psi_k}(R_{k_1}(t), R_{k_2}(t), t) \leq \bar{\xi}_\psi < 1$, and $|\xi_{v_{i,\ell}}(x(t), v_i(t), t)| \leq \bar{\xi}_v < 1$, $\forall k \in \mathcal{K}$, $\ell \in \{1,\ldots,6\}$, $i \in \mathcal{N}$, $t \in [0, t_{\max})$. Therefore, we conclude that there exist strictly positive constants $\epsilon_{p,v}$, $\epsilon_R \in \mathbb{R}_{>0}$ such that $d_{\mathcal{S}}((z_{p,v}(t),t),\partial\Omega_{p,v}) \geq \epsilon_{p,v}$ and $d_{\mathcal{S},SO(3)}((R(t),t),\partial\Omega_R) \geq \epsilon_R$, $\forall t \in [0, t_{\max})$. Therefore, we have proved that

$$L \leq (t_{\max} + 1)\bar{v} + 3N + \frac{1}{\epsilon_{p,v} + \epsilon_R} < \infty,$$

since $t_{\max}$ is finite. This contradicts (3.36) and hence, we conclude that $t_{\max} = \infty$.

We have proved the containment of the errors $e_k(t)$, $\psi_k(t)$ in the domain defined by the prescribed performance funnels:

$$\begin{aligned} -C_{k,\text{col}}\rho_{e_k}(t) < e_k(t) < C_{k,\text{con}}\rho_{e_k}(t), \\ 0 \leq \psi_k(t) < \rho_{\psi_k}(t), \end{aligned}$$

$\forall k \in \mathcal{K}$, $t \in \mathbb{R}_{\geq 0}$, which also implies that

$$\begin{aligned} d_{k,\text{col}} <& \|p_{k_1}(t) - p_{k_2}(t)\| < d_{k,\text{con}}, \\ 0 \leq& \psi_k(t) < 2, \end{aligned}$$

$\forall k \in \mathcal{K}$, $t \in \mathbb{R}_{\geq 0}$, i.e., avoidance of the singularity $\psi_k = 2$ and satisfaction of the collision and connectivity constraints for the initially connected edge set $\mathcal{E}$. □

Remark 3.6. [**Prescribed performance**] We can deduce from the aforementioned proof that the proposed control scheme achieves its goals without resorting to the need of rendering $\bar{\varepsilon}_e$, $\bar{\varepsilon}_\psi$, $\bar{\varepsilon}_v$ by adopting extreme values of the control gains δ_i, γ_i. Notice that (3.26), (3.30), and (3.34) hold no matter how large the finite bounds $\bar{\varepsilon}_e$, $\bar{\varepsilon}_\psi$, $\bar{\varepsilon}_v$ are. Hence, the actual performance of the system is determined solely by the performance functions $\rho_e(t), \rho_\psi(t), \rho_v(t)$ and the parameters $C_{k,\text{col}}, C_{k,\text{con}}$, as mentioned in Remark 3.5.

3.5 Simulation Results

We considered $N = 4$ spherical agents with $\mathcal{N} = \{1, 2, 3, 4\}$ and dynamics of the form (3.1), with $r_i = 1$m and $s_i = 4$m, $i \in \{1, \dots, 4\}$. We selected the exogenous disturbances and measurement noise as $w_i = A_{w_i} \sin(\omega_{w,i}t)\dot{x}_i$, and $n_i = A_{n_i} \sin(\omega_{n,i}t)\dot{x}_i$, where the parameters $A_{w_i}, A_{n_i}, \omega_{w,i}, \omega_{n,i}$ as well as the dynamic parameters (mass and moment of inertia) of the agents were randomly chosen in $[0, 1]$, $\forall i \in \mathcal{N}$. The initial conditions were taken as: $p_1(0) = [0, 0, 0]^\top$ m, $p_2(0) = [2, 2, 2]^\top$ m, $p_3(0) = [2, 4, 4]^\top$ m, $p_4(0) = [2, 3, 3]^\top$ m, $R_1(0) = R_3(0) = R_4(0) = I_3$ and

$$R_2(0) = \begin{bmatrix} -0.3624 & 0.0000 & 0.9320 \\ 0.6591 & 0.7071 & 0.2562 \\ -0.6591 & 0.7071 & -0.2562 \end{bmatrix},$$

$v_1(0) = v_2(0 = v_3(0) = v(4) = 0_{6\times 1}$, which give the edge set $\mathcal{E} = \{\{1, 2\}, \{2, 3\}, \{2, 4\}\}$ and the incidence matrix:

$$D(\mathcal{G}) = \begin{bmatrix} -1 & 0 & 0 \\ 1 & -1 & -1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

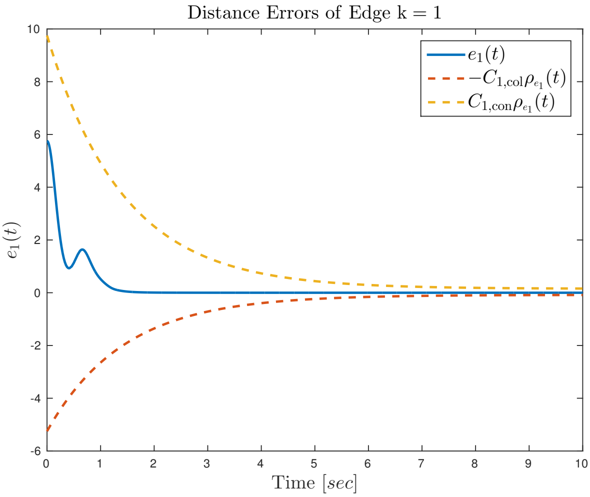


Figure 3.2: The distance error signal of the edge $(1, 2)$.

The desired graph formation was defined by the constants $d_{k,\text{des}} = 2.5\text{m}$,

$$R_{k,\text{des}} = \begin{bmatrix} 0.5000 & -0.8660 & 0.0000 \\ 0.6124 & 0.3536 & -0.7071 \\ 0.6124 & 0.3536 & 0.7071 \end{bmatrix}, \forall k \in \{1, 2, 3\}.$$

The definitions of $d_{k,\text{col}}$, $d_{k,\text{con}}$ yield: $d_{k,\text{col}} = 2$ and $d_{k,\text{con}} = 4$. Invoking (3.10), we have $C_{k,\text{col}} = 2.25$ and $C_{k,\text{con}} = 9.75$. Moreover, the parameters of the performance functions were chosen as $\rho_{e_k,\infty} = \rho_{\psi_k,\infty} = 0.1$, $\rho_{\psi_k,0} = 1.99 > \max\{\rho_{\psi_1}(0), \rho_{\psi_2}(0), \rho_{\psi_3}(0)\}$ and $l_{e_k} = l_{\psi_k} = 0.7$. In addition, we chose $\rho_{v^0_{i,\ell}} = 2|e_{v_i,\ell}(0)| + 0.5$, $l_{v_{i_\ell}} = 1.55$ and $\rho_{v^\infty_{i,\ell}} = 0.15$, for every $i \in \{1, \ldots, 4\}$, $\ell \in \{1, \ldots, 6\}$. Finally, the control gains were set to $\Gamma = 10 I_{24}$ and $\Delta = I_{24}$.

The simulation results are shown in Fig.3.2-3.12. In particular, Fig. 3.2-3.4 and Fig. 3.5-3.7 show the distance error signals and the orientation error signals, respectively. All the errors remain within the predefined bounds and converge to 0. Fig. 3.8 shows the distance between the agents. The connectivity is maintained for all times as well as the agents do not collide with each other. Finally, Fig. 3.9-Fig. 3.12 depict the control input signals of the agents which remain bounded for all times.

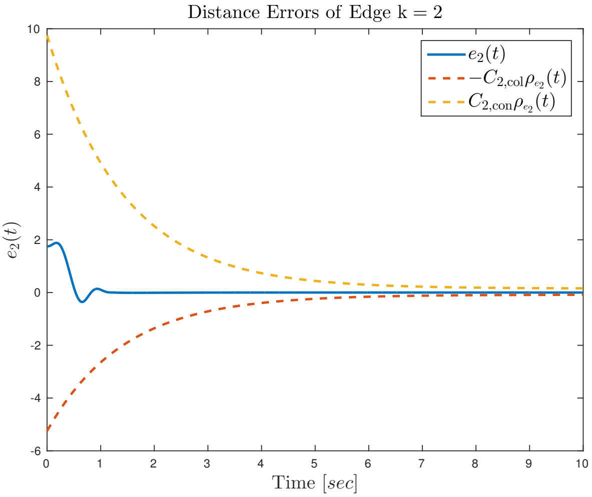


Figure 3.3: The distance error signal of the edge $(2, 3)$.

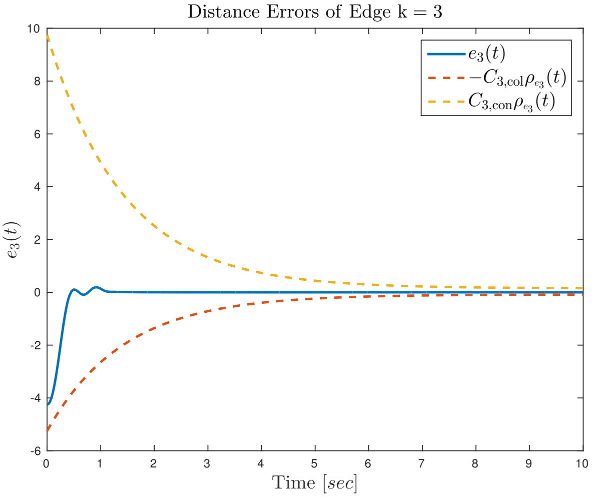


Figure 3.4: The distance error signal of the edge $(2, 4)$.

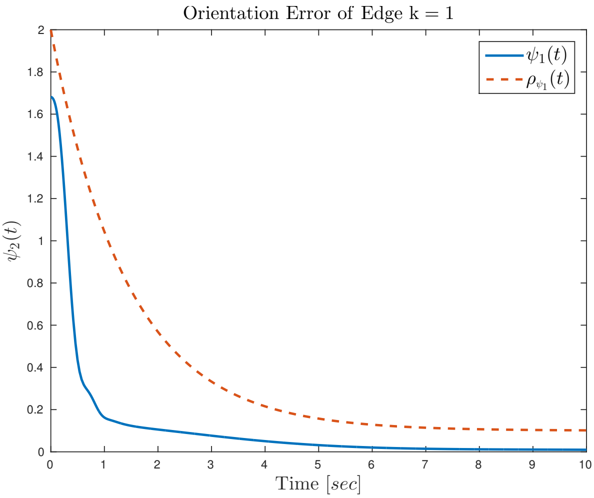


Figure 3.5: The orientation error signal of the edge $(1, 2)$.

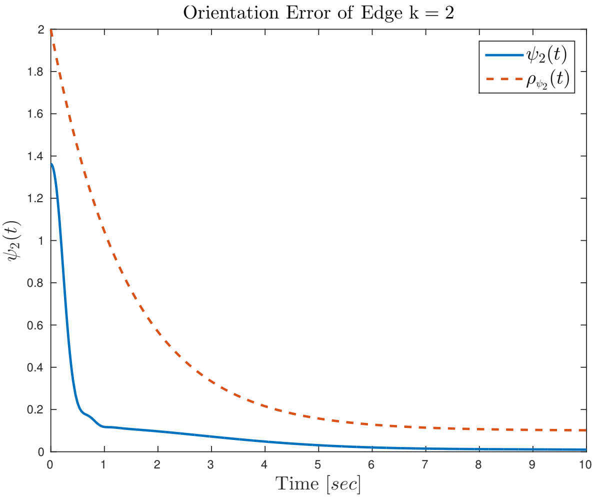


Figure 3.6: The orientation error signal of the edge $(2, 3)$.

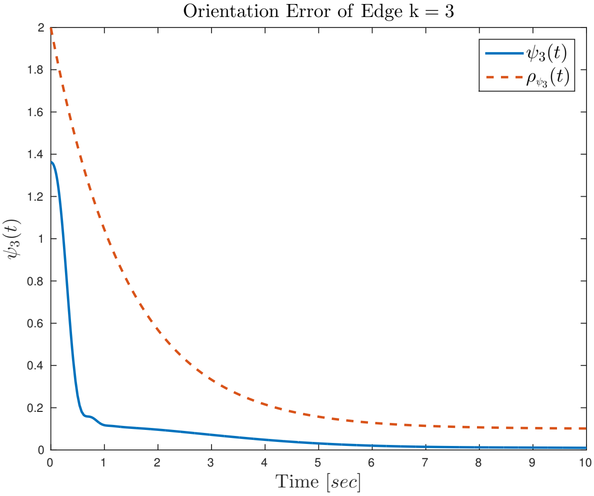


Figure 3.7: The orientation error signal of the edge $(2, 4)$.

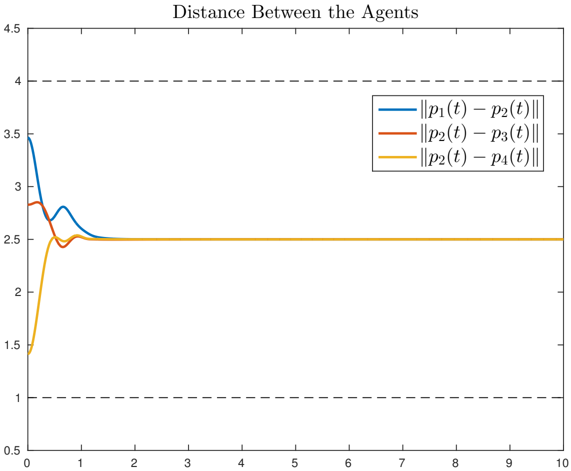


Figure 3.8: The distance between the agents.

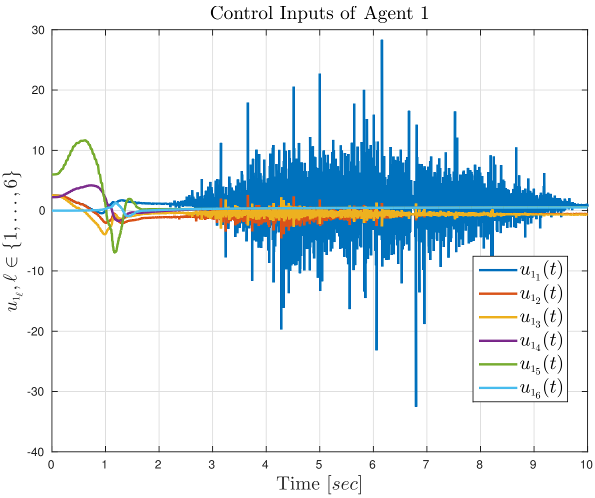


Figure 3.9: The control input signals of agent 1.

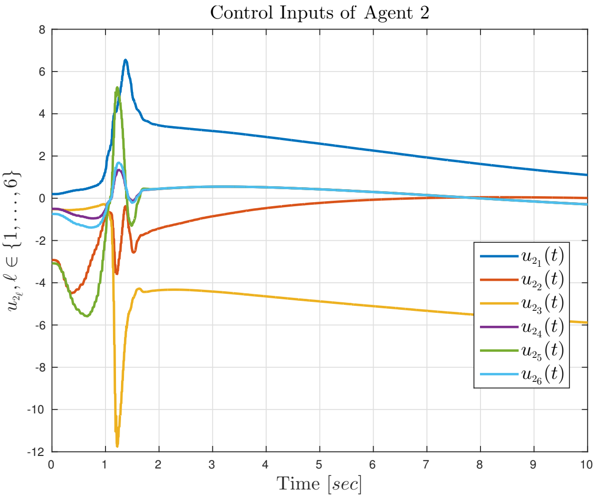


Figure 3.10: The control input signals of agent 2.

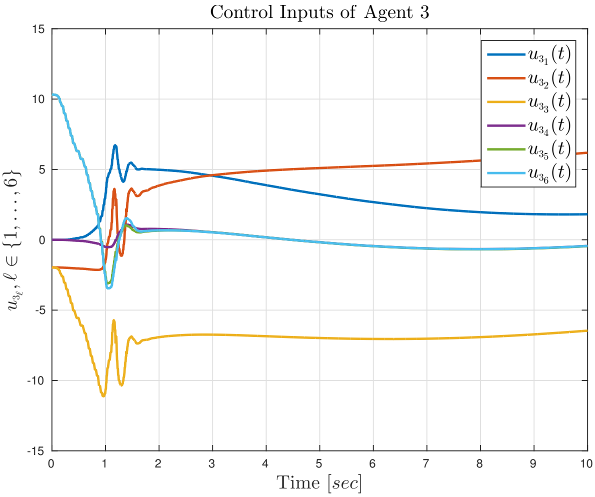


Figure 3.11: The control input signals of agent 3.

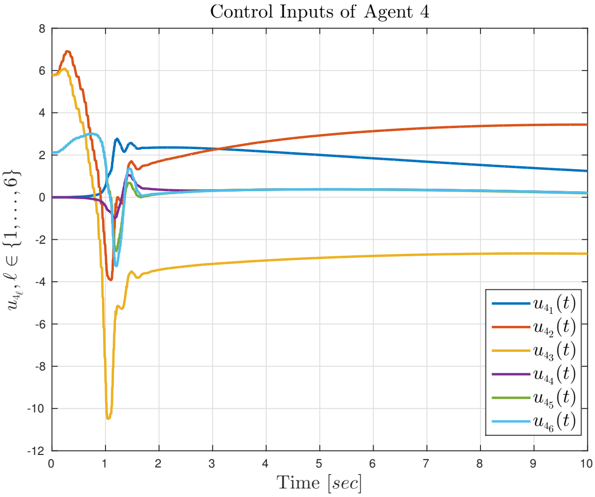


Figure 3.12: The control input signals of agent 4.

Chapter 4

Cooperative Manipulation

This chapter addresses the problem of cooperative manipulation of a single object by multiple robotic agents. More specifically, we present two novel control methodologies for the trajectory tracking of the object's center of mass. Firstly, we design an adaptive control protocol which employs quaternion-based feedback for the object orientation to avoid potential representation singularities. Secondly, we propose a control protocol that guarantees predefined transient and steady-state performance for the object trajectory. Both methodologies are decentralized, since the agents calculate their own signals without communicating with each other, as well as robust to external disturbances and model uncertainties. Moreover, we consider that the grasping points are rigid, and avoid the need for force/torque measurements. Load sharing coefficients are also introduced to account for potential differences in the agents' power capabilities. Finally, simulation and experimental results with two robotic arms support the theoretical findings.

4.1 Introduction

As highlighted in the previous chapter, multi-agent systems have gained significant attention the last years due to the numerous advantages they yield with respect to single-agent setups. In the case of robotic manipulation, heavy payloads and challenging maneuvers necessitate the employment of multiple robotic agents. Although collaborative manipulation of a single object, both in terms of transportation (regulation) and trajectory tracking, has been considered in the research community in the last decades, there still exist several challenges that need to be taken account by on-going research, both in control design as well as experimental evaluation. Moreover, along the lines of designing well-defined discretized abstractions for cooperative manipulation tasks, successful manipulation/transportation of objects plays a crucial role for the potential transitions between the states of the derived discrete system representation. In the previous chapter we addressed the problem of formation control, motivated by its potential application in cooperative manipulation schemes. In this chapter we model explicitly a system of multiple robotic agents grasping an object and develop control protocols for the pose and time trajectory tracking of the center of mass of the object.

Early works develop control architectures where the robotic agents communicate and share information with each other, and completely decentralized schemes, where each agent uses only local information or observers, avoiding potential communication delays (see, indicatively, [72–81]). Impedance and hybrid force/position control is the most common methodology used in the related literature [79–95], where a desired impedance behavior is imposed potentially with force regulation. Most of the aforementioned works employ force/torque sensors to acquire feedback of the object-robots contact forces/torques, which however may result in a performance decline due to sensor noise or mounting difficulties. Recent technological advances allow manipulator grippers to grasp rigidly certain objects (see e.g., [96]), which can render the use of force/torque sensors unnecessary. Force/Torque sensor-free methodologies can be found in [77, 79, 87], which have inspired the dynamic modeling in this work. Moreover, [90] uses an external force estimator, without employing force sensors, [75] presents a force sensor-free control protocol with gain tuning, and [82] considers the object regulation problem without force/torque feedback. Finally, force/torque sensor-free methodologies are developed in [97], where the robot dynamics are not taken into account, and in [94], where a linearization technique is employed.

Another important characteristic is the representation of the agent and object orientation. The most commonly used tools for orientation representation consist of rotation matrices, Euler angles, and the pair angle-axis convention. Rotation matrices, however, are not commonly used in robotic manipulation tasks due to the difficulty of extracting an error vector from them. Moreover, the mapping from Euler angle/axis values to angular velocities exhibits singularities at certain points, rendering thus these representations incompetent. On the other hand, the representation using unit quaternions, which is employed in this work, constitutes a singularity-free orientation representation, without complicating the control design. In cooperative manipulation tasks, unit quaternions are employed in [82, 83, 98] as well as in [99], where the interaction dynamics of cooperative manipulation are analyzed.

In addition, most works in the related literature consider known dynamic parameters regarding the object and the robotic agents. However, the accurate knowledge of such parameters, such as masses or moments of inertia, can be a challenging issue, especially for complex robotic manipulators; adaptive control protocols are proposed in [76] with a gain tuning scheme, in [82], where the object regulation problem is considered, and in [77], [91]. An estimation of parameters is included in [97], whereas [92] and [93] employ fuzzy mechanisms to compensate for model uncertainties. In [95] the authors develop a task-oriented adaptive control protocol using observers. Kinematic uncertainties are handled in [98] and [85].

An internal force and load distribution analysis is performed in [100]; [89] employs a leader-follower scheme, and [101] develops a decentralized force consensus algorithm. Furthermore, [102] introduces hybrid modeling of cooperative manipulation schemes and [103] includes intermittent contact; [104] proposes a kinematic-based multi-robot manipulation scheme, and [12] addresses the problem from a formation-control point of view. Finally, in [105] a navigation function-based approach is used.

The main contribution of this chapter is the introduction of two novel nonlinear control protocols for the trajectory tracking by the center of mass of an object that is rigidly grasped by N robotic agents, without using force/torque measurements at the grasping points.

Firstly, we develop a decentralized control scheme that combines (i) adaptation laws to compensate for external disturbances and uncertainties of the agents' and the object's dynamic parameters, with (ii) quaternion modeling of the object's orientation which avoids undesired representation singularities. A preliminary result on the specific adaptive scheme was developed in [14], whose control law, however, was slightly different and was not tested experimentally or in simulation. Secondly, we propose a decentralized model-free control scheme that guarantees *predefined* transient and steady-state performance for the object's center of mass. We provide detailed stability analyses for both control schemes. Finally, simulation and experimental results with two agents verify the validity of the proposed schemes.

The rest of the chapter is organized as follows. Section 4.2 provides the notation used throughout the paper and necessary background. The modeling of the system as well as the problem formulation are given in Section 4.3. Section 4.4 presents the details of the two proposed control schemes with the corresponding stability analyses, and Section 4.5 illustrates the simulation and experimental results. Finally, Section 4.6 concludes the paper.

4.2 Preliminaries

4.2.1 Unit Quaternions

Given two frames $\{A\}$ and $\{B\}$, we define a unit quaternion $\zeta_{B/A} := [\varphi_{B/A}, \epsilon_{B/A}^\top]^\top \in S^3$ describing the orientation of $\{B\}$ with respect to $\{A\}$, with $\varphi_{B/A} \in \mathbb{R}, \epsilon_{B/A} \in \mathbb{R}^3$, subject to the constraint $\varphi_{B/A}^2 + \epsilon_{B/A}^\top \epsilon_{B/A} = 1$, where S^n denotes the $(n+1)$-dimensional sphere. The relation between $\zeta_{B/A}$ and the corresponding rotation matrix $R_{B/A}$ as well as the axis/angle representation can be found in [26]. For a given quaternion $\zeta_{B/A} = [\varphi_{B/A}, \epsilon_{B/A}^\top]^\top \in S^3$, its conjugate, that corresponds to the orientation of $\{A\}$ with respect to $\{B\}$, is [26] $\zeta_{B/A}^+ = [\varphi_{B/A}, -\epsilon_{B/A}^\top]^\top \in S^3$. Moreover, given two quaternions $\zeta_i := \zeta_{B_i/A_i} = [\varphi_{B_i/A_i}, \epsilon_{B_i/A_i}^\top]^\top, \forall i \in \{1, 2\}$, the quaternion product is defined as [26]

$$\zeta_1 \otimes \zeta_2 = \begin{bmatrix} \varphi_1\varphi_2 - \epsilon_1^\top \epsilon_2 \\ \varphi_1\epsilon_2 + \varphi_2\epsilon_1 + S(\epsilon_1)\epsilon_2 \end{bmatrix} \in S^3, \tag{4.1}$$

where $\varphi_i := \varphi_{B_i/A_i}, \epsilon_i := \epsilon_{B_i/A_i}, \forall i \in \{1, 2\}$.

For a moving frame $\{B\}$ (with respect to $\{A\}$), the time derivative of the

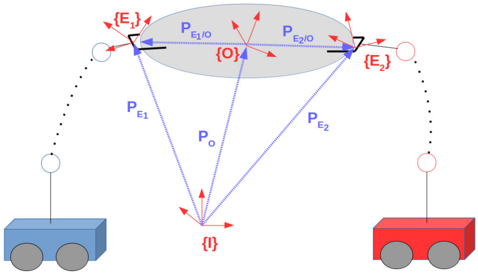


Figure 4.1: Two robotic agents rigidly grasping an object.

quaternion $\zeta_{B/A} = [\varphi_{B/A}, \epsilon_{B/A}^\top]^\top \in S^3$ is given by [26]:

$$\dot{\zeta}_{B/A} = \frac{1}{2} E(\zeta_{B/A}) \omega_{B/A}^A, \tag{4.2a}$$

where $E : S^3 \to \mathbb{R}^{4\times 3}$ is defined as:

$$E(\zeta) = \begin{bmatrix} -\epsilon^\top \\ \varphi I_3 - S(\epsilon) \end{bmatrix}, \forall \zeta = [\varphi, \epsilon^\top]^\top \in S^3.$$

Finally, it can be shown that $[E(\zeta)]^\top E(\zeta) = I_3, \forall \zeta \in S^3$ and hence (4.2a) implies

$$\omega_{B/A}^A = 2[E(\zeta_{B/A})]^\top \dot{\zeta}_{B/A}. \tag{4.2b}$$

It can be also shown that

$$\dot{\omega}_{B/A}^A = 2[E(\zeta_{B/A})]^\top \ddot{\zeta}_{B/A}. \tag{4.2c}$$

4.3 Problem Formulation

Consider N fully actuated robotic agents rigidly grasping an object (see Fig. 4.1). We denote by $\{E_i\}$, $\{O\}$ the end-effector and object's center of mass frames, respectively; $\{I\}$ corresponds to an inertial frame of reference. The rigidity assumption implies that the agents can exert both forces and torques along all directions to the object. In the following, we present the modeling of the coupled kinematics and dynamics of the object and the agents.

4.3.1 Robotic Agents

We denote by $q_i, \dot{q}_i \in \mathbb{R}^{n_i}$, with $n_i \in \mathbb{N}, \forall i \in \mathcal{N}$, the generalized joint-space variables and their time derivatives of agent i, with $q_i := [q_{i_1}, \dots, q_{i_{n_i}}]$. The overall joint configuration is then $q := [q_1^\top, \dots, q_N^\top]^\top, \dot{q} := [\dot{q}_1^\top, \dots, \dot{q}_N^\top]^\top \in \mathbb{R}^n$, with $n := \sum_{i \in \mathcal{N}} n_i$. In addition, the inertial position and Euler-angle orientation of the ith end-effector, denoted by p_{E_i} and η_{E_i}, respectively, can be derived by the forward kinematics and are smooth functions of q_i, i.e. $p_{E_i} : \mathbb{R}^{n_i} \to \mathbb{R}^3$, $\eta_{E_i} : \mathbb{R}^{n_i} \to \mathbb{T}$. The differential equation describing the dynamics of each agent is [26]:

$$B_i(q_i)\ddot{q}_i + C_{q_i}(q_i, \dot{q}_i)\dot{q}_i + g_{q_i}(q_i) + d_{q_i}(q_i, \dot{q}_i, t) = \tau_i - [J_i(q_i)]^\top f_i, \tag{4.3}$$

where $B_i : \mathbb{R}^{n_i} \to \mathbb{R}^{n_i \times n_i}$ is the positive definite inertia matrix, $C_{q_i} : \mathbb{R}^{2n_i} \to \mathbb{R}^{n_i \times n_i}$ is the Coriolis matrix, $g_{q_i} : \mathbb{R}^{n_i} \to \mathbb{R}^{n_i}$ is the joint-space gravity term, $d_{q_i} : \mathbb{R}^{2n_i} \times \mathbb{R}_{\geq 0} \to \mathbb{R}^{n_i}$ is a bounded vector representing unmodeled friction, uncertainties and external disturbances, $f_i \in \mathbb{R}^6$ is the vector of generalized forces that agent i exerts on the grasping point with the object and $\tau_i \in \mathbb{R}^{n_i}$ is the vector of joint torques, acting as control inputs $\forall i \in \mathcal{N}$.

The generalized velocity of each agent's end-effector $v_i := [\dot{p}_{E_i}^\top, \omega_{E_i}^\top]^\top \in \mathbb{R}^6$, can be considered as a transformed state through the differential kinematics $v_i = J_i(q_i)\dot{q}_i$ [26], where $J_i : \mathbb{R}^{n_i} \to \mathbb{R}^{6 \times n_i}$ is a smooth function representing the geometric Jacobian matrix, $\forall i \in \mathcal{N}$ [26]. The latter leads also to

$$\dot{v}_i = J_i(q_i)\ddot{q}_i + J_i^d(q_i, \dot{q}_i)\dot{q}_i, \tag{4.4}$$

where $J_i^d : \mathbb{R}^{2n_i} \to \mathbb{R}^{6 \times n_i}$ represents the Jacobian derivative function, with $J_i^d(q_i, \dot{q}_i) := \dot{J}_i(q_i)$. Then, by employing the differential kinematics as well as (4.4), we obtain from (4.3) the transformed task space dynamics [26]:

$$M_i(q_i)\dot{v}_i + C_i(q_i, \dot{q}_i)v_i + g_i(q_i) + d_i(q_i, \dot{q}_i, t) = u_i - f_i, \tag{4.5}$$

with the corresponding task space terms $M_i : \mathbb{R}^{n_i} \to \mathbb{R}^{6 \times 6}$, $C_i : \mathbb{R}^{2n_i} \to \mathbb{R}^{6 \times 6}$, $g_i : \mathbb{R}^{n_i} \to \mathbb{R}^6$, $d_i : \mathbb{R}^{2n_i} \times \mathbb{R}_{\geq 0} \to \mathbb{R}^6$ and $u_i \in \mathbb{R}^6$ being the task space wrench, related to τ_i via $\tau_i = J_i^\top(q_i)u_i + (I_{n_i} - J_i^\top(q_i)\widetilde{J}_i^\top(q_i))\tau_{i0}$, where $\widetilde{J}_i$ is a generalized inverse of J_i [26]; τ_{i0} concerns redundant agents ($n_i > 6$) and does not contribute to end-effector forces.

The agent task-space dynamics (4.5) can be written in vector form as:

$$M(q)\dot{v} + C(q, \dot{q})v + g(q) + d(q, \dot{q}, t) = u - f, \tag{4.6}$$

where $v := [v_1^\top, \dots, v_N^\top] \in \mathbb{R}^{6N}$, $M := \text{diag}\{[M_i]_{i \in \mathcal{N}}\} \in \mathbb{R}^{6N \times 6N}$, $C := \text{diag}\{[C_i]_{i \in \mathcal{N}}\} \in \mathbb{R}^{6N \times 6N}$, $f := [f_1^\top, \dots, f_N^\top]^\top$, $u := [u_1^\top, \dots, u_N^\top]^\top$, $g := [g_1^\top, \dots, g_N^\top]^\top$, $d := [d_1^\top, \dots, d_N^\top]^\top \in \mathbb{R}^{6N}$.

4.3.2 Object

Regarding the object, we denote by $x_O := [p_O^\top, \eta_O^\top]^\top \in \mathbb{M}$, $v_O := [\dot{p}_O^\top, \omega_O^\top]^\top \in \mathbb{R}^6$ the pose and generalized velocity of the object's center of mass, which is considered as the object's state. We consider the following second order dynamics, which can be derived based on the Newton-Euler formulation:

$$\dot{x}_O = J_O(\eta_O)v_O, \tag{4.7a}$$

$$M_O(x_O)\dot{v}_O + C_O(x_O, \dot{x}_O)v_O + g_O(x_O) + d_O(x_O, \dot{x}_O, t) = f_O, \tag{4.7b}$$

where $M_O : \mathbb{M} \to \mathbb{R}^{6\times 6}$ is the positive definite inertia matrix, $C_O : \mathbb{M} \times \mathbb{R}^6 \to \mathbb{R}^{6\times 6}$ is the Coriolis matrix, $g_O : \mathbb{M} \to \mathbb{R}^6$ is the gravity vector, $d_O : \mathbb{M} \times \mathbb{R}^6 \times \mathbb{R}_{\geq 0} \to \mathbb{R}^6$ a bounded vector representing modeling uncertainties and external disturbances, and $f_O \in \mathbb{R}^6$ is the vector of generalized forces acting on the object's center of mass. Moreover, $J_O : \mathbb{T} \to \mathbb{R}^{6\times 6}$ is the object representation Jacobian $J_O(\eta_O) := \text{diag}\{I_3, J_{O_\eta}(\eta_O)\}$, where $J_{O_\eta} : \mathbb{T} \to \mathbb{R}^{3\times 3}$:

$$J_{O_\eta}(\eta_O) := \begin{bmatrix} 1 & \sin(\phi_O)\tan(\theta_O) & \cos(\phi_O)\tan(\theta_O) \\ 0 & \cos(\phi_O) & -\sin(\theta_O) \\ 0 & \frac{\sin(\phi_O)}{\cos(\theta_O)} & \frac{\cos(\phi_O)}{\cos(\theta_O)} \end{bmatrix},$$

and is not well-defined when $\theta_O = \pm\frac{\pi}{2}$, which is referred to as *representation singularity*. Moreover, it can be proved that

$$\|J_O(\eta_O)\| = \sqrt{\tfrac{|\sin(\theta_O)|+1}{1-\sin^2(\theta_O)}}, \tag{4.8a}$$

$$\|J_O(\eta_O)^{-1}\| = \sqrt{1+\sin(\theta_O)} \leq \sqrt{2}, \tag{4.8b}$$

where $J_O(\cdot)^{-1}$ denotes the matrix inverse.

A possible way to avoid the aforementioned singularity is to transform the Euler angles to a unit quaternion representation for the orientation. Hence, the term η_O can be transformed to the unit quaternion $\zeta_O = [\varphi_O, \epsilon_O^\top]^\top \in S^3$ [26], for which, following Section 4.2.1 and (4.2), one obtains:

$$\dot{\zeta}_O = \frac{1}{2}E(\zeta_O)\omega_O$$
$$\omega_O = 2[E(\zeta_O)]^\top \dot{\zeta}_O,$$

which is a singularity-free representation.

4.3.3 Coupled Dynamics

In view of Fig. 4.1, one concludes that the pose of the agents and the object's center of mass are related as

$$p_{E_i}(q_i) = p_O + p_{E_i/O}(q_i) = p_O + R_{E_i}(q_i)p^{E_i}_{E_i/O}, \tag{4.9a}$$

$$\eta_{E_i}(q_i) = \eta_O + \eta_{E_i/O}, \tag{4.9b}$$

$\forall i \in \mathcal{N}$, where $p^{E_i}_{E_i/O}$ and $\eta_{E_i/O}$ are the *constant* distance and orientation offset vectors between $\{O\}$ and $\{E_i\}$. Following (4.9), along with the fact that, due to the grasping rigidity, it holds that $\omega_{E_i} = \omega_O, \forall i \in \mathcal{N}$, one obtains

$$v_i = J_{O_i}(q_i)v_O, \tag{4.10}$$

where $J_{O_i} : \mathbb{R}^{n_i} \to \mathbb{R}^{6\times 6}$ is the object-to-agent Jacobian matrix, with

$$J_{O_i}(x) = \begin{bmatrix} I_3 & S(p_{O/E_i}(x)) \\ 0_{3\times 3} & I_3 \end{bmatrix}, \forall x \in \mathbb{R}^{n_i}, \tag{4.11}$$

which is always full-rank. Moreover, from (4.10), one obtains

$$\dot{v}_i = J_{O_i}(q_i)\dot{v}_O + J^d_{O_i}(q_i, \dot{q}_i)v_O \tag{4.12}$$

and $J^d_{O_i} : \mathbb{R}^{2n_i} \to \mathbb{R}^{6\times 6}$, with $J^d_{O_i}(q_i, \dot{q}_i) \coloneqq \dot{J}_{O_i}(q_i)$. In addition, it can be proved for J_{O_i} that

$$\|J_{O_i}(x)\| \leq \left\|p^{E_i}_{O/E_i}\right\| + 1, \forall x \in \mathbb{R}^{n_i}, i \in \mathcal{N}, \tag{4.13}$$

which will be used in the subsequent analysis.

The kineto-statics duality along with the grasp rigidity suggest that the force f_O acting on the object's center of mass and the generalized forces $f_i, i \in \mathcal{N}$, exerted by the agents at the grasping points, are related through:

$$f_O = [G(q)]^\top f, \tag{4.14}$$

where $G : \mathbb{R}^n \to \mathbb{R}^{6N\times 6}$, with $G(q) \coloneqq [[J_{O_1}(q_1)]^\top, \dots, [J_{O_N}(q_N)]^\top]^\top$, is the full column-rank grasp matrix. By substituting (4.6) into (4.14), we obtain:

$$f_O = [G(q)]^\top \left(u - M(q)\dot{v} - C(q, \dot{q})v - g(q) - d(q, \dot{q}, t)\right),$$

which, after substituting (4.10), (4.12) , (4.7), and rearranging terms, yields the overall system coupled dynamics:

$$\widetilde{M}(x)\dot{v}_O + \widetilde{C}(x)v_O + \widetilde{g}(x) + \widetilde{d}(x,t) = [G(q)]^\top u, \tag{4.15}$$

where

$$\begin{align}
\widetilde{M}(x) \coloneqq& M_O(x_O) + [G(q)]^\top M(q)G(q) \tag{4.16a}\\
\widetilde{C}(x) \coloneqq& C_O(x_O, \dot{x}_O) + [G(q)]^\top C(q, \dot{q})G(q) + [G(q)]^\top M(q)G^d(q, \dot{q}) \tag{4.16b}\\
\widetilde{g}(x) \coloneqq& g_O(x_O) + [G(q)]^\top g(q). \tag{4.16c}\\
\widetilde{d}(x,t) \coloneqq& d_O(x_O, \dot{x}_O, t) + [G(q)]^\top d(q, \dot{q}, t) \tag{4.16d}\\
G^d(q, \dot{q}) \coloneqq& \Big[[J^d_{O_1}(q_1, \dot{q}_1)]^\top, \dots, [J^d_{O_N}(q_N, \dot{q}_N)]^\top\Big]^\top, \tag{4.16e}
\end{align}$$

and x is the overall state $x \coloneqq [q^\top, \dot{q}^\top, x_O^\top, \dot{x}_O^\top]^\top \in \mathbb{R}^{2n+6} \times \mathbb{M}$. Moreover, the following Lemma is necessary for the following analysis.

Lemma 4.1. *The matrix $\widetilde{M}(x)$ is symmetric and positive definite and the matrix $\dot{\widetilde{M}}(x) - 2\widetilde{C}(x)$ is skew symmetric, i.e.,*

$$\begin{align*}
\Big[\dot{\widetilde{M}}(x) - 2\widetilde{C}(x)\Big]^\top &= -\Big[\dot{\widetilde{M}}(x) - 2\widetilde{C}(x)\Big], \forall x \in \mathbb{R}^6\\
y^\top \Big[\dot{\widetilde{M}}(x) - 2\widetilde{C}(x)\Big]y &= 0, \quad \forall x, y \in \mathbb{R}^6.
\end{align*}$$

Proof. The matrices M_O and M_i are symmetric and positive definite, $\forall i \in \mathcal{N}$ and the matrices $\dot{M}_i(q_i) - 2C_i(q_i, \dot{q}_i)$, $M_O(x_O) - 2C_O(x_O, \dot{x}_O)$ are skew-symmetric, $\forall i \in \mathcal{N}$ [26], which leads to the skew-symmetry of $\dot{M}(q) - 2C(q, \dot{q})$. Therefore, since $G(q)$ is full column-rank, we can conclude the symmetry and positive definiteness of $\widetilde{M}(x)$. Regarding the skew symmetry of $\dot{\widetilde{M}}(x) - 2\widetilde{C}(x)$, we define first $A(x) := [\dot{G}(q)]^\top M(q) G(q)$, and we have from (4.16b):

$$\begin{aligned}\dot{\widetilde{M}}(x) - 2\widetilde{C}(x) = &\dot{M}_O(x_O) - 2C_O(x_O, \dot{x}_O) + [G(q)]^\top(\dot{M}(q) - 2C(q, \dot{q}))G(q) + \\ &A(x) - [A(x)]^\top,\end{aligned}$$

which, by employing the skew-symmetry of $\dot{M}_O(x_O) - 2C_O(x_O, \dot{x}_O)$ and $\dot{M}(q) - 2C(q, \dot{q})$, leads to $[\dot{\widetilde{M}}(x) - 2\widetilde{C}(x)]^\top = -[\dot{\widetilde{M}}(x) - 2\widetilde{C}(x)]$, which completes the proof. □

The positive definiteness of $\widetilde{M}(x)$ leads to the property

$$\underline{m} I_6 \leq \widetilde{M}(x) \leq \bar{m} I_6, \tag{4.17}$$

$\forall x \in \mathbb{R}^{2n+6} \times \mathbb{R}$, where $\underline{m}$ and $\bar{m}$ are positive unknown constants.

We are now ready to state the problem treated in this paper:

Problem 4.1. Given a desired bounded object smooth pose trajectory specified by $\bar{x}_\mathrm{d}(t) := [(p_\mathrm{d}(t))^\top, (\eta\mathrm{d}(t))^\top]^\top$, $p_\mathrm{d}(t) \in \mathbb{R}^3, \eta_\mathrm{d}(t) := [\varphi_\mathrm{d}(t), \theta_\mathrm{d}(t), \psi_\mathrm{d}(t)] \in \mathbb{T}$, with bounded first and second derivatives, and $\theta_\mathrm{d}(t) \in [-\bar{\theta}, \bar{\theta}] \subset (-\frac{\pi}{2}, \frac{\pi}{2}), \forall t \in \mathbb{R}_{\geq 0}$, as well as $v_O(0) = 0_6$, determine a continuous time-varying control law u in (4.15) such that

$$\lim_{t\to\infty} \begin{bmatrix} p_O(t) \\ \eta_O(t) \end{bmatrix} = \begin{bmatrix} p_\mathrm{d}(t) \\ \eta_\mathrm{d}(t) \end{bmatrix}.$$

The requirement $\theta_\mathrm{d}(t) \in [-\bar{\theta}, \bar{\theta}] \subset (-\frac{\pi}{2}, \frac{\pi}{2}), \forall t \in \mathbb{R}_{\geq 0}$ is a necessary condition needed to ensure that tracking of θ_d will not result in singular configurations of $J_O(\eta_O)$, which is needed for the control protocol of Section 4.4.2. The constant $\bar{\theta} \in [0, \frac{\pi}{2})$ can be taken arbitrarily close to $\frac{\pi}{2}$.

To solve the aforementioned problem, we need the following assumptions regarding the agent feedback, the bounds of the uncertainties/disturbances, and the kinematic singularities.

Assumption 4.1. (Feedback) Each agent $i \in \mathcal{N}$ has continuous feedback of its own state $q_i, \dot{q}_i$.

Assumption 4.2. (Object geometry) Each agent $i \in \mathcal{N}$ knows the constant offsets $p^{E_i}_{E_i/O}$ and $\eta_{E_i/O}, \forall i \in \mathcal{N}$.

Assumption 4.3. (Kinematic singularities) The robotic agents operate away from kinematic singularities, i.e., $q_i \in \{q_i \in \mathbb{R}^{n_i} : |\det(J_i(q_i)[J_i(q_i)]^\top)| \geq \underline{J}_i > 0\}$ for positive constants $\underline{J}_i > 0, \forall i \in \mathcal{N}$.

Assumption 4.1 is realistic for real manipulation systems, since on-board sensor can provide accurately the measurements $q_i, \dot{q}_i$. The object geometrical characteristics in Assumption 4.2 can be obtained by on-board sensors, whose inaccuracies are not modeled here and constitute part of future work. Finally, Assumption 4.3 is used for those q_i that bring x_O close to the desired trajectory $\bar{x}_\mathrm{d}(t)$. Intuitively, this assumption states that the q_i that achieve $x_O(t) = \bar{x}_\mathrm{d}(t), \forall t \in \mathbb{R}_{\geq 0}$ are sufficiently far from kinematic singular configurations.

Since each agent has feedback from its state $q_i, \dot{q}_i$, it can compute through the forward and differential kinematics the end-effector pose $p_{E_i}(q_i), \eta_{E_i}(q_i)$ and the velocity v_i, $\forall i \in \mathcal{N}$. Moreover, since it knows $p^{E_i}_{E_i/O}$ and $\eta_{E_i/O}$, it can compute $J_{O_i}(q_i)$ from (4.11), and x_O, v_O by inverting (4.9) and (4.10), respectively. Consequently, each agent can then compute the object unit quaternion ζ_O as well as $\dot{\zeta}_O$.

4.4 Main Results

In this section we present two control schemes for the solution of Problem 4.1. The proposed controllers are decentralized, in the sense that the agents calculate their control signal on their own, without communicating with each other, as well as robust, since they do not take into account the dynamic properties of the agents or the object (mass/inertia moments) or the uncertainties/external disturbances modeled by the function $\widetilde{d}(x, t)$ in (4.15). The first control scheme is presented in Section 4.4.1, and is based on quaternion feedback and adaptation laws, while the second control scheme is given in Section 4.4.2 and is inspired by the Prescribed Performance Control (PPC) methodology introduced in [29].

4.4.1 Adaptive Control with Quaternion Feedback

Firstly, we need the following assumption regarding the model uncertainties/external disturbances.

Assumption 4.4. (Uncertainties/Disturbance parameterization) There exist positive, finite *unknown* constants $\bar{d}_O, \bar{d}_i$ and known bounded functions $\delta_O : \mathbb{M} \times \mathbb{R}^6 \times \mathbb{R}_{\geq 0} \to \mathbb{R}^6, \delta_i : \mathbb{R}^{2n_i} \times \mathbb{R}_{\geq 0} \to \mathbb{R}^6$, such that

$$d_O(x_O, \dot{x}_O, t) = \bar{d}_O \delta_O(x_O, \dot{x}_O, t),$$
$$d_i(q_i, \dot{q}_i, t) = \bar{d}_i \delta_i(q_i, \dot{q}_i, t),$$

$\forall q_i, \dot{q}_i \in \mathbb{R}^{n_i}, x_O \in \mathbb{M}, \dot{x}_O \in \mathbb{R}^6, t \in \mathbb{R}_{\geq 0}, i \in \mathcal{N}$.

The desired Euler angle orientation vector $\eta_\mathrm{d} : \mathbb{R}_{\geq 0} \to \mathbb{T}$ is transformed first to the unit quaternion $\zeta_\mathrm{d} : \mathbb{R}_{\geq 0} \to S^3$ [26]. Then, we need to define the errors associated with the object pose and the desired pose trajectory. We first define the state that corresponds to the position error:

$$e_p := p_O - p_{\mathrm{d}}(t).$$

Since unit quaternions do not form a vector space, they cannot be subtracted to form an orientation error; instead we should use the properties of the quaternion group algebra. Let $e_\zeta = [e_\varphi, e_\epsilon^\top]^\top \in S^3$ be the unit quaternion describing the orientation error. Then, it holds that [26],

$$e_\zeta = \zeta_{\mathrm{d}}(t) \otimes \zeta_O^+ = \begin{bmatrix} \varphi_{\mathrm{d}}(t) \\ \epsilon_{\mathrm{d}}(t) \end{bmatrix} \otimes \begin{bmatrix} \varphi_O \\ -\epsilon_O \end{bmatrix},$$

which, by using (4.1), becomes:

$$e_\zeta = \begin{bmatrix} e_\varphi \\ e_\epsilon \end{bmatrix} := \begin{bmatrix} \varphi_O \varphi_{\mathrm{d}}(t) + \epsilon_O^\top \epsilon_{\mathrm{d}}(t) \\ \varphi_O \epsilon_{\mathrm{d}}(t) - \varphi_{\mathrm{d}}(t)\epsilon_O + S(\epsilon_O)\epsilon_{\mathrm{d}}(t) \end{bmatrix}. \tag{4.18}$$

By employing (4.2) and certain properties of skew-symmetric matrices [106], it can be shown that the error dynamics of e_p, e_φ are:

$$\dot{e}_p = \dot{p}_O - \dot{p}_{\mathrm{d}}(t) \tag{4.19a}$$

$$\dot{e}_\varphi = \tfrac{1}{2} e_\epsilon^\top e_\omega \tag{4.19b}$$

$$\dot{e}_\epsilon = -\tfrac{1}{2}\left[e_\varphi I_3 + S(e_\epsilon)\right] e_\omega - S(e_\epsilon)\omega_{\mathrm{d}}(t), \tag{4.19c}$$

where $e_\omega := \omega_O - \omega_{\mathrm{d}}(t)$ is a state for the angular velocity error, with $\omega_{\mathrm{d}}(t) = 2[E(\zeta_{\mathrm{d}}(t))]^\top \dot{\zeta}_{\mathrm{d}}(t)$, as indicated by (4.2b).

Due to the ambiguity of unit quaternions, when $\zeta_O = \zeta_{\mathrm{d}}$, then $e_\zeta = [1, 0_3^\top]^\top \in S^3$. If $\zeta_O = -\zeta_{\mathrm{d}}$, then $e_\zeta = [-1, 0_3^\top]^\top \in S^3$, which, however, represents the same orientation. Therefore, the control objective established in Problem 4.1 is equivalent to

$$\lim_{t\to\infty} \begin{bmatrix} e_p(t) \\ |e_\varphi(t)| \\ e_\epsilon(t) \end{bmatrix} = \begin{bmatrix} 0_3 \\ 1 \\ 0_3 \end{bmatrix}.$$

The left hand side of (4.5), after employing (4.10) and (4.12), becomes

$$M_i(q_i)\dot{v}_i + C_i(q_i,\dot{q}_i)v_i + g_i(q_i) + d_i(q_i,\dot{q}_i,t) = M_i(q_i)\Big(J_{O_i}(q_i)\dot{v}_O + J_{O_i}^d(q_i,\dot{q}_i)v_O\Big) + C_i(q_i,\dot{q}_i)J_{O_i}(q_i)v_O + g_i(q_i) + d_i(q_i,\dot{q}_i,t).$$

which, according to Assumption 4.4 and the fact that the manipulator dynamics can be linearly parameterized with respect to dynamic parameters [107], becomes

$$M_i(q_i)J_{O_i}(q_i)\dot{v}_O + \Big(M_i(q_i)J_{O_i}^d(q_i,\dot{q}_i) + C_i(q_i,\dot{q}_i)J_{O_i}(q_i)\Big)v_O + g_i(q_i) + d_i(q_i,\dot{q}_i,t) = H_i(q_i,\dot{q}_i,v_O,\dot{v}_O)\vartheta_i + \bar{d}_i\delta_i(q_i,\dot{q}_i,t), \tag{4.20a}$$

$\forall i \in \mathcal{N}$, where $\vartheta_i \in \mathbb{R}^{\ell}, \ell \in \mathbb{N}$, are vectors of unknown but constant dynamic parameters of the agents, appearing in the terms M_i, C_i, g_i, and $H_i : \mathbb{R}^{2n_i+12} \to \mathbb{R}^{6\times\ell}$ are known regressor matrices, independent of $\vartheta_i, i \in \mathcal{N}$. Without loss of generality, we assume here that the dimension of ϑ_i is the same, ℓ for all the agents. Similarly, the dynamical terms of the left hand side of (4.7b) can be written as

$$\begin{aligned} &M_O(x_O)\dot{v}_O + C_O(x_O,\dot{x}_O)v_O + g_O(x_O) + d_O(x_O,\dot{x}_O,t) = Y_O(x_O,\dot{x}_O,v_O,\dot{v}_O)\vartheta_O + \\ &\bar{d}_O\delta_O(x_O,\dot{x}_O,t), \end{aligned} \tag{4.20b}$$

where $\vartheta_O \in \mathbb{R}^{\ell_O}, \ell_O \in \mathbb{N}$ is a vector of unknown but constant dynamic parameters of the object, appearing in the terms M_O, C_O, g_O, and $Y_O : \mathbb{M} \times \mathbb{R}^{18} \to \mathbb{R}^{6\times\ell_O}$ is a known regressor matrix, independent of ϑ_O. It is worth noting that the choice for ℓ and ℓ_O is not unique. In the same vein, since $J_{O_i}(q_i)$, as given in (4.11), depends only on q_i and not on $\vartheta_i, \vartheta_O, \forall i \in \mathcal{N}$, we can write:

$$\begin{aligned} &[J_{O_i}(q_i)]^\top M_i(q_i) J_{O_i}(q_i)\dot{v}_O + [J_{O_i}(q_i)]^\top g_i(q_i) + \Big([J_{O_i}(q_i)]^\top M_i(q_i) J^d_{O_i}(q_i,\dot{q}_i) \\ &+ [J_{O_i}(q_i)]^\top C_i(q_i,\dot{q}_i) J_{O_i}(q_i)\Big) v_O + [J_{O_i}(q_i)]^\top d_i(q_i,\dot{q}_i,t) = Y_i(q_i,\dot{q}_i,v_O,\dot{v}_O)\vartheta_i \\ &+ [J_{O_i}(q_i)]^\top \bar{d}_i \delta_i(q_i,\dot{q}_i,t), \end{aligned} \tag{4.21}$$

where $Y_i : \mathbb{R}^{2n_i+12} \to \mathbb{R}^{6\times\ell}$ are also regressor matrices independent of ϑ_i, ϑ_O. Hence, in view of (4.16), (4.20) and (4.21), the left-hand side of (4.15) can be written as:

$$\begin{aligned} &\widetilde{M}(x)\dot{v}_O + \widetilde{C}(x)v_O + \widetilde{g}(x) + \widetilde{d}(x,t) = Y_O(x_O,\dot{x}_O,v_O,\dot{v}_O)\vartheta_O + [Y(q,\dot{q},v_O,\dot{v}_O)]^\top\vartheta + \\ &\bar{d}_O\delta_O(x_O,\dot{x}_O) + \sum_{i\in\mathcal{N}} [J_{O_i}(q_i)]^\top \bar{d}_i\delta_i(q_i,\dot{q}_i,t) \end{aligned} \tag{4.22}$$

where $\vartheta := [\vartheta_1^\top, \dots, \vartheta_N^\top]^\top \in \mathbb{R}^{N\ell}$ and $Y(q,\dot{q},v_O,\dot{v}_O) := [[Y_1(q_1,\dot{q}_1,v_O,\dot{v}_O)]^\top, \dots, [Y_N(q_N,\dot{q}_N,v_O,\dot{v}_O)]^\top]^\top \in \mathbb{R}^{N\ell\times 6}$.

Let us now introduce the states $\hat{\vartheta}_{O_i} \in \mathbb{R}^{\ell_O}$ and $\hat{\vartheta}_i \in \mathbb{R}^{\ell}$ which represent the estimates of ϑ_O and ϑ_i, respectively, by agent $i \in \mathcal{N}$, and the corresponding stack vectors $\hat{\vartheta}_O := [(\hat{\vartheta}_{O_1})^\top, \dots, (\hat{\vartheta}_{O_N})^\top]^\top \in \mathbb{R}^{N\ell_O}, \hat{\vartheta} := [\hat{\vartheta}_1^\top, \dots, \hat{\vartheta}_N^\top]^\top \in \mathbb{R}^{N\ell}$, for which we formulate the associated errors $e_{\vartheta_O} \in \mathbb{R}^{N\ell_O}, e_\vartheta \in \mathbb{R}^{N\ell}$ as

$$e_{\vartheta_O} := \begin{bmatrix} e_{\vartheta_{O,1}} \\ \vdots \\ e_{\vartheta_{O,N}} \end{bmatrix} := \begin{bmatrix} \vartheta_O - \hat{\vartheta}_{O_1} \\ \vdots \\ \vartheta_O - \hat{\vartheta}_{O_N} \end{bmatrix} = \vartheta_O \cdot \mathbb{1}_{N\ell_O} - \hat{\vartheta}_O \tag{4.23a}$$

$$e_\vartheta := \begin{bmatrix} e_{\vartheta_1} \\ \vdots \\ e_{\vartheta_N} \end{bmatrix} := \begin{bmatrix} \vartheta_1 - \hat{\vartheta}_1 \\ \vdots \\ \vartheta_N - \hat{\vartheta}_N \end{bmatrix} = \vartheta - \hat{\vartheta}. \tag{4.23b}$$

In the same vein, we introduce the states $\hat{d}_{O_i} \in \mathbb{R}$ and $\hat{d}_i \in \mathbb{R}$ that correspond to the estimates of the constants $\bar{d}_O$ and $\bar{d}_i$, respectively, by agent $i \in \mathcal{N}$, and the corresponding stack vectors $\hat{d}_O \coloneqq [\hat{d}_{O_1}, \dots, \hat{d}_{O_N}]^\top \in \mathbb{R}^N$, $\hat{d} \coloneqq [\hat{d}_1, \dots, \hat{d}_N]^\top \in \mathbb{R}^N$, for which we also formulate the associated errors $e_{d_O}, e_d \in \mathbb{R}^N$ as

$$e_{d_O} \coloneqq \begin{bmatrix} e_{d_{O,1}} \\ \vdots \\ e_{d_{O,2}} \end{bmatrix} \coloneqq \begin{bmatrix} \bar{d}_O - \hat{d}_{O_1} \\ \vdots \\ \bar{d}_O - \hat{d}_{O_N} \end{bmatrix} = \bar{d}_O \cdot \mathbb{1}_N - \hat{d}_O \tag{4.24a}$$

$$e_d \coloneqq \begin{bmatrix} e_{d_1} \\ \vdots \\ e_{d_N} \end{bmatrix} \coloneqq \begin{bmatrix} \bar{d}_1 - \hat{d}_1 \\ \vdots \\ \bar{d}_N - \hat{d}_N \end{bmatrix} = \bar{d} - \hat{d}, \tag{4.24b}$$

where we have also used the notation $\bar{d} \coloneqq [\bar{d}_1, \dots, \bar{d}_N]^\top$.

Next, we design the reference velocity $v_f : \mathbb{R}^6 \times \mathbb{R}_{\geq 0} \to \mathbb{R}^6$ as

$$v_f(e, t) \coloneqq v_{\mathrm{d}}(t) - K_f e = \begin{bmatrix} \dot{p}_{\mathrm{d}}(t) - k_p e_p \\ \omega_{\mathrm{d}}(t) + k_\zeta e_\epsilon \end{bmatrix} \tag{4.25}$$

with $K_f \coloneqq \mathrm{diag}\{k_p, k_\zeta\}$, $e \coloneqq [e_p^\top, -e_\epsilon^\top]^\top \in \mathbb{R}^6$, k_p, k_ζ positive control gains, and the derivative $v_f^d : \mathbb{R}^6 \times S^3 \times \mathbb{R}_{\geq 0} \to \mathbb{R}^6$, with

$$v_f^d(\dot{p}_O, e_\omega, e_\zeta, t) \coloneqq \dot{v}_f(e, t) = \begin{bmatrix} \ddot{p}_{\mathrm{d}}(t) - k_p \dot{e}_p \\ \dot{\omega}_{\mathrm{d}}(t) + k_\zeta \dot{e}_\epsilon, \end{bmatrix} \tag{4.26}$$

where we have implicitly used (4.19); $\dot{\omega}_{\mathrm{d}}(t)$ can be calculated via (4.2c). We also introduce the respective velocity error e_v as

$$e_{v_f} \coloneqq v_O - v_f(e, t). \tag{4.27}$$

Denote by χ_i the overall state $\chi_i = [q_i^\top, \dot{q}_i^\top, x_O^\top, \dot{x}_O^\top, e_p^\top, e_\zeta^\top, e_{v_f}^\top, e_\omega^\top, e_{\vartheta_i}^\top, e_{\vartheta_{O,i}}^\top, e_{d_i}, e_{d_{O,i}}]^\top$ of appropriate dimension $\mathbb{X}_i$, and design the adaptive control law $u_i : \mathbb{X}_i \times \mathbb{R}_{\geq 0} \to \mathbb{R}^6$ in (4.15), for each agent $i \in \mathcal{N}$, as:

$$\begin{aligned} u_i(\chi_i, t) = [J_{O_i}(q_i)]^{-\top} \Big[& Y_i\Big(q_i, \dot{q}_i, v_f(e,t), v_f^d(\dot{p}_O, e_\omega, e_\zeta, t)\Big)\hat{\vartheta}_i - c_i e - K_{v_i} e_{v_f} + \\ & c_i Y_O\Big(x_O, \dot{x}_O, v_f(e,t), v_f^d(\dot{p}_O, e_\omega, e_\zeta, t)\Big)\hat{\vartheta}_{O_i} + c_i \hat{d}_{O_i} \delta_O(x_O, \dot{x}_O, t)\Big] + \hat{d}_i \delta_i(q_i, \dot{q}_i, t), \end{aligned} \tag{4.28}$$

where K_{v_i} are diagonal positive definite gain matrices, $\forall i \in \mathcal{N}$, and c_i are load sharing coefficients satisfying $c_i \in (0,1), \forall i \in \mathcal{N}$, and $\sum_{i \in \mathcal{N}} c_i = 1$. Note that we have implicitly used the expressions $\hat{\vartheta}_i = e_{\vartheta_i} + \vartheta_i, \hat{\vartheta}_{O_i} = e_{\vartheta_{O,i}} + \vartheta_O, \hat{d}_i =$

$e_{d_i} + \bar{d}_i, \hat{d}_{O_i} = e_{d_{O,i}} + \bar{d}_O$ to express $u_i, i \in \mathcal{N}$, as a function of the errors (4.23) and (4.24).

In addition, we design the following adaptation laws:

$$\dot{\hat{\theta}}_i = -\gamma_i \Big[Y_i\Big(q_i, \dot{q}_i, v_f(e,t), v_f^d(\dot{p}_O, e_\omega, e_\zeta, t)\Big)\Big]^\top e_{v_f}, \tag{4.29a}$$

$$\dot{\hat{\theta}}_{O_i} = -c_i\gamma_{O_i} \Big[Y_O\Big(x_O, \dot{x}_O, v_f(e,t), v_f^d(\dot{p}_O, e_\omega, e_\zeta, t)\Big)\Big]^\top e_{v_f} \tag{4.29b}$$

$$\dot{\hat{d}}_i = -\beta_i(\delta_i(q_i, \dot{q}_i, t))^\top J_{O_i}(q_i) e_{v_f} \tag{4.29c}$$

$$\dot{\hat{d}}_{O_i} = -c_i\beta_{O_i}(\delta_O(x_O, \dot{x}_O, t))^\top e_{v_f}, \tag{4.29d}$$

with arbitrary bounded initial conditions, where $\beta_i, \beta_{O_i}, \gamma_i, \gamma_{O_i} \in \mathbb{R}_{>0}$ are positive gains, $\forall i \in \mathcal{N}$.

The following theorem summarizes the main results of this subsection.

Theorem 4.1. *Consider N robotic agents rigidly grasping an object with coupled dynamics described by (4.15) and unknown dynamic parameters. Then, under Assumptions 4.1-4.4, by applying the control protocol* (4.28) *with the adaptation laws* (4.29)*, the object pose converges asymptotically to the desired pose trajectory. Moreover, all closed loop signals are bounded.*

Proof. Consider the stack state vector $\chi := [\chi_1^\top, \dots, \chi_N^\top]^\top \in \mathbb{X} := \mathbb{X}_1 \times \cdots \times \mathbb{X}_N$, and the nonnegative function $V : \mathbb{X} \to \mathbb{R}_{\geq 0}$, with

$$V(\chi) := \tfrac{1}{2}e_p^\top e_p + 2(1 - e_\varphi) + \tfrac{1}{2}e_{v_f}^\top \widetilde{M}(x) e_{v_f} + \tfrac{1}{2}e_\vartheta^\top \Gamma^{-1} e_\vartheta + \tfrac{1}{2}e_{\vartheta_O}^\top \Gamma_O^{-1} e_{\vartheta_O} + \\ \tfrac{1}{2}e_d^\top B^{-1} e_d + \tfrac{1}{2}e_{d_O}^\top B_O^{-1} e_{d_O}, \tag{4.30}$$

where $B := \text{diag}\{[\beta_i]_{i\in\mathcal{N}}\}, B_O := \text{diag}\{[\beta_{O_i}]_{i\in\mathcal{N}}\}, \Gamma := \text{diag}\{[\gamma_i I_\ell]_{i\in\mathcal{N}}\}, \Gamma_O := \text{diag}\{[\gamma_{O_i} I_{\ell_O}]_{i\in\mathcal{N}}\}$. Recall that the error quaternion e_ζ as defined (4.18) is a unit quaternion and hence $e_\varphi \in [-1, 1]$.

By considering the derivatives of the elements in χ, it can be concluded from the aforementioned dynamics that the closed loop dynamics can be written in the form $\dot{\chi} = f_{\text{cl}}(\chi, t)$, for a locally Lipschitz function $f_{\text{cl}} : \mathbb{X} \times \mathbb{R}_{\geq 0} \to \mathbb{X}$. By taking the derivative of V along the solutions of the closed loop system, we obtain

$$\dot{V}(\chi) = e^\top(v_O - v_{\text{d}}(t)) + \tfrac{1}{2}e_{v_f}^\top \dot{\widetilde{M}}(x) e_{v_f} - e_{v_f}^\top \widetilde{M}(x)\dot{v}_f + e_{v_f}^\top \Big([G(q)]^\top u - \widetilde{C}(x) v_O - \\ \widetilde{g}(x) - \widetilde{d}(x)\Big) - e_\vartheta^\top \Gamma^{-1}\dot{\hat{\vartheta}} - e_{\vartheta_O}^\top \Gamma_O^{-1}\dot{\hat{\vartheta}}_O - e_d^\top B^{-1}\dot{\hat{d}} - e_{d_O}^\top B_O^{-1}\dot{\hat{d}}_O,$$

which, by using (4.27) and (4.25), becomes

$$\dot{V}(\chi) = -e^\top K_f e + \frac{1}{2}e_{v_f}^\top \dot{\widetilde{M}}(x) e_{v_f} - e_{v_f}^\top \widetilde{C}(x) e_{v_f} + e_{v_f}^\top \Big([G(q)]^\top u + e - \\ \widetilde{M}(x) v_f^d(\dot{p}_O, e_\omega, e_\zeta, t) - \widetilde{g}(x) - \widetilde{C}(x) v_f(e,t) - \widetilde{d}(x,t)\Big) - e_\vartheta^\top \Gamma^{-1}\dot{\hat{\vartheta}} - e_{\vartheta_O}^\top \Gamma_O^{-1}\dot{\hat{\vartheta}}_O \\ - e_d^\top B^{-1}\dot{\hat{d}} - e_{d_O}^\top B_O^{-1}\dot{\hat{d}}_O.$$

By employing Lemma 4.1 as well as (4.22), $\dot{V}(\chi)$ can be written as

$$\dot{V}(\chi) = -e^\top K_f e + e_{v_f}^\top \sum_{i\in\mathcal{N}} \Big[[J_{O_i}(q_i)]^\top u_i + c_i e - \delta_O(x_O, \dot{x}_O, t) c_i \bar{d}_O -$$

$$Y_O\Big(x_O, \dot{x}_O, v_f(e,t), v_f^d(\dot{p}_O, e_\omega, e_\zeta, t)\Big) c_i \vartheta_O - Y_i\Big(q_i, \dot{q}_i, v_f(e,t), v_f^d(\dot{p}_O, e_\omega, e_\zeta, t)\Big)\theta_i$$

$$- [J_{O_i}(q_i)]^\top \delta_i(q_i, \dot{q}_i, t)\bar{d}_i \Big] - \sum_{i\in\mathcal{N}} \Big(\frac{e_{\vartheta_i}^\top}{\gamma_i}\dot{\hat{\vartheta}}_i + \frac{e_{\vartheta_{O,i}}^\top}{\gamma_{O_i}}\dot{\hat{\vartheta}}_{O_i} + \frac{e_{d_i}}{\beta_i}\dot{\hat{d}}_i + \frac{e_{d_{O,i}}}{\beta_{O_i}}\dot{\hat{d}}_{O_i} \Big),$$

and after substituting the adaptive control laws (4.28),

$$\dot{V}(\chi) = -e^\top K_f e - \sum_{i\in\mathcal{N}} e_{v_f}^\top K_{v_i} e_{v_f} - \sum_{i\in\mathcal{N}} \Big(\frac{e_{\vartheta_i}^\top}{\gamma_i}\dot{\hat{\vartheta}}_i + \frac{e_{\vartheta_{O,i}}^\top}{\gamma_{O_i}}\dot{\hat{\vartheta}}_{O_i} + \frac{e_{d_i}}{\beta_i}\dot{\hat{d}}_i - \frac{e_{d_{O,i}}}{\beta_{O_i}}\dot{\hat{d}}_{O_i} \Big)$$

$$- e_{v_f}^\top \sum_{i\in\mathcal{N}} \Big[Y_O\Big(x_O, \dot{x}_O, v_f(e,t), v_f^d(\dot{p}_O, e_\omega, e_\zeta, t)\Big) c_i e_{\vartheta_{O,i}} + \delta_O(x_O, \dot{x}_O, t) c_i e_{d_{O,i}} +$$

$$Y_i\Big(q_i, \dot{q}_i, v_f(e,t), v_f^d(\dot{p}_O, e_\omega, e_\zeta, t)\Big) e_{\theta_i} + [J_{O_i}(q_i)]^\top \delta_i(q_i, \dot{q}_i, t) e_{d_i} \Big],$$

which, after substituting the adaptation laws (4.29), finally becomes

$$\dot{V}(\chi) = -e^\top K_f e - \sum_{i\in\mathcal{N}} e_{v_f}^\top K_{v_i} e_{v_f} = -k_p \|e_p\|^2 - k_\zeta \|e_\epsilon\|^2 - \sum_{i\in\mathcal{N}} e_{v_f}^\top K_{v_i} e_{v_f},$$

which is non-positive. We conclude therefore the boundedness of V and of χ, which implies the boundedness of the dynamic terms $\widetilde{M}(x), \widetilde{C}(x), \widetilde{g}(x)$. Moreover, by invoking the boundedness of $p_\mathrm{d}(t), v_\mathrm{d}(t), \omega_\mathrm{d}(t), \dot{v}_\mathrm{d}(t), \dot{\omega}_\mathrm{d}(t)$, we conclude the boundedness of $v_f(e,t), v_O, v_i, \hat{\vartheta}_O$, $\hat{\vartheta}$, $\hat{d}$, $\hat{d}_O$. From (4.19) and (4.26) we also conclude the boundedness of $\dot{v}_f(e,t)$ and therefore, the boundedness of the control and adaptation laws (4.28) and (4.29). Thus, we can conclude the boundedness of the second derivative $\ddot{V}(\chi)$ and hence the uniform continuity of $\dot{V}(\chi)$. By invoking Barbalat's lemma [108], we deduce therefore that $\lim_{t\to\infty} \dot{V}(\chi(t)) = 0$ and, consequently, that $\lim_{t\to\infty} e_p(t) = 0_3$, $\lim_{t\to\infty} e_{v_f}(t) = 0_6$, and $\lim_{t\to\infty} \|e_\epsilon(t)\|^2 = 0$, which, given that e_ζ is a unit quaternion, leads to the configuration $(e_p, e_{v_f}, e_\varphi, e_\epsilon) = (0_3, 0_6, \pm1, 0_3)$. The closed loop dynamics of e_φ, as given in (4.19b), can be written, in view of (4.25), as $\dot{e}_\varphi = k_\zeta \frac{1}{2}\|e_\epsilon\|^2 + \frac{1}{2}[0_3^\top, e_\epsilon^\top] e_{v_f}$. Since the first term is always positive, we conclude that the equilibrium point $(e_p, e_{v_f}, e_\varphi, e_\epsilon) = (0_3, 0_6, -1, 0_3)$ is unstable and, therefore, the system will converge to the configuration $(e_p, e_{v_f}, e_\varphi, e_\epsilon) = (0_3, 0_6, 1, 0_3)$.

□

Remark 4.1 (Unwinding)**.** Note that the two configurations where $e_\varphi = 1$ and $e_\varphi = -1$, respectively, represent the same orientation. The closed loop dynamics of e_φ, as given in (4.19b), can be written, in view of (4.25), as $\dot{e}_\varphi = k_\zeta \frac{1}{2}\|e_\epsilon\|^2 + \frac{1}{2}[0_3^\top, e_\epsilon^\top] e_{v_f}$. Since the first term is always positive, we conclude that the equilibrium point $(e_p, e_{v_f}, e_\varphi, e_\epsilon) = (0_3, 0_6, -1, 0_3)$ is unstable. Therefore, there might be trajectories close to the configuration $e_\varphi = -1$ that will move away and approach $e_\varphi = 1$, i.e., a full rotation will be performed to reach the desired orientation (of course, if the system starts at the equilibrium $(e_p, e_{v_f}, e_\varphi, e_\epsilon) = (0_3, 0_6, -1, 0_3)$, it will stay there, which also corresponds to the desired orientation behavior). This is the so-called *unwinding phenomenon* [70]. Note, however, that the desired equilibrium point $(e_p, e_{v_f}, e_\varphi, e_\epsilon) = (0_3, 0_6, 1, 0_3)$ is ***eventually attractive***, meaning that for each $\delta_\varepsilon > 0$, there exist finite a time instant $T \geq 0$ such that $1 - e_\varphi(t) < \delta_\varepsilon, \forall t > T \geq 0$. A similar behavior is observed if we stabilize the point $e_\varphi = -1$ instead of $e_\varphi = 1$, by setting $e := [e_p^\top, e_\epsilon^\top]^\top$ in (4.25) and considering the term $2(1 + e_\varphi)$ instead of $2(1 - e_\varphi)$ in the Lyapunov function (4.30).

In order to avoid the unwinding phenomenon, instead of the error $e = [e_p^\top, -e_\epsilon^\top]^\top$, we can instead choose $e = [e_p^\top, -e_\varphi e_\epsilon^\top]^\top$ (see [14]). Then by considering the Lyapunov function

$$\begin{aligned} V(\chi) = {} & \tfrac{1}{2}e_p^\top e_p + 1 - e_\varphi^2 + \tfrac{1}{2}e_{v_f}^\top \widetilde{M}(x) e_{v_f} + \tfrac{1}{2}e_\vartheta^\top \Gamma^{-1} e_\vartheta + \tfrac{1}{2}e_{\vartheta_O}^\top \Gamma_O^{-1} e_{\vartheta_O} \\ & + \tfrac{1}{2}e_d^\top B^{-1} e_d + \tfrac{1}{2}e_{d_O}^\top B_O^{-1} e_{d_O}, \end{aligned}$$

and the design (4.25), (4.28), and (4.29), we conclude by proceeding with a similar analysis that $(e_p, \|e_\epsilon\| e_\varphi, e_{v_f}) \to (0_3, 0, 0_6)$, which implies that the system is asymptotically driven to either the configuration $(e_p, e_{v_f}, e_\varphi, e_\epsilon) = (0_3, 0_6, \pm 1, 0_3)$, which is the desired one, or a configuration $(e_p, e_{v_f}, e_\varphi, e_\epsilon) = (0_3, 0_6, 0, \widetilde{e}_\epsilon)$, where $\widetilde{e}_\epsilon \in S^2$ is a unit vector. The latter represents a set of invariant undesired equilibrium points. The closed loop dynamics are the following:

$$\frac{\partial}{\partial t} e_\varphi = \frac{1}{2} e_\varphi \|e_\epsilon\|^2 + \frac{1}{2}[0_3^\top, e_\epsilon^\top] e_v, \tag{4.31a}$$

$$\frac{\partial}{\partial t}\left(\|e_\epsilon\|^2\right) = -e_\varphi^2 \|e_\epsilon\|^2 - e_\varphi [0_3^\top, e_\epsilon^\top] e_v. \tag{4.31b}$$

We can conclude from the term $[0_3^\top, e_\epsilon^\top] e_v$ in (4.31) that there exist trajectories that can bring the system close to the undesired equilibrium, rendering thus the point $(e_p, e_{v_f}, e_\varphi, e_\epsilon) = (0_3, 0_6, \pm 1, 0_3)$ only locally asymptotically stable. It has been proven that $e_\varphi = \pm 1$ cannot be globally stabilized with a purely continuous controller [70]. Discontinuous control laws have also been proposed (e.g., [71]), whose combination with adaptation techniques constitute part of our future research directions. Another possible direction is tracking directly on $SO(3)$ (see e.g., [69]).

Remark 4.2. Notice also that the control protocol compensates the uncertain dynamic parameters and external disturbances through the adaptation laws (4.29), although the errors (4.23), (4.24) do not converge to zero, but remain bounded. Finally, the control gains k_p, k_ζ, K_{v_i} can be tuned appropriately so that the proposed control inputs do not reach motor saturations in real scenarios.

4.4.2 Prescribed Performance Control

In this section, we adopt the concepts and techniques of prescribed performance control, recently proposed in [29], in order to achieve predefined transient and steady state response for the derived error, as well as ensure that $\theta_O(t) \in (-\frac{\pi}{2}, \frac{\pi}{2}), \forall t \in \mathbb{R}_{\geq 0}$. As stated in Section 2.1, prescribed performance characterizes the behavior where a signal evolves strictly within a predefined region that is bounded by absolutely decaying functions of time, called performance functions. This signal is represented by the object's pose error

$$e_s := \begin{bmatrix} e_{s_x} \\ e_{s_y} \\ e_{s_z} \\ e_{s_\phi} \\ e_{s_\theta} \\ e_{s_\psi} \end{bmatrix} := x_O - x_{\mathrm{d}}(t) \tag{4.32}$$

First, we reformulate Assumption 4.4, which is now required to be less strict, stating that the functions d_O, d_i are bounded:

Assumption 4.5. (Uncertainties/Disturbance bound)

- For each fixed $t \in \mathbb{R}_{\geq 0}$, the functions $(x_O, \dot{x}_O) \to d_O(x_O, \dot{x}_O, t)$ and $(q_i, \dot{q}_i) \to d_O(q_i, \dot{q}_i, t)$ are continuous, $\forall i \in \mathcal{N}$.
- There exist positive, finite *unknown* constants $\bar{d}_O, \bar{d}_i$ such that, for each fixed $(x_O, \dot{x}_O) \in \mathbb{M} \times \mathbb{R}^6$ and $(q_i, \dot{q}_i) \in \mathbb{R}^{2n_i}$, the functions $t \to d_O(x_O, \dot{x}_O, t)$ and $t \to d_O(q_i, \dot{q}_i, t)$ are bounded by $\bar{d}_O$ and $\bar{d}_i$, respectively, i.e., $\|d_O(x_O, \dot{x}_O, t)\| \leq \bar{d}_O$, and $\|d_i(q_i, \dot{q}_i, t)\| \leq \bar{d}_i$, $\forall t \in \mathbb{R}_{\geq 0}$, $i \in \mathcal{N}$.

The mathematical expressions of prescribed performance are given by the following inequalities:

$$-\rho_{s_k}(t) < e_{s_k}(t) < \rho_{s_k}(t), \forall k \in \mathcal{K}, \tag{4.33}$$

where $\mathcal{K} := \{x, y, z, \phi, \theta, \psi\}$ and $\rho_k : \mathbb{R}_{\geq 0} \to \mathbb{R}_{>0}$, with

$$\rho_{s_k}(t) := (\rho_{s_k,0} - \rho_{s_k,\infty}) \exp(-l_{s_k} t) + \rho_{s_k,\infty}, \ \forall k \in \mathcal{K}, \tag{4.34}$$

are designer-specified, smooth, bounded and decreasing positive functions of time with $l_{s_k}, \rho_{s_k,\infty}, k \in \mathcal{K}$, positive parameters incorporating the desired transient and steady state performance respectively. The terms $\rho_{s_k,\infty}$ can be set arbitrarily small, achieving thus practical convergence of the errors to zero. Next, we propose a state feedback control protocol that does not incorporate any information on the agents' or the object's dynamics or the external disturbances and guarantees (4.33) for all $t \in \mathbb{R}_{\geq 0}$. Given the errors (4.32):

Step I-a. Select the corresponding functions ρ_{s_k} as in (4.34) with

(i) $\rho_{s_\theta,0} = \rho_{s_\theta}(0) = \theta^*, \rho_{s_k,0} = \rho_{s_k}(0) > |e_{s_k}(0)|, \forall k \in \mathcal{K}\backslash\{\theta\}$,

(ii) $l_{s_k} \in \mathbb{R}_{>0}, \forall k \in \mathcal{K}$,

(iii) $\rho_{s_k,\infty} \in (0, \rho_{s_k,0}), \forall k \in \mathcal{K}$,

where θ^* is a positive constant satisfying $\theta^* + \bar{\theta} < \frac{\pi}{2}$ and $\bar{\theta}$ is the desired trajectory bound (see statement of Problem 4.1)

Step I-b. Introduce the transformed states representing the normalized errors

$$\xi_s := \begin{bmatrix} \xi_{s_x} \\ \vdots \\ \xi_{s_\psi} \end{bmatrix} := [\rho_s(t)]^{-1} e_s, \tag{4.35}$$

where $\rho_s(t) := \text{diag}\{[\rho_{s_k}(t)]_{k\in\mathcal{K}}\} \in \mathbb{R}^{6\times6}$, as well as the transformed state functions $\varepsilon_s : (-1,1)^6 \to \mathbb{R}^6$, and signals $r_s : (-1,1)^6 \to \mathbb{R}^{6\times6}$, with

$$\varepsilon_s(\xi_s) := \begin{bmatrix} \varepsilon_{s_x}(\xi_{s_x}) \\ \vdots \\ \varepsilon_{s_\psi}(\xi_{s_\psi}) \end{bmatrix} := \begin{bmatrix} \ln\left(\frac{1+\xi_{s_x}}{1-\xi_{s_x}}\right) \\ \vdots \\ \ln\left(\frac{1+\xi_{s_\psi}}{1-\xi_{s_\psi}}\right) \end{bmatrix} \tag{4.36}$$

$$r_s(\xi_s) := \text{diag}\{[q_{s_k}]_{k\in\mathcal{K}}\} := \text{diag}\Big\{\Big[\frac{\partial \varepsilon_{v_k}(\xi_{s_k})}{\partial \xi_{s_k}}\Big]_{k\in\mathcal{K}}\Big\} = \text{diag}\Big\{\Big[\frac{2}{1-\xi_{s_k}^2}\Big]_{k\in\mathcal{K}}\Big\}, \tag{4.37}$$

and design the reference velocity vector $v_r : (-1,1)^6 \times \mathbb{R}_{\geq 0} \to \mathbb{R}^6$, with:

$$v_r(\xi_s, t) := -g_s J_{O,\text{inv}}\Big(\eta_\text{d}(t) + \rho_{s_\eta}(t)\xi_{s_\eta}\Big)[\rho_s(t)]^{-1} r_s(\xi_s)\varepsilon_s(\xi_s), \tag{4.38}$$

where $J_{O,\text{inv}} : \mathbb{T} \to \mathbb{R}^{6\times6}$ is the matrix inverse $J_{O,\text{inv}}(\eta_O) := [J_O(\eta_O)]^{-1}$, $\rho_{s_\eta}(t) := \text{diag}\{\rho_{s_\phi}(t), \rho_{s_\theta}(t), \rho_{s_\psi}(t)\}$, $\xi_{s_\eta} := [\xi_{s_\phi}, \xi_{s_\theta}, \xi_{s_\psi}]^\top$, and we have further used the relation $\xi_s = [\rho_s(t)]^{-1}(x_O - x_\text{d}(t))$ from (4.32) and (4.35).

Step II-a. Define the velocity error vector

$$e_v := \begin{bmatrix} e_{v_x} \\ \vdots \\ e_{v_\psi} \end{bmatrix} := v_O - v_r(\xi_s, t), \tag{4.39}$$

and select the corresponding positive performance functions $\rho_{v_k} : \mathbb{R}_{\geq 0} \to \mathbb{R}_{>0}$ with $\rho_{v_k}(t) := (\rho_{v_k,0} - \rho_{v_k,\infty})\exp(-l_{v_k}t) + \rho_{v_k,\infty}$, such that $\rho_{v_k,0} = \|e_v(0)\| + \alpha, l_{v_k} > 0$ and $\rho_{v_k,\infty} \in (0, \rho_{v_k,0}), \forall k \in \mathcal{K}$, where α an is arbitrary positive constant.

Step II-b. Define the normalized velocity error

$$\xi_v := \begin{bmatrix} \xi_{v_x} \\ \vdots \\ \xi_{v_\psi} \end{bmatrix} := [\rho_v(t)]^{-1} e_v, \tag{4.40}$$

where $\rho_v(t) := \text{diag}\{[\rho_{v_k}(t)]_{k\in\mathcal{K}}\}$, as well as the transformed states $\varepsilon_v : (-1,1)^6 \to \mathbb{R}^6$ and signals $r_v : (-1,1)^6 \to \mathbb{R}^{6\times 6}$, with

$$\varepsilon_v(\xi_v) := \begin{bmatrix} \varepsilon_{v_x}(\xi_{v_x}) \\ \vdots \\ \varepsilon_{v_\psi}(\xi_{v_\psi}) \end{bmatrix} := \begin{bmatrix} \ln\left(\frac{1+\xi_{v_x}}{1-\xi_{v_x}}\right) \\ \vdots \\ \ln\left(\frac{1+\xi_{v_\psi}}{1-\xi_{v_\psi}}\right) \end{bmatrix}$$

$$r_v(\xi_v) := \text{diag}\{[q_{v_k}]_{k\in\mathcal{K}}\} := \text{diag}\Big\{\Big[\frac{\partial \varepsilon_{v_k}(\xi_{v_k})}{\partial \xi_{v_k}}\Big]_{k\in\mathcal{K}}\Big\} = \text{diag}\Big\{\Big[\frac{2}{1-\xi_{v_k}^2}\Big]_{k\in\mathcal{K}}\Big\},$$

and design the distributed control protocol for each agent $i \in \mathcal{N}$ as $u_i : \mathbb{R}^{n_i} \times (-1,1)^6 \times \mathbb{R}_{\geq 0} \to \mathbb{R}^6$:

$$u_i(q_i, \xi_v, t) := -c_i g_v [J_{O_i}(q_i)]^\top [\rho_v(t)]^{-1} r_v(\xi_v)\varepsilon_v(\xi_v), \tag{4.41}$$

where g_v is a positive constant gain, J_{O_i} as defined in (4.11), and c_i the load sharing coefficients that were also used in (4.28).

The control laws (4.41) can be written in vector form $u := [u_1^\top, \ldots, u_N^\top]^\top$, with:

$$u(q, \xi_v, t) = -C_g G^*(q)[\rho_v(t)]^{-1} r_v(\xi_v)\varepsilon_v(\xi_v), \tag{4.42}$$

where $G^\star(q) := [J_{O_1}(q_1)]^{-1}, \ldots, [J_{O_N}(q_N)]^{-1}]^\top \in \mathbb{R}^{6N\times 6}$ (recall that, due to the rigidity assumption, the matrices $J_{O_i}(q_i)$ are invertible, for all $i \in \mathcal{N}$), and $C_g := g_v \text{diag}\{[c_i I_6]_{i\in\mathcal{N}}\} \in \mathbb{R}^{6N\times 6N}$.

Remark 4.3. Notice from (4.28), (4.29), and (4.41) that in both control methodologies each agent $i \in \mathcal{N}$ can calculate its own control signal, without communicating with the rest of the team, rendering thus the overall control scheme decentralized. In fact, it needs feedback only from its own state $q_i, \dot{q}_i$, knowledge of the object geometric characteristics, the desired object profile $p_\text{d}(t), \eta_\text{d}(t)$, as well as the terms l_k, $\rho_{k,0}$, $\rho_{k,\infty}$, α, l_{v_k}, and $\rho_{v_k,\infty}$, $k \in \mathcal{K}$, since the performance functions concern the object pose error and hence are common for all the agents. Moreover, both schemes guarantee robustness to uncertainties of model uncertainties and external disturbances. In particular, note that the Prescribed Performance Control protocol does not even require the structure of the terms $\widetilde{M}, \widetilde{C}, \widetilde{g}, \widetilde{d}$, but only the positive definiteness of $\widetilde{M}$, as will be observed in the subsequent proof of Theorem 4.2. It is worth noting that, in the case that one or more agents failed to participate in the task, then the remaining agents would need to appropriately update their control protocols (e.g., update the load-sharing coefficients c_i) to compensate for the failure.

Remark 4.4. Internal force regulation can be also guaranteed by including in the control laws (4.28) and (4.41) a term of the form $(I_{6N} - \frac{1}{N}G^{\star}(q)[G(q)]^{\top})\hat{f}_{\text{int,d}}$, where$\hat{f}_{\text{int,d}} \in \mathbb{R}^{6N}$ is a constant vector representing desired internal forces, that can be transmitted off-line to the agents. The computation, though, of $G^{\star}(q)[G(q)]^{\top}$, by each agent, requires knowledge of all the grasping points p_{E_i} , which reduces to knowledge of the offsets $p^{O}_{E_i/O}$ (since all the agents can compute R_O and p_O), that can be also transmitted off-line to the agents.

The main results of this subsection are summarized in the following theorem.

Theorem 4.2. *Consider N agents rigidly grasping an object with unknown coupled dynamics* (4.15)*. Then, under Assumptions 4.1-4.3, 4.5, the distributed control protocol* (4.35)-(4.41) *guarantees that $-\rho_{s_k}(t) < e_{s_k}(t) < \rho_{s_k}(t), \forall k \in \mathcal{K}, t \in \mathbb{R}_{\geq 0}$ from all the initial conditions satisfying $|\theta(0) - \theta_d(0)| < \theta^*$ (where θ^* was used in* ***Step I-a*** *(i)), with all closed loop signals being bounded.*

Proof. Note first from (4.32), (4.35), (4.39), and (4.40), that the states x_O, v_O can be expressed as

$$x_O = x_{\text{d}}(t) + \rho_s(t)\xi_s, \tag{4.43a}$$

$$v_O = \rho_v(t)\xi_v + v_r(\xi_s, t), \tag{4.43b}$$

which was used in (4.38) and will be also used in the sequel.

Consider the combined state $\sigma = [q, \xi_s, \xi_v] \in \mathbb{R}^{n+12}$. From the differential kinematics $v_i = J_i(q_i)\dot{q}_i$, Assumption 4.3, as well as (4.10) and (4.39), (4.40), we can derive

$$\dot{q} = \widetilde{J}(q)v = \widetilde{J}(q)G(q)v_O = \widetilde{J}(q)G(q)\Big(\rho_v(t)\xi_v + v_r(\xi_s, t)\Big) =: f_{\text{cl},q}(\sigma, t), \tag{4.44}$$

where $\widetilde{J}(q) := \text{diag}\{[(J_i(q_i))^{\top}(J_i(q_i)(J_i(q_i))^{\top})^{-1}]_{i\in\mathcal{N}}\} \in \mathbb{R}^{6N\times n}$. Next, we obtain from (4.35):

$$\dot{\xi}_s = [\rho_s(t)]^{-1}\Big(\dot{e} - \dot{\rho}_s(t)\xi_s\Big) = [\rho_s(t)]^{-1}\Big(\dot{x}_O - \dot{x}_{\text{d}}(t) - \dot{\rho}_s(t)\xi_s\Big),$$

which, after employing (4.7a), (4.32), (4.38), as well as (4.43), becomes

$$\begin{aligned}\dot{\xi}_s =& [\rho_s(t)]^{-1}\Big[J_O\Big(\eta_{\text{d}}(t) + \rho_{s_\eta}(t)\xi_{s_\eta}\Big)\rho_v(t)\xi_v - \dot{\rho}_s(t)\xi_s - g_s[\rho_s(t)]^{-1}r_s(\xi_s)\varepsilon_s(\xi_s) \\ & - \dot{x}_{\text{d}}(t)\Big] =: f_{\text{cl},s}(\sigma, t).\end{aligned} \tag{4.45}$$

Consider now the derivative of $J_{O,\text{inv}}(\eta_O)$ (as was defined in (4.38)) as $J^d_{O,\text{inv}} : \mathbb{M} \times \mathbb{R}^6 \to \mathbb{R}^{6\times 6}$, with $J^d_{O,\text{inv}}(\eta_O, \dot{x}_O) := \dot{J}_{O,\text{inv}}(\eta_O)$, which, by employing (4.43), can be expressed as $\widetilde{J}^d_{O,\text{inv}} : \mathbb{R}^{n+12} \times \mathbb{R}_{\geq 0} \to \mathbb{R}^{6\times 6}$, with

$$\widetilde{J}^d_{O,\text{inv}}(\sigma, t) := J^d_{O,\text{inv}}\Big(\eta_{\text{d}}(t) + \rho_{s_\eta}(t)\xi_{s_\eta}, J_O(x_{\text{d}}(t) + \rho_s(t)\xi_s)[\rho_v(t)\xi_v + v_r(\xi_s, t)]\Big).$$

Moreover, we define the derivative function of the signal $r_s(\xi_s)$ as $r_s^d : \mathbb{R}^{n+12}\times\mathbb{R}_{\geq 0} \to \mathbb{R}^{6\times 6}$, with $r_s^d(\sigma,t) \coloneqq \dot{r}_s(\xi_s)$, which can be explicitly written as

$$\begin{aligned} r_s^d(\sigma,t) =& \text{diag}\Big\{\Big[\frac{2\xi_{s_k}}{(1-\xi_{s_k})^2}\Big]_{k\in\mathcal{K}}\Big\}\sum_{k\in\mathcal{K}} \bar{E}_k\dot{\xi}_s\bar{e}_k \\ =& \text{diag}\Big\{\Big[\frac{2\xi_{s_k}}{(1-\xi_{s_k})^2}\Big]_{k\in\mathcal{K}}\Big\}\sum_{k\in\mathcal{K}} \bar{E}_k f_{\text{cl},s}(\sigma,t)\bar{e}_k, \end{aligned}$$

where $\bar{E}_k \in \mathbb{R}^{6\times 6}$ is the matrix with 1 in the position (k,k) and zeros everywhere else, and $\bar{e}_k \in \mathbb{R}^6$ is the vector with 1 in the position k and zeros everywhere else.

Hence, we can now define the derivative of the reference velocity v_r as $v_r^d : \mathbb{R}^{n+12} \to \mathbb{R}^6$, with $v_r^d(\sigma,t) \coloneqq \dot{v}_r(\xi_s,t)$, or, equivalently,

$$\begin{aligned} &v_r^d(\sigma,t) = -g_s J_{O,\text{inv}}\Big(\eta_\text{d}(t) + \rho_s(t)_{s_\eta}(t)\xi_{s_\eta}\Big)\Big[[\rho_s(t)]^{-1}r_s^d(\sigma,t)\varepsilon_s(\xi_s) \\ &+ [\rho_s(t)]^{-1}[r_s(\xi_s)]^2 f_{\text{cl},s}(\sigma,t) - [\rho_s(t)]^{-2}\dot{r}_s(t)r_s(\xi_s)\varepsilon_s(\xi_s)\Big] \\ &- g_s\widetilde{J}^d_{O,\text{inv}}(\sigma,t)[\rho_s(t)]^{-1}r_s(\xi_s)\varepsilon_s(\xi_s) \end{aligned} \tag{4.46}$$

Moreover, from (4.39) and (4.40) one obtains:

$$\begin{aligned} \dot{\xi}_v &= [\rho_v(t)]^{-1}\Big(\dot{e}_v - \dot{\rho}_v(t)\xi_v\Big), \\ &= [\rho_v(t)]^{-1}\Big(\dot{v}_O - v_r^d(\sigma,t) - \dot{\rho}_v(t)\xi_v\Big), \end{aligned}$$

which, after employing (4.15), (4.42), and the fact that $\sum_{i\in\mathcal{N}} c_i = 1$, becomes

$$\begin{aligned} \dot{\xi}_v =& [\rho_v(t)]^{-1}\Big(- \dot{\rho}_v(t)\xi_v - \widetilde{M}(x(\sigma,t))\Big[\widetilde{C}(x(\sigma,t))[\rho_v(t)\xi_v + v_r(\xi_s,t)] + \widetilde{g}(x(\sigma,t)) \\ &+ \widetilde{d}(x(\sigma,t),t) - g_v[\rho_v(t)]^{-1}r_v(\xi_v)\varepsilon_v(\xi_v)\Big] - v_r^d(\sigma,t)\Big) \eqqcolon f_{\text{cl},v}(\sigma,t) \end{aligned} \tag{4.47}$$

and where, with a slight abuse of notation and by using (4.43) and (4.44), we have written x (that was first defined in (4.15)) as a function of σ and t, i.e.,

$$x(\sigma,t) = \begin{bmatrix} q \\ \dot{q} \\ x_O \\ \dot{x}_O \end{bmatrix} = \begin{bmatrix} q \\ f_{\text{cl},q}(\sigma,t) \\ x_\text{d}(t) + \rho_s(t)\xi_s \\ J_O\Big(\eta_\text{d}(t) + \rho_{s_\eta}(t)\xi_{s_\eta}\Big)[\rho_v(t)\xi_v + v_r(\xi_s,t)] \end{bmatrix}.$$

Hence, we can write (4.44)-(4.47) in compact form

$$\dot{\sigma} = f_\text{cl}(\sigma,t) \coloneqq \begin{bmatrix} f_{\text{cl},q}(\sigma,t) \\ f_{\text{cl},s}(\sigma,t) \\ f_{\text{cl},v}(\sigma,t) \end{bmatrix}.$$

Consider now the open and nonempty set $\Omega \coloneqq \mathbb{R}^n \times (-1,1)^{12}$. The choice of the parameters $\rho_{s_k,0}$ and $\rho_{v_k,0}, k \in \mathcal{K}$ in **Step I-a** and **Step II-a**, respectively, along with the fact that the initial conditions satisfy $|\theta_O(0) - \theta_\mathrm{d}(0)| < \theta^*$ imply that $|e_{s_k}(0)| < \rho_{s_k}(0), |e_{v_k}(0)| < \rho_{v_k}(0), \forall k \in \mathcal{K}$ and hence $[[\xi_s(0)]^\top, [\xi_v(0)]^\top]^\top \in (-1,1)^{12}$. Moreover, it can be verified that $f_\mathrm{cl} : \Omega \times \mathbb{R}_{\geq 0} \to \mathbb{R}^{n+12}$ is locally Lipschitz in σ over the set Ω and is continuous in t, which makes it also locally integrable in t for each fixed $\sigma \in \Omega$. Therefore, the hypotheses of Theorem 2.3 hold and the existence of a maximal solution $\sigma : [0, \tau_\mathrm{max}) \to \Omega$, for $\tau_\mathrm{max} > 0$, is ensured. We thus conclude that

$$\xi_{s_k}(t) = \frac{e_{s_k}(t)}{\rho_{s_k}(t)} \in (-1,1), \tag{4.48a}$$

$$\xi_{v_k}(t) = \frac{e_{v_k}(t)}{\rho_{v_k}(t)} \in (-1,1), \tag{4.48b}$$

$\forall k \in \mathcal{K}, t \in [0, \tau_\mathrm{max})$, which also implies that $\|\xi_s(t)\| \leq \sqrt{6}$, and $\|\xi_v(t)\| \leq \sqrt{6}, \forall t \in [0, \tau_\mathrm{max})$. Next, we need to show the boundedness of all closed loop signals as well as that $\tau_\mathrm{max} = \infty$. Note first from (4.48), that $|\theta_O(t) - \theta_\mathrm{d}(t)| < \rho_\theta(t) \leq \rho_\theta(0) = \theta^*$, which, since $\theta_\mathrm{d}(t) \in [-\bar{\theta}, \bar{\theta}], \forall t \in \mathbb{R}_{\geq 0}$, implies that $|\theta_O(t)| \leq \widetilde{\theta} \coloneqq \bar{\theta} + \theta^* < \frac{\pi}{2}, \forall t \in [0, \tau_\mathrm{max})$. Therefore, by employing (4.8), one obtains

$$\|J_O(\eta_O(t))\| \leq \bar{J}_O \coloneqq \sqrt{\frac{|\sin(\widetilde{\theta})| + 1}{1 - \sin^2(\widetilde{\theta})}} < \infty, \ \forall t \in [0, \tau_\mathrm{max}). \tag{4.49}$$

Consider now the positive definite and radially unbounded function $V_s : \mathbb{R}^6 \to \mathbb{R}_{\geq 0}$, with $V_s(\varepsilon_s(\xi_s)) = \frac{1}{2}\|\varepsilon_s(\xi_s)\|^2$, and its derivative along the solutions of the closed loop system, which, in view of (4.45), yields

$$\begin{aligned}
\dot{V}_s(\varepsilon_s(\xi_s)) = &-g_s\|[\rho_s(t)]^{-1}r_s(\xi_s)\varepsilon_s(\xi_s)\|^2 + [\varepsilon_s(\xi_s)]^\top r_s(\xi_s)[\rho_s(t)]^{-1}\Big(- \dot{\rho}_s(t)\xi_s \\
&- \dot{x}_\mathrm{d}(t) + J_O(\eta_\mathrm{d}(t) + \rho_{s_\eta}(t)\xi_{s_\eta})\rho_v(t)\xi_v\Big) \\
\leq &\ g_s\|[\rho_s(t)]^{-1}r_s(\xi_s)\varepsilon_s(\xi_s)\|^2 + \|[\rho_s(t)]^{-1}r_s(\xi_s)\varepsilon_s(\xi_s)\|\Big(\|\dot{x}_\mathrm{d}(t)\| + \|\dot{\rho}_s(t)\xi_s\| + \\
&\|J_O(\eta_\mathrm{d}(t) + \rho_{s_\eta}(t)\xi_{s_\eta})\rho_v(t)\xi_v\|\Big).
\end{aligned}$$

In view of (4.49), (4.48), and the structure of $\rho_{s_k}, \rho_{v_k}, k \in \mathcal{K}$, as well as the fact that $v_O(0) = 0$ and the boundedness of $\dot{x}_\mathrm{d}(t)$, the last inequality becomes

$$\dot{V}_s(\varepsilon_s(\xi_s)) \leq - g_s\|[\rho_s(t)]^{-1}r_s(\xi_s(t))\varepsilon_s(\xi_s(t))\|^2 + \|[\rho_s(t)]^{-1}r_s(\xi_s(t))\varepsilon_s(\xi_s(t))\|\bar{B}_s,$$

$\forall t \in [0, \tau_\mathrm{max})$, where

$$\bar{B}_s \coloneqq \sqrt{6}\bar{J}_O(\|v_r^0\| + \alpha) + \bar{\dot{x}}_\mathrm{d} + \sqrt{6}\max_{k \in \mathcal{K}}\{l_k(\rho_{s_k,0} - \rho_{s_k,\infty})\},$$

is a positive constant independent of $\tau_{\max}$, $\bar{\dot{x}}_{\mathrm{d}}$ is the bound of $\dot{x}_{\mathrm{d}}(t)$, and $v_r^0 := v_r(\xi_s(0), 0)$. Therefore, $\dot{V}_s(\varepsilon_s(\xi_s))$ is negative when $\|[\rho_s(t)]^{-1} r_s(\xi_s(t))\varepsilon_s(\xi_s(t))\| > \frac{\bar{B}_s}{g_s}$, which, by employing (4.37), the decreasing property of $\rho_{s_k}(t), k \in \mathcal{K}$ as well as (4.48a), is satisfied when $\|\varepsilon_s(\xi_s(t))\| > \frac{\max_{k\in\mathcal{K}}\{\rho_{s_k,0}\}\bar{B}_s}{2g_s}$. Hence, we conclude that

$$\|\varepsilon_s(\xi_s(t))\| \leq \bar{\varepsilon}_s := \max\left\{\|\varepsilon_s(\xi_s(0))\|, \frac{\max\limits_{k\in\mathcal{K}}\{\rho_{s_k,0}\}\bar{B}_s}{2g_s}\right\}, \tag{4.50}$$

$\forall t \in [0, \tau_{\max})$. Furthermore, since $|\varepsilon_{s_k}(\xi_{s_k})| \leq \|\varepsilon_s(\xi_s)\|, \forall k \in \mathcal{K}$, taking the inverse logarithm function from (4.36), we obtain

$$-1 < \frac{\exp(-\bar{\varepsilon}_s) - 1}{\exp(-\bar{\varepsilon}_s) + 1} =: -\bar{\xi}_s \leq \xi_{s_k}(t) \leq \bar{\xi}_s := \frac{\exp(\bar{\varepsilon}_s) - 1}{\exp(\bar{\varepsilon}_s) + 1} < 1, \tag{4.51}$$

$\forall t \in [0, \tau_{\max})$. Hence, recalling (4.37), one obtains

$$\|r_s(\xi_s(t))\| \leq \bar{r}_s := \frac{2}{1 - \bar{\xi}_s^2} = \frac{(\exp(\bar{\varepsilon}_s) + 1)^2}{2\exp(\bar{\varepsilon}_s)},$$

$\forall t \in [0, \tau_{\max})$. Therefore, we obtain from (4.38) the boundedness of v_r with

$$\|v_r(\xi_s(t), t)\| \leq \bar{v}_r := g_s \bar{J}_O \frac{\bar{\varepsilon}_s(\exp(\bar{\varepsilon}_s) + 1)^2}{2\min\limits_{k\in\mathcal{K}}\{\rho_{s_k,\infty}\}\exp(\bar{\varepsilon}_s)}, \tag{4.52}$$

$\forall t \in [0, \tau_{\max})$. Since $v_O = v_r(\xi_s, t) + \rho_v(t)\xi_v$, we also conclude that

$$\|v_O(t)\| \leq \bar{v}_O := g_s \bar{J}_O \frac{\bar{\varepsilon}_s(\exp(\bar{\varepsilon}_s) + 1)^2}{2\min\limits_{k\in\mathcal{K}}\{\rho_{k,\infty}\}\exp(\bar{\varepsilon}_s)} + \sqrt{6}\max_{k\in\mathcal{K}}\{\rho_{v_k,0}\}, \tag{4.53}$$

$\forall t \in [0, \tau_{\max})$, which, through (4.10) and (4.13), leads to

$$\|v_i(t)\| \leq \bar{v}_i := (\|p^{E_i}_{O/E_i}\| + 1)\bar{v}_O, \forall i \in \mathcal{N}, t \in [0, \tau_{\max}). \tag{4.54}$$

In a similar vein, we can also derive a bound for the derivative of the reference velocity (4.46), $\|v_r^d(\sigma(t), t)\| \leq \bar{v}_r^d, \forall t \in [0, \tau_{\max})$, which is not written explicitly for presentation clarity. From (4.7), (4.51) and (4.32) we also conclude that $\|x_O(t)\| \leq \bar{x}_O := \bar{x}_{\mathrm{d}} + \sqrt{6}\bar{\xi}_s \max\limits_{k\in\mathcal{K}}\{\rho_{s_k,0}\}, t \in [0, \tau_{\max})$, as well as $\|\dot{x}_O(t)\| \leq \bar{J}_O \bar{v}_O$.

Applying the aforementioned line of proof, we consider the positive definite and radially unbounded function $V_v : \mathbb{R}^6 \to \mathbb{R}_{\geq 0}$, with $V_v(\varepsilon_v(\xi_v)) = \frac{1}{2}\|\varepsilon_v(\xi_v)\|^2$, and its derivative along the solutions of the closed loop system, which, in view of (4.47), yields

$$\begin{aligned}
\dot{V}_v(\varepsilon_v(\xi_v)) &= -g_v[\varepsilon_v(\xi_v)]^\top r_v(\xi_v)[\rho_v(t)]^{-1}\widetilde{M}(x(\sigma, t))[\rho_v(t)]^{-1} r_v(\xi_v)\varepsilon_v(\xi_v) \\
&+ [\varepsilon_v(\xi_v)]^\top r_v(\xi_v)[\rho_v(t)]^{-1}\Big(- \dot{\rho}_v(t)\xi_v - \widetilde{M}(x(\sigma, t))\Big[\widetilde{C}(x(\sigma, t))[\rho_v(t)\xi_v + v_r(\xi_s, t)] \\
&+ \widetilde{g}(x(\sigma, t)) + \widetilde{d}(x(\sigma, t), t)\Big] - v_r^d(\sigma, t)\Big),
\end{aligned} \tag{4.55}$$

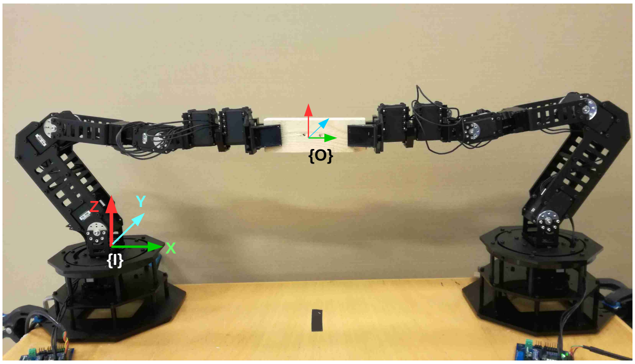


Figure 4.2: Two WidowX Robot Arms rigidly grasping an object.

By using (4.9) and the fact that the rotation matrix $R_{E_i}(q_i)$ is an orthogonal matrix, we obtain $\|x_{E_i}(t)\| := \|[p_{E_i}^\top(q_i(t), \eta_{E_i}^\top(q_i(t))]^\top\| \leq \|x_O(t)\| + [(p^{E_i}_{E_i/O})^\top, \eta^\top_{E_i/O}]^\top$ and hence, in view of the inverse kinematics of the agents [26], we conclude the boundedness of $q(t)$ as

$$\|q(t)\| \leq \bar{q}, \forall t \in [0, \tau_{\max}), \tag{4.56}$$

where $\bar{q}$ is a positive constant. From Assumption 4.3 and the forward differential agent kinematics, we can also conclude that there exists a positive constant $\bar{J}$ such that $\|\dot{q}(t)\| \leq \bar{J}\|v\| \leq \bar{J}\sum_{i\in\mathcal{N}} \bar{v}_i, \forall t \in [0, \tau_{\max})$, where $\bar{v}_i$ was defined in (4.54). Therefore, we conclude that

$$\|x(t)\| \leq \bar{x} := \bar{q} + \bar{J}\sum_{i\in\mathcal{N}} \bar{v}_i + \bar{x}_O + \bar{J}_O\bar{v}_O, \tag{4.57}$$

$\forall t \in [0, \tau_{\max})$.

Invoking Assumption 4.5 and the boundedness of the states $q_i(t)$, $\dot{q}_i(t)$, $x_O(t)$, $\dot{x}_O(t)$, $\forall t \in [0, \tau_{\max})$, we conclude the boundedness of $d_O(x_O(t), \dot{x}_O(t), t)$ and $d_i(q_i(t), \dot{q}_i(t), t)$, $\forall t \in [0, \tau_{\max})$, by positive and finite constants $\bar{d}'_O$ and $\bar{d}'_i$, respectively, $\forall i \in \mathcal{N}$. Hence, from expressions (4.13) and (4.16d), we obtain $\|\widetilde{d}(x(t))\| \leq \bar{d} := \bar{d}'_O + \sum_{i\in\mathcal{N}}\{\|p^{E_i}_{O/E_i}\| + 1\}\bar{d}'_i$.

In addition, since the terms $\widetilde{C}(x), \widetilde{g}(x)$ are continuous, we conclude from (4.57) that there exist positive constants $\bar{c}, \bar{g}$, independent of $\tau_{\max}$ (since $\bar{x}$ is also independent of $\tau_{\max}$), such that $\|\widetilde{C}(x(t))\| \leq \bar{c}$, $\|\widetilde{g}(x(t))\| \leq \bar{g}$, $\forall t \in [0, \tau_{\max})$.

Thus, by combining the aforementioned results along with the boundedness of v_r, (4.48), (4.52), (4.57) as well as (4.17), we obtain from (4.55)

$$\dot{V}_v(\varepsilon_v(\xi_v)) \leq -g_v\underline{m}\|[\rho_v(t)]^{-1}r_v(\xi_v(t))\varepsilon_v(\xi_v(t))\|^2 + \|[\rho_v(t)]^{-1}r_v(\xi_v(t))\varepsilon_v(\xi_v(t))\|\bar{B}_v,$$

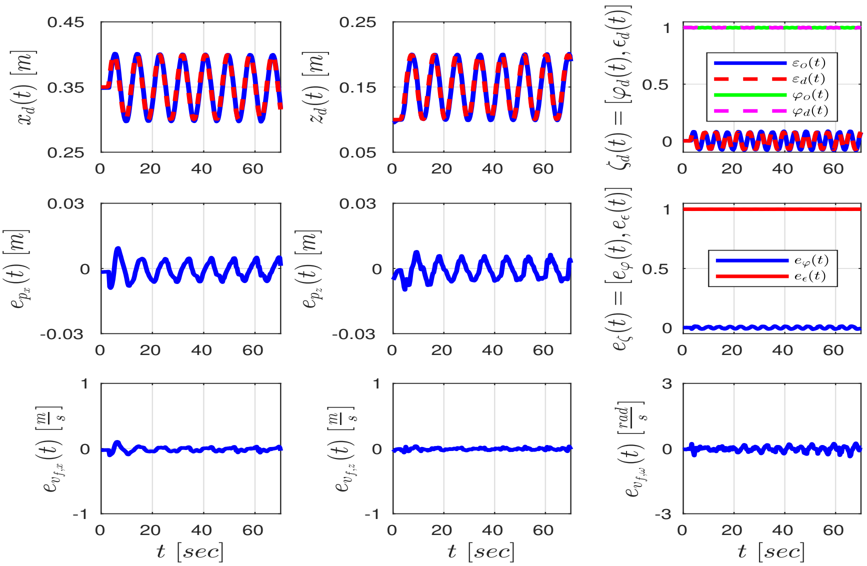


Figure 4.3: Simulations results for the controller of subsection 4.4.1, for $t \in [0, 70]$ [sec]. Top: The desired (with blue) and actual (with red) object trajectory in x- and z-axis as well as the quaternion desired and actual object trajectory. Middle: The position and quaternion errors $e_p(t), e_\zeta(t)$, respectively. Bottom: The velocity errors $e_{v_f}(t)$.

$\forall t \in [0, \tau_{\max})$, where

$$\begin{aligned}\bar{B}_v :=& \sqrt{6} \max_{k \in \mathcal{K}} \{ l_{v_k} (\rho_{v_k,0} - \rho_{v_k,\infty}) \} + \bar{v}_r^d + \bar{m} \Big(\bar{g} + \bar{d} + \\ & \bar{c}(\bar{v}_r + \sqrt{6}(\|v_r^0\| + \alpha)) \Big). \end{aligned} \tag{4.58}$$

By proceeding similarly as with $\dot{V}(\varepsilon_s(\xi_s))$, we conclude that

$$\|\varepsilon_v(\xi_v(t))\| \leq \bar{\varepsilon}_v := \max \left\{ \|\varepsilon_v(\xi_v(0))\|, \frac{\max\limits_{k \in \mathcal{K}} \{\rho_{v_k,0}\} \bar{B}_v}{2 g_v \underline{m}} \right\}, \tag{4.59}$$

$\forall t \in [0, \tau_{\max})$, from which we obtain

$$-1 < \frac{\exp(-\bar{\varepsilon}_v) - 1}{\exp(-\bar{\varepsilon}_v) + 1} =: -\bar{\xi}_v \leq \xi_{v_k}(t) \leq \bar{\xi}_v := \frac{\exp(\bar{\varepsilon}_v) - 1}{\exp(\bar{\varepsilon}_v) + 1} < 1, \tag{4.60}$$

as well as

$$\|r_v(\xi_v(t))\| \leq \bar{r}_v := \frac{2}{1 - \bar{\xi}_{v_k}^2} = \frac{(\exp(\bar{\varepsilon}_v) + 1)^2}{2 \exp(\bar{\varepsilon}_v)},$$

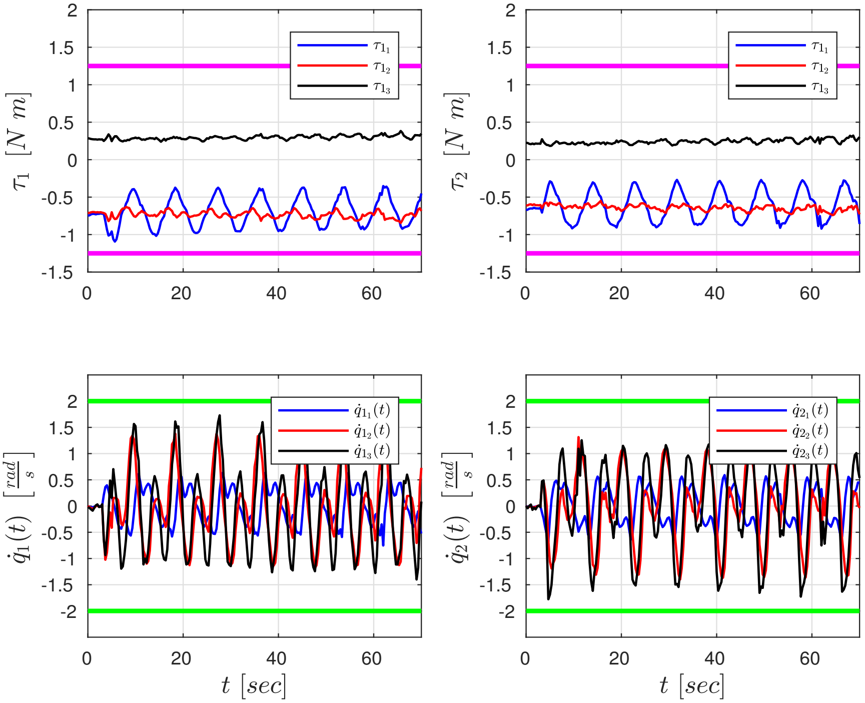


Figure 4.4: The agent joint torques and velocities of the simulation of the controller in subsection 4.4.1, for $t \in [0, 70]$ [sec], with their respective limits (purple and green lines, respectively). Top: The joint torques of agent 1 (left) and agent 2 (right). Bottom: The joint velocities of agent 1 (left) and agent 2 (right).

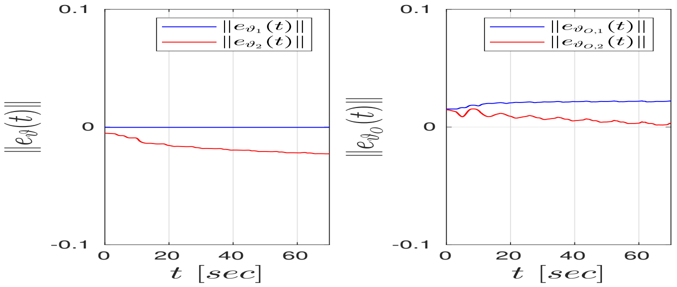


Figure 4.5: The norms of the adaptation signals $e_{\vartheta_i}(t)$ (left) and $e_{\vartheta_{O,i}}(t)$, $\forall i \in \{1, 2\}$, $t \in [0, 70]$ [sec] of the simulation of the controller in subsection 4.4.1.

$\forall t \in [0, \tau_{\max})$. Hence, we can also conclude the boundedness of the control inputs (4.41)

$$\|u_i(q_i(t), \xi_v(t), t)\| \leq \bar{u}_i := c_i g_v(\|p^{E_i}_{O/E_i}\| + 1) \max_{k \in \mathcal{K}} \left\{ \frac{1}{\rho_{v_k,\infty}} \right\} \bar{r}_v \bar{\varepsilon}_v, \ \forall t \in [0, \tau_{\max}). \tag{4.61}$$

What remains to be shown is that $\tau_{\max} = \infty$. To this end, note from (4.56), (4.51), (4.60), that the solution $\sigma(t)$ remains in a compact subset of $\Omega = \mathbb{R}^n \times (-1, 1)^{12}$, i.e.,

$$\sigma(t) \in \Omega' := [-\bar{q}, \bar{q}] \times [-\bar{\xi}_s, \bar{\xi}_s]^6 \times [-\bar{\xi}_v, \bar{\xi}_v]^6,$$

$\forall t \in [0, \tau_{\max})$. Hence, assuming $\tau_{\max} < \infty$ and since $\Omega' \subset \Omega$, Proposition 2.1 dictates the existence of a time instant $t' \in [0, \tau_{\max})$ such that $\sigma(t') \notin \Omega'$, which is a contradiction. Therefore, $\tau_{\max} = \infty$. Thus, all closed loop signals remain bounded and moreover $\sigma(t) \in \Omega' \subset \Omega, \forall t \in \mathbb{R}_{\geq 0}$. Finally, by multiplying (4.51) by $\rho_k(t), k \in \mathcal{K}$, we obtain

$$-\rho_{s_k}(t) < -\bar{\xi}_s \rho_{s_k}(t) \leq e_{s_k}(t) \leq \bar{\xi}_s \rho_{s_k}(t) < \rho_{s_k}(t), \tag{4.62}$$

$\forall t \in \mathbb{R}_{\geq 0}$, which leads to the conclusion of the proof. □

Remark 4.5. From the aforementioned proof it can be deduced that the Prescribed Performance Control scheme achieves its goal without resorting to the need of rendering the ultimate bounds $\bar{\varepsilon}_s, \bar{\varepsilon}_v$ of the modulated pose and velocity errors $\varepsilon_s(\xi_s(t)), \varepsilon_v(\xi_v(t))$ arbitrarily small by adopting extreme values of the control gains g_s and g_v (see (4.50) and (4.59)). More specifically, notice that (4.51) and (4.60) hold no matter how large the finite bounds $\bar{\varepsilon}_s, \bar{\varepsilon}_v$ are. In the same spirit, large uncertainties involved in the coupled model (4.15) can be compensated, as they affect only the size of $\bar{\varepsilon}_v$ through $\bar{B}_v$ (see (4.58)), but leave unaltered the achieved stability properties. Hence, the actual performance given in (4.62), which is solely determined by the designed-specified performance functions $\rho_{s_k}(t), \rho_{v_k}(t), k \in \mathcal{K}$, becomes isolated against model uncertainties, thus extending greatly the robustness of the proposed control scheme.

Remark 4.6 (Control Input Bounds). The aforementioned analysis of the Prescribed Performance Control methodology reveals the derivation of implicit bounds for the velocity v_i and control input u_i of each agent. More specifically, notice that (4.54) and (4.53) provide a bound for the agents' velocity, $\|v_i(t)\| \leq \bar{v}_i$. Therefore, given a bound for the agents' velocity $\bar{v}_{i,b}$ (derived from bounds on the joint velocities $\dot{q}_i$), $i \in \mathcal{N}$, the desired trajectory velocity bound $\bar{\dot{x}}_\text{d}$, as well as the initial velocity error v^0_r (which is proportional to $\varepsilon_s(\xi_s(0))$), we can tune appropriately the control gain g_s as well as the parameters $\rho_{s_k,0}, \rho_{v_k,0}, \rho_{s_k,\infty}, \rho_{v_k,\infty}, l_{s_k}, l_{v_k}$, and α, to achieve $\bar{v}_i \leq \bar{v}_{i,b}, \forall i \in \mathcal{N}$. In the same spirit, (4.61) provides a bound $\bar{u}_i$ for the control inputs of the agents. Hence, given bounds for the agents' inputs $\bar{u}_{i,b}$ (derived from bounds on the joint torques τ_i), $i \in \mathcal{N}$, if the upper bound term $\bar{B}_v$

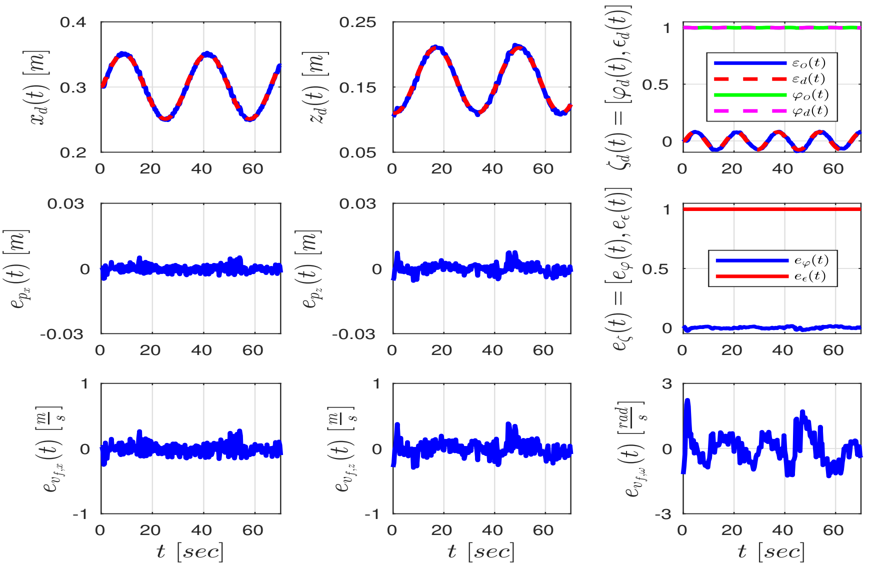


Figure 4.6: Experimental results for the controller of subsection 4.4.1, for $t \in [0, 70]$ [sec]. Top: The desired (with blue) and actual (with red) object trajectory in x- and z-axis as well as the quaternion desired and actual object trajectory. Middle: The position and quaternion errors $e_p(t), e_\zeta(t)$, respectively. Bottom: The velocity errors $e_{v_f}(t)$.

is known, we can further tune the control gain g_v as as well as the performance function parameters to achieve $\bar{u} \leq \bar{u}_{i,b}$. Explicit closed-loop expressions for the choice of these gains and parameters are beyond the scope of this paper and consist part of future work. It is also worth noting that the selection of the control gains g_s, g_v affects the evolution of the errors e, e_v inside the corresponding performance envelope.

4.5 Simulation and Experimental Results

In this section, we provide simulation and experimental results for the two developed control schemes. Firstly, in subsection 4.5.1 we present results from computer simulations using the realistic environment of V-REP [109] as well as experimental results using the adaptive control protocol developed in Section 4.4.1. Then, in subsection 4.5.2, we provide simulation and experimental results using the Prescribed Performance Control algorithm developed in Section 4.4.2.

The tested scenario consists of two WidowX Robot Arms [110] rigidly grasping a wooden cuboid object (see Fig. 4.2) that has to track a planar time trajectory $p_{\mathrm{d}}(t) = [x_{\mathrm{d}}(t), 0, z_{\mathrm{d}}(t)]^\top, \eta_{\mathrm{d}}(t) = [0, \theta_{\mathrm{d}}(t), 0]^\top$. For that purpose, we employ the three rotational -with respect to the y axis - joints of the arms. The lower joint

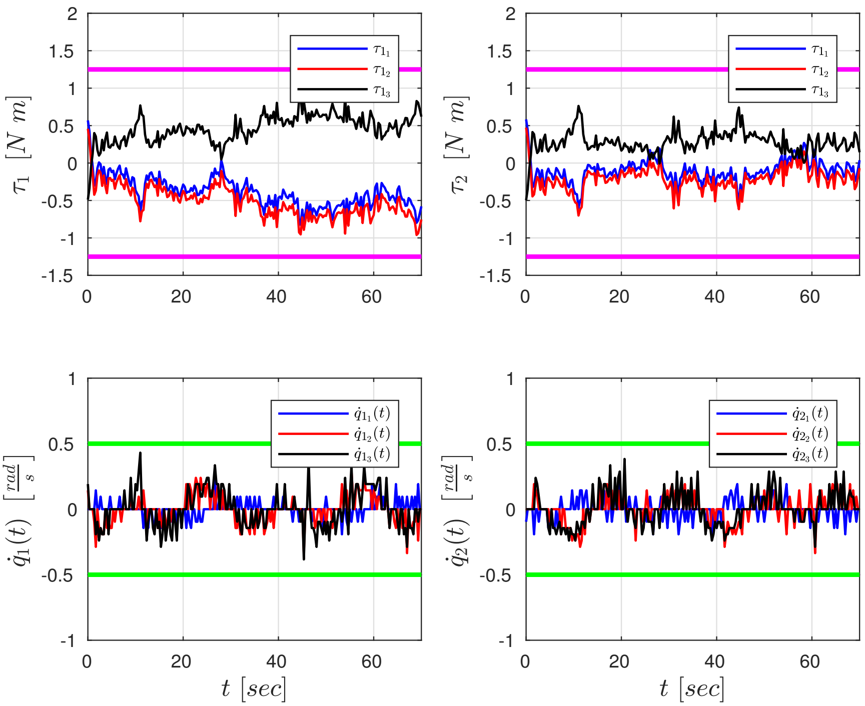


Figure 4.7: The agent joint torques and velocities of the experiment of the controller in subsection 4.4.1, for $t \in [0, 70]$ [sec], with their respective limits (purple and green lines, respectively). Top: The joint torques of agent 1 (left) and agent 2 (right). Bottom: The joint velocities of agent 1 (left) and agent 2 (right).

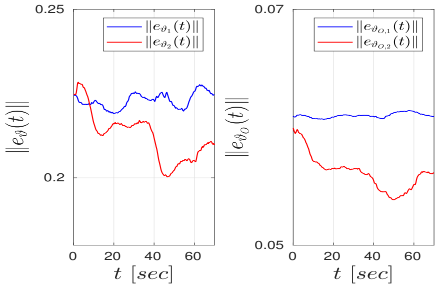


Figure 4.8: The norms of the adaptation signals $e_{\vartheta_i}(t)$ (left) and $e_{\vartheta_{O,i}}(t)$, $\forall i \in \{1, 2\}, t \in [0, 70]$ [sec] of the experiment of the controller in subsection 4.4.1.

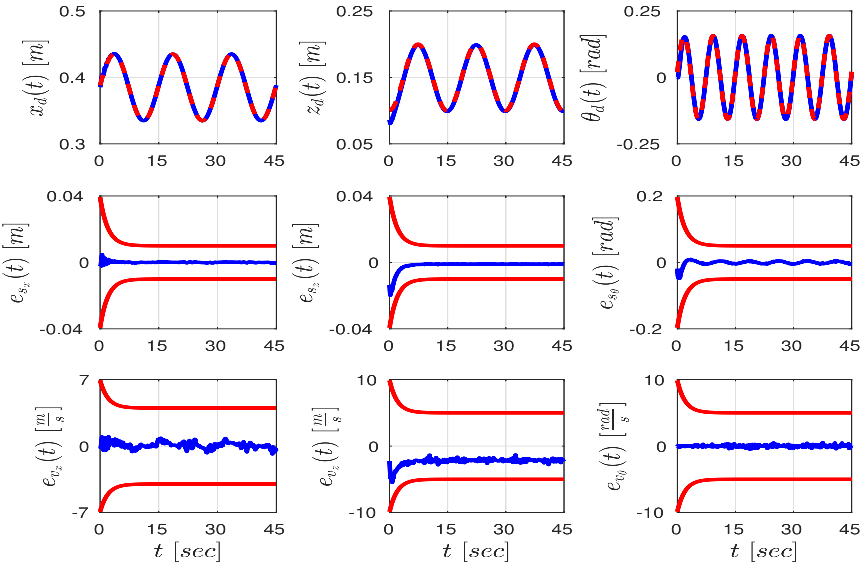


Figure 4.9: Simulations results for the controller of subsection 4.4.2, for $t \in [0, 45]$ [sec]. Top: The desired (with blue) and actual (with red) object trajectory in x-, z-axis, and around y-axis. Middle: The pose errors $e_s(t)$ (with blue), as well as the performance functions $\rho_s(t)$ (with red), respectively. Bottom: The velocity errors $e_v(t)$ (with blue), as well as the performance functions $\rho_v(t)$ (with red), respectively.

consists of a MX-64 Dynamixel Actuator, whereas each of the two upper joints consists of a MX-28 Dynamixel Actuator from the MX Series [111]. Both actuators provide feedback of the joint angle and rate $q_i, \dot{q}_i$, $\forall i \in \{1, 2\}$. The micro-controller used for the actuators of each arm is the ArbotiX-M Robocontroller [112], which is serially connected to an i-7 desktop computer with 4 cores and 16GB RAM. All the computations for the real-time experiments are performed at a frequency of 120 [Hz] and for the V-REP simulations at 60 [Hz]. Finally, we consider that the MX-64 motor can exert a maximum torque of 3 [Nm], and the MX-28 motors can exert a maximum torque of 1.25 [Nm], values that are slightly more conservative than the actual limits. In the same vein, we also assume a velocity bound of 0.5 [rad/s] for the experiments and 2 [rad/s] for the V-REP simulations. In all cases, we set the load sharing coefficients at $c_1 = 0.75$ and $c_2 = 0.25$, in order to demonstrate a potential difference in the power capabilities of the agents.

4.5.1 Adaptive Control with Quaternion Feedback

In this subsection, we present simulation and experimental results for the control protocol developed in Section 4.4.1.

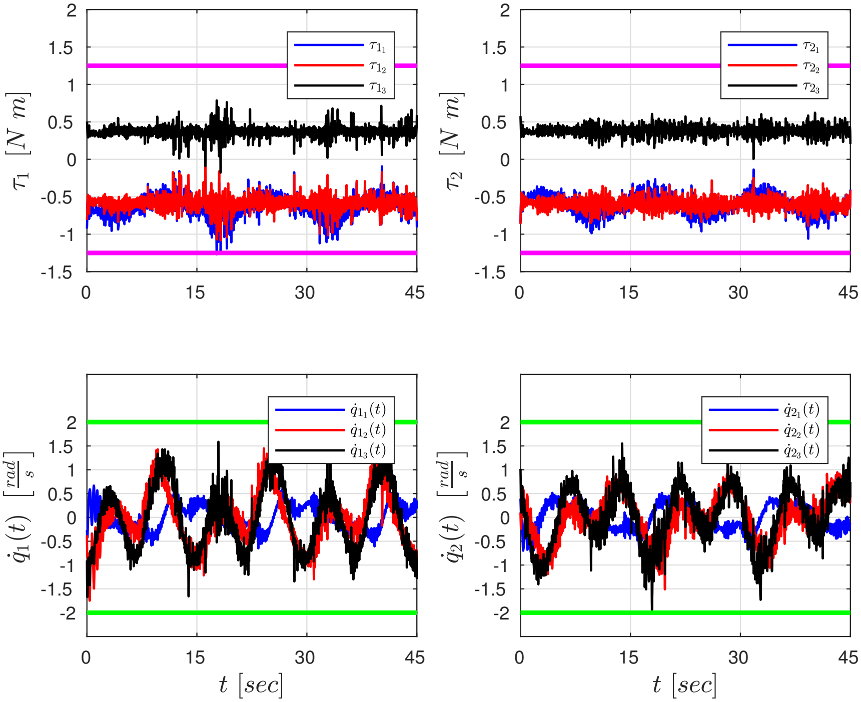


Figure 4.10: The agent joint torques and velocities of the simulation of the controller in subsection 4.4.2, for $t \in [0, 45]$ [sec], with their respective limits (purple and green lines, respectively). Top: The joint torques of agent 1 (left) and agent 2 (right). Bottom: The joint velocities of agent 1 (left) and agent 2 (right).

Simulation Results

The desired trajectory for the V-REP simulations is set to $x_{\mathrm{d}}(t) = 0.35+0.05\sin\left(\frac{2\pi t}{15}\right)$ [m], $z_{\mathrm{d}}(t) = 0.15-0.05\cos\left(\frac{2\pi t}{15}\right)$ [m], which defines a circle in the x-z plane of center $(0.35, 0.15)$ [m] and radius 0.05 [m], and $\theta_{\mathrm{d}}(t) = \frac{\pi}{20}\sin\left(\frac{5\pi t}{15}\right)$ [rad], which is translated to the desired 2D quaternion trajectory $\zeta_{\mathrm{d}}(t) = \left[\cos\left(\frac{\theta_{\mathrm{d}}(t)}{2}\right), 0, 0, \sin\left(\frac{\theta_{\mathrm{d}}(t)}{2}\right)\right]^\top$. The simulation results are depicted in Figs. 4.3-4.5 for $t \in [0, 70]$ [sec]; Fig. 4.3 pictures the desired and actual trajectory of the object's center of mass (top), the pose errors $e_p(t), e_\zeta(t)$ (middle), as well as the velocity error $e_{v_f}(t)$ (bottom). We can verify from the figure that the desired trajectory is tracked almost perfectly, with negligible oscillations. The control inputs as well as the agent velocities with their respective limits are illustrated in Fig. 4.4. By appropriately tuning the control gains, which were set as $k_p = 15$, $k_\zeta = 30$, $K_{v_1} = K_{v_2} = \mathrm{diag}\{5, 2, 0.1\}$, we achieved confinement of the signals in the domain formed by the limits. Note also the difference due to the different load sharing coefficients. Finally, Fig. 4.3 depicts the norms of the adaptation signals $e_{\vartheta_i}(t)$ and $e_{\vartheta_{O_i}}(t)$, $\forall i \in \{1, 2\}$, which, as proven in the theoretical

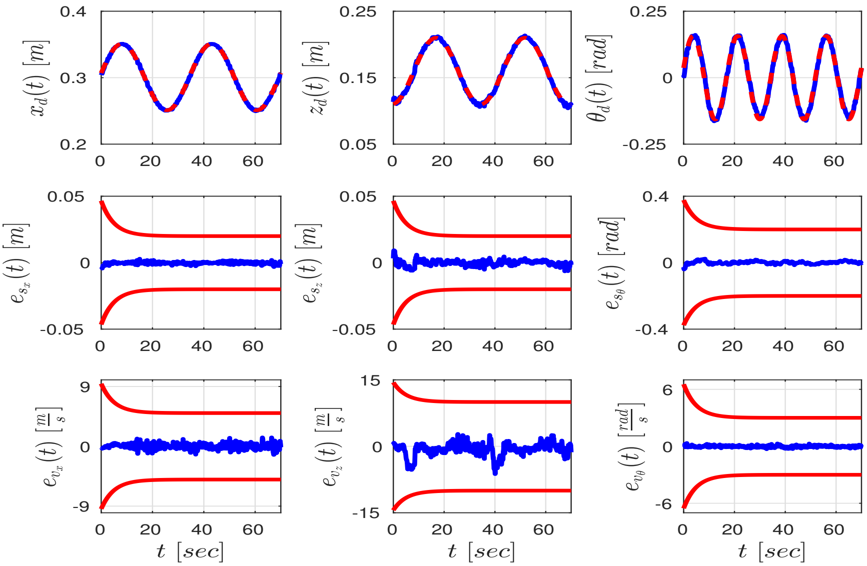


Figure 4.11: Experimental results for the controller of subsection 4.4.2, for $t \in [0, 45]$ [sec]. Top: The desired (with blue) and actual (with red) object trajectory in x-, z-axis, and around y-axis. Middle: The pose errors $e_s(t)$ (with blue), as well as the performance functions $\rho_s(t)$ (with red), respectively. Bottom: The velocity errors $e_v(t)$ (with blue), as well as the performance functions $\rho_v(t)$ (with red), respectively.

analysis, remain bounded. The functions $\delta_O(x_O, \dot{x}_O, t), \delta_i(q_i, \dot{q}_i, t)$ were taken as 0_6 and 0_{n_i}, respectively, and hence, the adaptation controller (4.29c), (4.29d) were not employed. Loosely speaking, the disturbances $d_O(x_O, \dot{x}_O, t), d_i(q_i, \dot{q}_i, t)$ were not taken into account in our model, without, however, degrading the performance of the proposed scheme.

Experimental Results

The desired trajectory for the experimental results of the controller developed in Section 4.4.1 was set to $x_\mathrm{d}(t) = 0.3 + 0.05 \sin\left(\frac{2\pi t}{35}\right)$ [m], $z_\mathrm{d}(t) = 0.15 - 0.05 \cos\left(\frac{2\pi t}{35}\right)$ [m], which defines a similar circle with the simulations section, and $\theta_\mathrm{d}(t) = \frac{\pi}{20} \sin\left(\frac{5\pi t}{35}\right)$ [rad], which is translated to the corresponding desired 2D quaternion trajectory $\zeta_\mathrm{d}(t) = \left[\cos\left(\frac{\theta_\mathrm{d}(t)}{2}\right), 0, 0, \sin\left(\frac{\theta_\mathrm{d}(t)}{2}\right)\right]^\top$. The control gains here were chosen as $k_p = 50$, $k_\zeta = 80$, $K_{v_1} = K_{v_2} = \mathrm{diag}\{3.5, 0.5, 0.5\}$. The disturbances $d_O(x_O, \dot{x}_O, t), d_i(q_i, \dot{q}_i, t)$ were also not taken into account in this case, by setting the functions $\delta_O(x_O, \dot{x}_O, t), \delta_i(q_i, \dot{q}_i, t)$ to 0_6 and 0_{n_i}, respectively. The simulation results are depicted in Figs. 4.6-4.8 for $t \in [0, 70]$ [sec]; Fig. 4.6 pictures the desired and actual trajectory of the object's center of mass (top), the pose errors $e_p(t), e_\zeta(t)$ (middle), as well as the velocity error $e_{v_f}(t)$ (bottom). We can verify from the

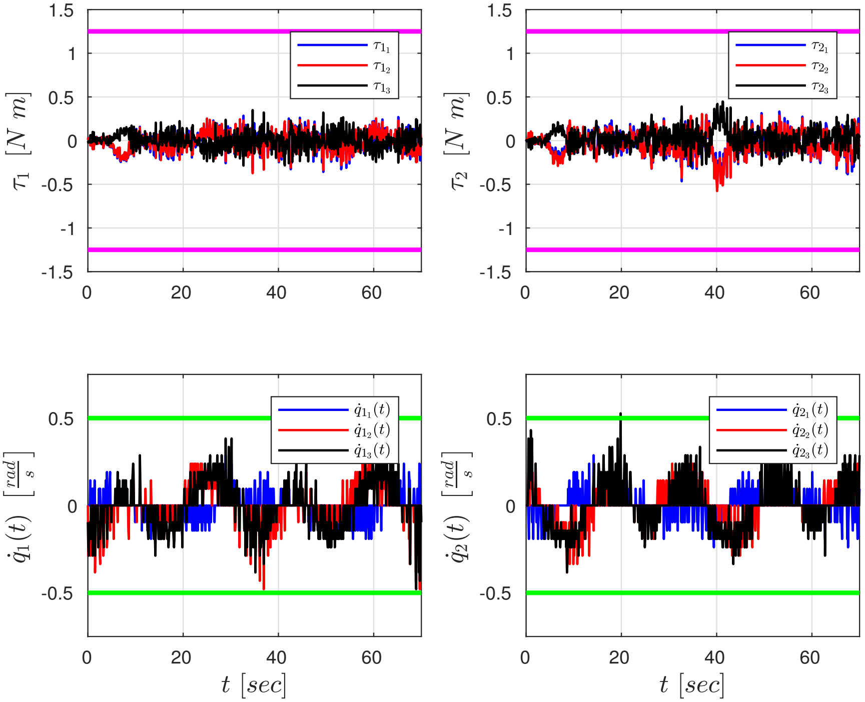


Figure 4.12: The agent joint torques and velocities of the experiment of the controller in subsection 4.4.2, for $t \in [0, 45]$ [sec], with their respective limits (purple and green lines, respectively). Top: The joint torques of agent 1 (left) and agent 2 (right). Bottom: The joint velocities of agent 1 (left) and agent 2 (right).

figure that the desired trajectory is tracked also for the experimental case, with oscillations in the velocity errors, which can be attributed to model uncertainties, sensor noise, or the un-modeled external disturbances, that have a larger effect in this (experimental) scenario. The control inputs as well as the agent velocities with their respective limits are illustrated in Fig. 4.7, which are confined in their respective limits. Finally, Fig. 4.6 depicts the norms of the adaptation signals $e_{\vartheta_i}(t)$ and $e_{\vartheta_{O_i}}(t)$, $\forall i \in \{1, 2\}$, which are bounded in this case as well.

4.5.2 Prescribed Performance Control

In this subsection, we present simulation and experimental results for the control protocol developed in Section 4.4.2.

Simulation Results

The desired trajectory in this subsection was chosen the same as in subsection 4.5.1, i.e., $x_{\mathrm{d}}(t) = 0.35 + 0.05 \sin\left(\frac{2\pi t}{15}\right)$ [m], $z_{\mathrm{d}}(t) = 0.15 - 0.05 \cos\left(\frac{2\pi t}{15}\right)$ [m], and

$\theta_\mathrm{d}(t) = \frac{\pi}{20}\sin\left(\frac{5\pi t}{15}\right)$ [rad]. The prescribed performance functions are chosen as: $\rho_{s_x}(t) = \rho_{s_z}(t) = 0.03\exp(-0.5t) + 0.01$ [m], $\rho_{s_\theta}(t) = 0.45\exp(-0.5t) + 0.05$ [rad], $\rho_{v_x}(t) = 3\exp(-0.5t) + 4$ [m/s], $\rho_{v_z}(t) = 5\exp(-0.5t) + 5$ [m/s], and $\rho_{v_\theta}(t) = 5\exp(-0.5t) + 5$ [rad/s]. The simulation results are depicted in Figs. 4.9 and 4.10; Fig. 4.9 shows the desired and actual trajectory of the object's center of mass (top), the pose errors $e_s(t)$ along with the performance functions $\rho_s(t)$ (middle), as well as the velocity error $e_v(t)$ along with the velocity performance functions $\rho_v(t)$ (bottom). It is verified that we achieve tracking of the desired trajectory with prescribed performance. The control inputs (joint torques) as well as the joint velocities are given in Fig. 4.10. By following the procedure described in the proof of theorem 4.2, we tune the gains to the values $g_s = 0.05$ and $g_v = 7$ so that the joint torques and velocities respect their respective bounds.

Experimental Results

We set the desired trajectory in this subsection as in subsection 4.5.1, i.e., $x_\mathrm{d}(t) = 0.3 + 0.05\sin\left(\frac{2\pi t}{35}\right)$ [m], $z_\mathrm{d}(t) = 0.15 - 0.05\cos\left(\frac{2\pi t}{35}\right)$ [m], and $\theta_\mathrm{d}(t) = \frac{\pi}{20}\sin\left(\frac{5\pi t}{35}\right)$ [rad]. The parameters for the prescribed performance functions are chosen as: $\rho_{s_x}(t) = \rho_{s_z}(t) = 0.03\exp(-0.2t) + 0.02$ [m], $\rho_{s_\theta}(t) = 0.2\exp(-0.2t) + 0.2$ [rad], $\rho_{v_x}(t) = 5\exp(-0.2t) + 5$ [m/s], $\rho_{v_z}(t) = 5\exp(-0.2t) + 10$ [m/s], and $\rho_{v_\theta}(t) = 4\exp(-0.2t) + 3$ [m/s]. The values for the gains are set at $g_s = 0.05$ and $g_v = 6.8$. The simulation results are depicted in Figs. 4.11 and 4.12; Fig. 4.11 shows the desired and actual trajectory of the object's center of mass (top), the pose errors $e_s(t)$ along with the performance functions $\rho_s(t)$ (middle), as well as the velocity error $e_v(t)$ along with the velocity performance functions $\rho_v(t)$ (bottom). The prescribed performance tracking can be verified in the experimental case as well. The control inputs (joint torques) as well as the joint velocities are given in Fig. 4.12, where it is shown that they respect their corresponding limits.

4.5.3 Discussion

It is clear from the aforementioned figures that the tracking of the desired trajectory is achieved by both controllers in the computer simulation and experimental cases. It is worth noting first the difference between the simulations and experiments, which, for both control protocols, lies in the velocity errors. The latter present some oscillatory behavior in the experimental case, which can be attributed to sensor noise, external disturbances/model uncertainties, or inaccuracies/delays of the internal ArbotiX-M controller. The same reason led us to choose slower desired trajectories for the real-time experiments. Nonetheless, these oscillations do not affect the overall tracking performance. Recall that for the Prescribed Performance Controller, tracking of the desired trajectory does not require the functions $\rho_{v_k}(t)$ - and hence the velocity errors e_{v_k}, $k \in \mathcal{K}$ - to asymptotically approach zero.

Among the two control schemes, we can notice a slightly more aggressive behavior of the joint velocities and torques for the Prescribed Performance Controller, which can be attributed to the virtual force that "pushes" the errors e_{s_k}, e_{v_k} not to hit the bounds of the performance functions $\rho_s(t), \rho_v(t)$, respectively, $\forall k \in \mathcal{K}$. Note, however, that this methodology does not require knowledge of the structure of the dynamic terms $M_i(q_i)$, $C_i(q_i, \dot{q}_i)$, $g_i(q_i)$, whose derivation can be tedious, and yields thus significantly lower analytic complexity, without sacrificing the actual performance.

4.6 Conclusion and Future Work

We presented two novel decentralized control protocols for the cooperative manipulation of a single object by N robotics agents. Firstly, we developed a quaternion-based approach that avoids representation singularities with adaptation laws to compensate for dynamic uncertainties. Secondly, we developed a robust control law that guarantees prescribed performance for the transient and steady state of the object. Both methodologies were validated via realistic simulations and experimental results. Future efforts will be devoted towards applying the proposed techniques to cases with non rigid grasping points as well as uncertain object geometric characteristics.

Chapter 5

Model-Predictive Cooperative Transportation

This chapter addresses the problem of cooperative transportation of an object rigidly grasped by N robotic agents. In particular, we propose two Nonlinear Model Predictive Control (NMPC) schemes that guarantee the navigation of the object to a desired pose in a bounded workspace with obstacles, while complying with certain input saturations of the agents. The first control scheme is centralized, in the sense that a central computer calculates the control inputs of the robotic agents, whereas the second control scheme is based on inter-agent communication and is decentralized, since each agent calculates its own control signal. Moreover, the proposed methodologies ensure that the agents do not collide with each other or with the workspace obstacles as well as that they do not pass through singular configurations. The feasibility and convergence analysis of the NMPC are explicitly provided. Finally, simulation results illustrate the validity and efficiency of the proposed methods.

5.1 Introduction

Constrained-based control has always been of special interest to the automatic control/robotics community, due to the advantages it yields, by keeping variables of interest in specific compact sets, while achieving a primary task. A widely employed methodology in the last years is the methodology of Model Predictive Control (MPC) [113], where a constrained optimization problem is solved for a finite horizon in the future, providing a prediction of the state evolution. Motivated by the power limitations of robot actuators as well as collision- and singularity avoidance properties in cooperative manipulation tasks, we aim to design a MPC scheme for the cooperative transportation of an object by N robotic agents, while complying with certain constraints.

Regarding manipulation tasks, such as pose/force or trajectory tracking (see the references of the previous chapter), collision with obstacles of the environment has been dealt with only by exploiting the extra degrees of freedom that appear in over-actuated robotic agents. Potential field-based algorithms may suffer from local minima and navigation functions [114] cannot be extended to multi-agent second order dynamical systems in a trivial way. Moreover, these methods usually result in high control input values near obstacles that need to be avoided, which might conflict the saturation of the actual motor inputs.

Another important property that concerns robotic manipulators is the singularities of the Jacobian matrix, which maps the joint velocities of the agent to a 6D vector of generalized velocities. Such *singular kinematic* configurations, which indicate directions towards which the agent cannot move, must be always avoided, especially when dealing with task-space control in the end-effector [26]. In the same vein, *representation* singularities can also occur in the mapping from coordinate rates to angular velocities of a rigid body.

In the previous chapter we considered the problem of trajectory tracking for the center of mass of the object, without taking into account potential workspace obstacles or kinematic singularities. Such constraints are incorporated in this chapter, where we address the problem of cooperative transportation of an object in a bounded workspace with obstacles. In particular, given N agents that rigidly grasp an object, we design control inputs for the navigation of the object to a final pose, while avoiding inter-agent collisions as well as collisions with obstacles. Moreover, we take into account constraints that emanate from control input saturation as well kinematic and representation singularities. We propose both a centralized and a decentralized methodology.

For the design of a stabilizing feedback control law under such constraints, one would ideally look for a closed-loop solution for the feedback law satisfying the constraints while optimizing the performance. However, typically the optimal feedback law cannot be found analytically, even in the unconstrained case, since it involves the solution of the corresponding Hamilton-Jacobi-Bellman partial differential equations. One approach to circumvent this problem is the repeated solution of an open-loop finite-horizon optimal control problem for a given state. The first part of the resulting open-loop input signal is implemented and the whole process is repeated. Control approaches using this strategy are referred to as Nonlinear Model Predictive Control (NMPC) (see e.g. [113, 115–123]), which we aim to use in this work for the problem of the constraint cooperative object manipulation. We design a centralized control protocol, where a central computer calculates the control signal of all the agents as well as decentralized control laws, based on inter-agent communication.

The remainder of the chapter consists of two main parts. Section 5.2 presents the centralized methodology, with 5.2.1 and 5.2.2 presenting the problem formulation and its solution, respectively, and 5.2.3 providing simulation results. Similarly, Section 5.3 presents the decentralized methodology, where Sections 5.3.1, 5.3.2, and 5.3.3 give the respective problem formulation, its solution, and the simulation results, respectively. Finally, Section 5.4 concludes the paper.

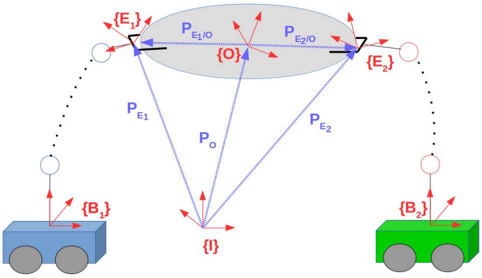


Figure 5.1: Two robotic arms rigidly grasping an object with the corresponding frames.

5.2 Centralized Cooperative Transportation

5.2.1 Problem Formulation

Consider a bounded and convex workspace $\mathcal{W} \subseteq \mathbb{R}^3$ consisting of N robotic agents rigidly grasping an object, (see Fig. 5.1), and Z obstacles described by the ellipsoids $\mathcal{O}_z := \left\{p \in \mathbb{R}^3 : (p-c_z)^\top P_{\mathcal{O}_z}(p-c_z) \le 1\right\}, z \in \mathcal{Z} := \{1, \dots, Z\}$, where $c_z \in \mathbb{R}^3$ is the center of the ellipsoid, and $P_{\mathcal{O}_z}$ is a matrix whose eigenvalues are the lengths of the ellipsoid's three semi-axes. The agents are considered to be fully actuated and they consist of a base that is able to move around the workspace (e.g., mobile or aerial vehicle) and a robotic arm. The reference frames corresponding to the i-th end-effector and the object's center of mass are denoted with $\{E_i\}$ and $\{O\}$, respectively, whereas $\{I\}$ corresponds to an inertial reference frame. The rigidity of the grasps implies that the agents can exert any forces/torques along every direction to the object. We consider that each agent i knows the position and velocity only of its own state as well as its own and the object's geometric parameters. Moreover, no interaction force/torque measurements or on-line communication is required.

System model

In this section we provide the modeling of the robotic agents, the object, and the coupled dynamics, which follows closely the modeling of chapter 4, elaborating in more detail on the structure of the agents.

Robotic Agents

We denote by $q_i \in \mathbb{R}^{n_i}$ the joint space variables of agent $i \in \mathcal{N}$, with $n_i := n_{\alpha_i} + 6$, $q_i := [p_{B_i}^\top, \eta_{B_i}^\top, \alpha_i^\top]^\top$, where $p_{B_i} := [x_{B_i}, y_{B_i}, z_{B_i}]^\top \in \mathbb{R}^3, \eta_{B_i} := [\phi_{B_i}, \theta_{B_i}, \psi_{B_i}]^\top \in \mathbb{T}$ is the position and Euler-angle orientation of the agent's base, and $\alpha_i :\in \mathbb{R}^{n_{\alpha_i}}, n_{\alpha_i} > 0$, are the degrees of freedom of the robotic arm. The overall joint space configuration vector is denoted as $q := [q_1^\top, \dots, q_N^\top]^\top \in \mathbb{R}^n$, with $n := \sum_{i \in \mathcal{N}} n_i$. In addition, we denote as $p_{E_i} : \mathbb{R}^{n_i} \to \mathbb{R}^3, \eta_{E_i} : \mathbb{R}^{n_i} \to \mathbb{T}$ the position and Euler-angle orientation of agent i's end-effector. Let also $v_i = [\dot{p}_{E_i}^\top, \omega_{E_i}^\top]^\top \in \mathbb{R}^6$ denote the velocity of agent i's end-effector, where $\dot{p}_{B_i}, \omega_{B_i}$ are the linear and angular velocity, respectively, of the agent's base.

We consider that each agent $i \in \mathcal{N}$ has access to its own state q_i as well as $\dot{p}_{B_i}^{B_i}, \omega_{B_i}^{B_i}$, and $\dot{\alpha}_i$ via on-board sensors. Then, $\dot{p}_{B_i}, \omega_{B_i}$ can be obtained via $\dot{p}_{B_i} = R_{B_i}(\eta_{B_i})\dot{p}_{B_i}^{B_i}$, $\omega_{B_i} = R_{B_i}(\eta_{B_i})\omega_{B_i}^{B_i}$, where $R_{B_i} : \mathbb{T} \to SO(3)$ is the rotation matrix of the agent i's base. Moreover, $\dot{\eta}_{B_i}$ is related to ω_{B_i} via $\omega_{B_i} = J_{B_i}(\eta_{B_i})\dot{\eta}_{B_i}$, where $J_{B_i} : \mathbb{T} \to \mathbb{R}^{3\times 3}$, with

$$J_{B_i}(\eta_{B_i}) := \begin{bmatrix} 1 & 0 & \sin(\theta_{B_i}) \\ 0 & \cos(\phi_{B_i}) & -\cos(\theta_{B_i})\sin(\phi_{B_i}) \\ 0 & \sin(\phi_{B_i}) & \cos(\theta_{B_i})\cos(\phi_{B_i}) \end{bmatrix}.$$

The pose of the ith end-effector can be computed via

$$\begin{aligned} p_{E_i}(q_i) &= p_{B_i} + R_{B_i}(\eta_{B_i})k_{p_i}(\alpha_i), \\ \eta_{E_i}(q_i) &= k_{\eta_i}(\eta_{B_i}, \alpha_i), \end{aligned}$$

where $k_{p_i} : \mathbb{R}^{n_{\alpha_i}} \to \mathbb{R}^3, k_{\eta_i} : \mathbb{T} \times \mathbb{R}^{n_{\alpha_i}} \to \mathbb{T}$ are the forward kinematics of the robotic arm [26]. Then, v_i can be computed as

$$v_i = \begin{bmatrix} \dot{p}_{E_i}(q_i) \\ \omega_{E_i}(q_i, \dot{q}_i) \end{bmatrix} = \begin{bmatrix} \dot{p}_{B_i} - S(R_{B_i}k_{p_i})\omega_{B_i} + R_{B_i}\frac{\partial k_{p_i}}{\partial \alpha_i} \\ \omega_{B_i} + R_{B_i}J_{A_i}\dot{\alpha}_i \end{bmatrix}, \tag{5.1}$$

where $J_{A_i} : \mathbb{R}^{n_{\alpha_i}} \to \mathbb{R}^{3\times n_{\alpha_i}}$ is the angular Jacobian of the robotic arm with respect to the agent's base. The differential kinematics (5.1) can be written as

$$v_i = \begin{bmatrix} \dot{p}_{E_i}(q_i) \\ \omega_{E_i}(q_i, \dot{q}_i) \end{bmatrix} = J_i(q_i)\dot{q}_i, \tag{5.2}$$

where $J_i : \mathbb{R}^{n_i} \to \mathbb{R}^{6\times n_i}$ is the agent Jacobian matrix, with

$$J_i(q_i) := \begin{bmatrix} I_3 & -S(R_{B_i}(\eta_{B_i})k_{p_i}(\alpha_i))J_{B_i}(\eta_{B_i}) & R_{B_i}(\eta_{B_i})\frac{\partial k_{p_i}(\alpha_i)}{\partial \alpha_i} \\ 0_{3\times 3} & J_{B_i}(\eta_{B_i}) & R_{B_i}(\eta_{B_i})J_{A_i}(q_i) \end{bmatrix}.$$

Remark 5.1. Note that J_{B_i} becomes singular at representation singularities, when $\theta_{B_i} = \pm\frac{\pi}{2}$ and J_i becomes singular at kinematic singularities defined by the set

$$\mathcal{Q}_i := \{q_i \in \mathbb{R}^{n_i} : \det(J_i^\top J_i) = 0\}, i \in \mathcal{N}.$$

In the following, we will aim at guaranteeing that q_i will always be in the closed set:

$$\widetilde{\mathcal{Q}}_i := \{q_i \in \mathbb{R}^{n_i} : |\det(J_i^\top J_i)| \geq \varepsilon > 0\}, i \in \mathcal{N},$$

for a small positive constant ε.

The joint-space dynamics for agent $i \in \mathcal{N}$ can be computed using the Lagrangian formulation, as in (4.3) (see [124] for the explicit derivation when a moving base with an attached robotic arm is concerned):

$$B_i(q_i)\ddot{q}_i + C_{q_i}(q_i, \dot{q}_i)\dot{q}_i + g_{q_i}(q_i) = \tau_i - J_i^\top \lambda_i, \tag{5.3}$$

where we use $\lambda_i \in \mathbb{R}^6$ for the generalized force vector that agent i exerts on the object; $\tau_i \in \mathbb{R}^{n_i}$ is the vector of generalized joint-space inputs, with $\tau_i := [\lambda_{B_i}^\top, \tau_{\alpha_i}^\top]^\top$, where $\lambda_{B_i} := [f_{B_i}^\top, \mu_{B_i}^\top]^\top \in \mathbb{R}^6$ is the generalized force vector on the center of mass of the agent's base and $\tau_{\alpha_i} \in \mathbb{R}^{n_{\alpha_i}}$ is the torque inputs of the robotic arms' joints. Similarly to (4.5), we obtain the task-space dynamics:

$$M_i(q_i)\dot{v}_i + C_i(q_i, \dot{q}_i)v_i + g_i(q_i) = u_i - \lambda_i. \tag{5.4}$$

We define by $\mathcal{A}_i(q_i)$, $i \in \mathcal{N}$, the ellipsoid that bounds the i th agent's volume, i.e., the workspace of the arm of agent i [26] enlarged so that it includes the ith base. Note that $\mathcal{A}_i$ depends on q_i and can be explicitly found.

Object Dynamics

Similarly to chapter 4 for the object, we denote by $x_O := [p_O^\top, \eta_O^\top]^\top$, $v_O := [\dot{p}_O^\top, \omega_O^\top]^\top$ the pose and velocity of the object, with the second order dynamics:

$$\dot{x}_O = J_O(\eta_O)v_O, \tag{5.5a}$$

$$\lambda_O = M_O(x_O)\dot{v}_O(t) + C_O(x_O, \dot{x}_O)v_O + g_O(x_O), \tag{5.5b}$$

with the corresponding terms as in (4.7); $\lambda_O \in \mathbb{R}^6$ is the force vector acting on the object's center of mass. Also, similarly to the robotic agents, we define by $\mathcal{C}_O(x_O)$ as the bounding ellipsoid of the object.

Coupled Dynamics By following Section 4.3, we obtain the coupled object-agents dynamics:

$$\widetilde{M}(q)\dot{v}_O + \widetilde{C}(q, \dot{q})v_O + \widetilde{g}(q) = G^T(q)u, \tag{5.6}$$

with the respective terms as in (4.15)

Remark 5.2. As mentioned in Chapter 4, since the geometric object parameters $p_{E_i/O}^{E_i}$ and $\eta_{E_i/O}$ are known, each agent can compute p_O, η_O and v_O from the coupled kinematics and dynamics, respectively, without employing any sensory data. In the same vein, all agents can also compute the object's bounding ellipsoid $\mathcal{C}_O$, which depends on q.

Remark 5.3. Note that the agents dynamics under consideration hold for generic robotic agents comprising of a moving base and a robotic arm. Hence, the considered framework can be applied for mobile, aerial, or underwater manipulators.

We can now formulate the problem considered in this work:

Problem 5.1. Consider N robotic agents rigidly grasping an object, governed by the coupled dynamics (5.6). Given the desired pose $x_{O,\text{des}}$, design the control input $u \in \mathbb{R}^{6N}$ such that $\lim\limits_{t\to\infty} x_O(t) = x_{O,\text{des}}$, while ensuring the satisfaction of the following collision avoidance and singularity properties:

1. $\mathcal{A}_i(q_i(t)) \cap \mathcal{O}_z = \emptyset, \forall i \in \mathcal{N}, z \in \mathcal{Z}$,

2. $\mathcal{C}_O(x_O(t)) \cap \mathcal{O}_z = \emptyset, \forall z \in \mathcal{Z}$,

3. $\mathcal{A}_i(q_i(t)) \cap \mathcal{A}_j(q_j(t)) = \emptyset, \forall i, j \in \mathcal{N}, i \neq j$,

4. $-\frac{\pi}{2} < -\bar{\theta} \leq \theta_O(t) \leq -\bar{\theta} < \frac{\pi}{2}$,

5. $-\frac{\pi}{2} < -\bar{\theta} \leq \theta_{B_i}(t) \leq -\bar{\theta} < \frac{\pi}{2}$,

6. $q_i(t) \in \widetilde{\mathcal{Q}}_i$,

$\forall t \in \mathbb{R}_{\geq 0}$, for a $0 < \bar{\theta} < \frac{\pi}{2}$, as well as the input and velocity magnitude constraints: $|\tau_{i_k}| \leq \bar{\tau}_i, |\dot{q}_{i_k}| \leq \bar{\dot{q}}_i, \forall k \in \{1, \ldots, n_i\}, i \in \mathcal{N}$, for some positive constants $\bar{\tau}_i, \bar{\dot{q}}_i, i \in \mathcal{N}$.

The aforementioned constraints correspond to the following specifications:

- 1) stands for collision avoidance between the agents and the obstacles.
- 2) stands for collision avoidance between the object and the obstacles.
- 3) stands for collision avoidance between the agents.
- 4) stands for representation singularity avoidance of the object.
- 5) stands for representation singularity avoidance of the agents' bases.
- 6) stands for kinematic singularity avoidance of the agents.

In order to solve the aforementioned problem, we need the following reasonable assumption regarding the workspace, which implies that the collision-free space is connected:

Assumption 5.1. (Problem feasibility) The set $\{q \in \mathbb{R}^n : \mathcal{A}_i(q_i) \cap \mathcal{O}_z = \emptyset, \mathcal{A}_i(q_i) \cap \mathcal{A}_j(q_j) = \emptyset, \mathcal{C}_i(x_{O_i}(q_i)) \cap \mathcal{O}_z = \emptyset, \forall i, j \in \mathcal{N}, i \neq j, z \in \mathcal{Z}\}$, is connected.

In the aforementioned assumption, $x_{O_i} \coloneqq [p_{O_i}^\top, \eta_{O_i}^\top]^\top$ denotes the pose of the object as a function of q_i, derived by inverting (4.9):

$$\begin{align} p_O = p_{O_i}(q_i) &\coloneqq p_{E_i}(q_i) + p_{O/E_i}(q_i) \coloneqq p_{E_i}(q_i) + R_{E_i}(q_i) p_{O/E_i}^{E_i}, \tag{5.7a} \\ \eta_O = \eta_{O_i}(q_i) &= \eta_{E_i}(q_i) + \eta_{O/E_i}, \tag{5.7b} \end{align}$$

We also define the following sets for every $i \in \mathcal{N}$:

$$\begin{align*} S_{i,O} &\coloneqq \{q_i \in \mathbb{R}^{n_i} : \mathcal{A}_i(q_i) \cap \mathcal{O}_z \neq \emptyset, \forall z \in \mathcal{Z}\}, \\ S_{i,A} &\coloneqq \{q \in \mathbb{R}^n : \mathcal{A}_i(q_i) \cap \mathcal{A}_j(q_j) \neq \emptyset, \forall j \in \mathcal{N} \backslash \{i\}\}, \\ S_O &\coloneqq \{x_O \in \mathbb{M} : \mathcal{C}_O(x_O) \cap \mathcal{O}_z \neq \emptyset\}. \end{align*}$$

associated with the desired collision-avoidance properties. Note that the aforementioned sets can be explicitly calculated, since the ellipsoids $\mathcal{A}_i$, $\mathcal{C}_O$, $\mathcal{O}_z$ are known.

5.2.2 Problem Solution

In this section, a systematic solution to Problem 1 is introduced. Our overall approach builds on designing a Nonlinear Model Predictive control scheme for the system of the manipulators and the object. Nonlinear Model Predictive Control (see e.g. [113, 115–122]) have been proven suitable for dealing with nonlinearities and state and input constraints.

The coupled agents-object *nonlinear dynamics* can be written in compact form as follows:

$$\dot{x} = f(x,u) \coloneqq \begin{bmatrix} f_1(x,u) \\ f_2(x,u) \\ f_3(x,u) \end{bmatrix}, x(0) \coloneqq x_0, \tag{5.8}$$

where $x \coloneqq [x_O^\top, v_O^\top, q^\top]^\top \in \mathbb{R}^{n+12}, u \in \mathbb{R}^{6N}$ and

$$\begin{aligned} f_1(x,u) &\coloneqq J_O(\eta_O)v_O, \\ f_2(x,u) &\coloneqq \widetilde{M}^{-1}(q)\left[G^\top(q)u - \widetilde{C}(q,\dot{q})v_O - \widetilde{g}(q)\right], \\ f_3(x,u) &\coloneqq \hat{J}(q)\bar{J}_O(q)\widetilde{I}v_O, \end{aligned}$$

where we have also used that:

$$\begin{aligned} \hat{J}(q) &\coloneqq \mathrm{diag}\left\{J_i^\top \left[(J_iJ_i^\top)^{-1}\right]_{i\in\mathcal{N}}\right\} \in \mathbb{R}^{n\times 6N}, \\ \bar{J}_O(q) &\coloneqq \mathrm{diag}\left\{\left[J_{O_i}\right]_{i\in\mathcal{N}}\right\} \in \mathbb{R}^{6N\times 6N}, \\ \widetilde{I} &\coloneqq \left[I_6, \cdots, I_6\right]^\top \in \mathbb{R}^{6N\times 6} \end{aligned} \tag{5.9}$$

Note that f is *locally Lipschitz continuous* in its domain since it is continuously differentiable in its domain. Next, we define the respective errors:

$$e(t) \coloneqq x(t) - x_{\mathrm{des}} = \begin{bmatrix} x_O(t) \\ v_O(t) \\ q(t) \end{bmatrix} - \begin{bmatrix} x_{O,\mathrm{des}} \\ \dot{x}_{O,\mathrm{des}} \\ q_{\mathrm{des}} \end{bmatrix} = \begin{bmatrix} x_O(t) - x_{O,\mathrm{des}} \\ v_O(t) \\ q(t) - q_{\mathrm{des}} \end{bmatrix} \in \mathbb{R}^{n+12}, \tag{5.10}$$

where $q_{\mathrm{des}} \coloneqq [q_{1,\mathrm{des}}, \ldots, q_{N,\mathrm{des}}]^\top$ is appropriately chosen such that $x_{O_i}(q_{i,\mathrm{des}}) = x_{O,\mathrm{des}}$, and $\dot{x}_{O,\mathrm{des}} = \dot{q}_{\mathrm{des}} = 0$. The error dynamics are then $\dot{e}(t) = f(x(t), u(t))$, which can be appropriately transformed to be written as:

$$\dot{e}(t) = f_e(e(t), u(t)), \quad e(0) = e_0 = x(0) - x_{\mathrm{des}}. \tag{5.11}$$

where $f_e(t) \coloneqq f(e(t) + x_{\mathrm{des}}, u(t))$. By ignoring over-actuated input terms, we have that $\tau_i = J_i^\top(q_i)u_i$, which becomes

$$\|\tau_i\| \leq \bar{\tau}_i \Leftrightarrow \sigma_{\mathrm{min},i}\|u_i\| \leq \bar{\tau}_i, \tag{5.12}$$

where we have employed the property $\sigma_{\min}(J_i^\top)\|u_i\| \leq \|J_i^\top u_i\|$, with $\sigma_{\min}(J_i^\top)$ denoting the minimum singular value of $J_i^\top$, which is strictly positive, if the constraint $q_i \in \widetilde{\mathcal{Q}}_i$ is always satisfied. Hence, the constraint $|\tau_{i_k}| \leq \bar{\tau}_i$ is equivalent to

$$\|u_i\| \leq \frac{\bar{\tau}_i}{\sigma_{\min}(J_i^\top)}, \forall i \in \mathcal{N}. \tag{5.13}$$

Let us now define the following set $U \subseteq \mathbb{R}^{6N}$:

$$U \coloneqq \{u \in \mathbb{R}^{6N} : \|u_i\| \leq \frac{\bar{\tau}_i}{\sigma_{\min}(J_i^\top)}, \forall i \in \mathcal{N}\}, \tag{5.14}$$

as the set that captures the control input constraints of the error dynamics system (5.11). Define also the set $X \subseteq \mathbb{R}^{n+12}$:

$$X \coloneqq \Big\{x \in \mathbb{R}^{n+12} : \theta_O \in [\bar{\theta}, \bar{\theta}], \theta_{B_i} \in [\bar{\theta}, \bar{\theta}],$$
$$|\dot{q}_{k_i}| \leq \bar{\dot{q}}_i, q_i \in \widetilde{\mathcal{Q}}_i \backslash \left(\mathcal{S}_{i,O}(q_i) \cup \mathcal{S}_{i,A}(q)\right), \forall i \in \mathcal{N}, x_O \in \mathbb{R}^3 \backslash S_O(x_O)\Big\}.$$

The set X captures all the state constraint of the system dynamics (5.8). In view of (5.10), we define the set $E \subseteq \mathbb{R}^{n+12}$ as:

$$E \coloneqq \{e \in \mathbb{R}^{n+12} : e \in X \oplus (-x_{\text{des}})\},$$

as the set that captures all the constraints of the error dynamics system (5.11).

The problem in hand is the design of a control input $u(t) \in U$ such that $\lim_{t\to\infty} \|e(t)\| = 0$ while ensuring $e(t) \in E, \forall t \in \mathbb{R}_{\geq 0}$. In order to solve the aforementioned problem, we propose a Nonlinear Model Predictive scheme, that is presented hereafter.

Consider a sequence of sampling times $\{t_i\}_{i\geq 0}$ with a constant sampling period $0 < h < T_p$, where is T_p is the prediction horizon, such that:

$$t_{i+1} = t_i + h, \forall\ i \geq 0. \tag{5.15}$$

In the sampling-data NMPC, a finite-horizon open-loop optimal control problem (OCP) is solved at discrete sampling time instants t_i based on the current state error information $e(t_i)$. The solution is an optimal control signal $\hat{u}(t)$, for $t \in [t_i, t_i + T_p]$. For more details, the reader is referred to [115]. The open-loop input signal applied in between the sampling instants is given by the solution of the following Optimal Control Problem (OCP):

$$\min_{\hat{u}(\cdot)} J(e(t_i), \hat{u}(\cdot)) = \min_{\hat{u}(\cdot)} \left\{ V(\hat{e}(t_i + T_p)) + \int_{t_i}^{t_i+T_p} F(\hat{e}(s), \hat{u}(s)) ds \right\} \tag{5.16a}$$

subject to:

$$\dot{\hat{e}}(s) = f_e(\hat{e}(s), \hat{u}(s)), \hat{e}(t_i) = e(t_i), \tag{5.16b}$$

$$\hat{e}(s) \in E, \hat{u}(s) \in U, s \in [t_i, t_i + T_p], \tag{5.16c}$$

$$\hat{e}(t_i + T_p) \in \mathcal{E}_f, \tag{5.16d}$$

where the hat $\hat{\cdot}$ denotes the predicted variables (internal to the controller), i.e. $\hat{e}(\cdot)$ is the solution of (5.16b) driven by the control input $\hat{u}(\cdot) : [t_i, t_i + T_p] \to \mathcal{U}$ with initial condition $e(t_i)$. Note that the predicted values are not necessarily the same with the actual closed-loop values (see [115]). The term $F : E \times U \to \mathbb{R}_{\geq 0}$, is the *running cost*, and is chosen as:

$$F(e(t), u(t)) \coloneqq e^\top Q e + u^\top R u. \tag{5.17}$$

The terms $V : E \to \mathbb{R}_{>0}$ and $\mathcal{E}_f$ are the *terminal penalty cost* and *bounded terminal set*, respectively, and are used to enforce the stability of the system (see Section 4.2). The terminal cost is given by $V(e) \coloneqq e^\top P e$. The terms $Q \in \mathbb{R}_{\geq 0}^{(n+12)\times(n+12)}$, $P \in \mathbb{R}_{>0}^{(n+12)\times(n+12)}$ and $R \in \mathbb{R}_{>0}^{6N\times 6N}$ are chosen as:

$$\begin{aligned}
Q &\coloneqq \text{diag}\{\widetilde{q}_1, \dots, \widetilde{q}_{n+12}\},\\
P &\coloneqq \text{diag}\{\widetilde{p}_1, \dots, \widetilde{p}_{n+12}\},\\
R &\coloneqq \text{diag}\{\widetilde{r}_1, \dots, \widetilde{r}_{6N}\}.
\end{aligned}$$

where $\widetilde{q}_i \in \mathbb{R}_{\geq 0}, \widetilde{p}_i \in \mathbb{R}_{>0}, \forall i \in \{1, \dots, n+12\}$ and $\widetilde{r}_j \in \mathbb{R}_{>0}, \forall j \in \{1, \dots, 6N\}$ are constant weights. For the running cost, it holds that $F(0, 0) = 0$, as well as:

$$m \| [e^\top u^\top]^\top \|^2 \leq m \left\| \begin{bmatrix} e \\ u \end{bmatrix} \right\|^2 \leq F(e, u) \leq M \left\| \begin{bmatrix} e \\ u \end{bmatrix} \right\|^2 \leq M \| [e^\top u^\top]^\top \|^2, \tag{5.18}$$

where

$$\begin{aligned}
m &\coloneqq \min\{\widetilde{q}_1, \dots, \widetilde{q}_{n+12}, \widetilde{r}_1, \dots, \widetilde{r}_{6N}\},\\
M &\coloneqq \max\{\widetilde{q}_1, \dots, \widetilde{q}_{n+12}, \widetilde{r}_1, \dots, \widetilde{r}_{6N}\}.
\end{aligned}$$

Note that $m\|[e^\top u^\top]^\top\|^2$, $M\|[e^\top u^\top]^\top\|^2$ are $\mathcal{K}_\infty$ functions, according to Definition 2.3.

The solution of the OCP (5.16a)-(5.16d) at time t_i provides an optimal control input denoted by $\hat{u}^\star(t; e(t_i))$, for $t \in [t_i, t_i + T_p]$. It defines the open-loop input that is applied to the system until the next sampling instant t_{i+1}:

$$u(t; e(t_i)) = \hat{u}^\star(t_i; e(t_i)), t \in [t_i, t_{i+1}). \tag{5.19}$$

The corresponding *optimal value function* is given by:

$$J^\star(e(t_i)) \triangleq J^\star(e(t_i), \hat{u}^\star(\cdot; e(t_i))). \tag{5.20}$$

where $J(\cdot)$ as is given in (5.16a). The control input $u(t; e(t_i))$ is a feedback, since it is recalculated at each sampling instant using the new state information. The solution of (5.11) starting at time t_1 from an initial condition $e(t_1)$, applying a control input $u : [t_1, t_2] \to \mathcal{U}$ is denoted by $e(s; u(\cdot), e(t_1)), s \in [t_1, t_2]$. The predicted state of the system (5.11) at time $t_i + s, s > 0$ is denoted by $\hat{e}(t_i + s; u(\cdot), e(t_i))$ and it is based on the measurement of the state $e(t_i)$ at time t_i, when a control input $u(\cdot; e(t_i))$ is applied to the system (5.11) for the time period $[t_i, t_i + s]$. Thus, it holds that:

$$e(t_i) = \hat{e}(t_i; u(\cdot), e(t_i)). \tag{5.21}$$

We define an admissible control input as:

Definition 5.1. A control input $u : [0, T_p] \to \mathbb{R}^{6N}$ for a state e_0 is called *admissible*, if all the following hold:

1. $u(\cdot)$ is piecewise continuous;
2. $u(s) \in U, \forall\ s \in [0, T_p]$;
3. $e(s; u(\cdot), e_0) \in E, \forall\ s \in [0, T_p]$;
4. $e(T_p; u(\cdot), e_0) \in \mathcal{E}_f$;

Lemma 5.1. *The terminal penalty function $V(\cdot)$ is Lipschitz continues in $\mathcal{E}_f$, with Lipschitz constant $L_V = 2\varepsilon_0\sigma_{\max}(P)$, for all $e(t) \in \mathcal{E}_f$, where $\varepsilon_0 := \sup\{\|e\| : e \in \mathcal{E}_f\}$.*

Proof. For every $e_1, e_2 \in \mathcal{E}_f$, the following holds:

$$\begin{aligned}|V(e_1) - V(e_2)| &= |e_1^\top Pe_1 - e_2^\top Pe_2| = |e_1^\top Pe_1 + e_1^\top Pe_2 - e_1^\top Pe_2 - e_2^\top Pe_2| \\ &= |e_1^\top P(e_1 - e_2) - e_2^\top P(e_1 - e_2)| \leq |e_1^\top P(e_1 - e_2)| + |e_2^\top P(e_1 - e_2)|.\end{aligned} \tag{5.22}$$

By employing the property that:

$$|x^\top Ay| \leq \sigma_{\max}(A)\|x\|\|y\|, \forall\ x, y \in \mathbb{R}^n, A \in \mathbb{R}^{n\times n},$$

(5.22) is written as:

$$\begin{aligned}|V(e_1) - V(e_2)| &\leq \sigma_{\max}(P)\|e_1\|\|e_1 - e_2\| + \sigma_{\max}(P)\|e_2\|\|e_1 - e_2\| \\ &= \sigma_{\max}(P)(\|e_1\| + \|e_2\|)\|e_1 - e_2\| \\ &\leq \sigma_{\max}(P)(\varepsilon_0 + \varepsilon_0)\|e_1 - e_2\| = [2\varepsilon_0\sigma_{\max}(P)]\,\|e_1 - e_2\|.\end{aligned}$$

which completes the proof. □

Through the following theorem, we guarantee the stability of the system which is the solution to Problem 1.

Theorem 5.1. *Let Assumption 5.1 hold. Suppose also that:*

1. *The OCP* (5.16a)-(5.16d) *is feasible for the initial time $t = 0$.*

2. *The terminal set* $\mathcal{E}_f \subseteq E$ *is closed, with* $0_{n+12} \in \mathcal{E}_f$.

3. *The terminal set* $\mathcal{E}_f$ *is chosen such that there exists an admissible control input* $u_f : [0, h] \to \mathcal{U}$ *such that for all* $e(s) \in \mathcal{E}_f$ *it holds that:*

   a) $e(s) \in \mathcal{E}_f, \forall\ s \in [0, h]$.

   b) $\dfrac{\partial V}{\partial e} f_e(e(s), u_f(s)) + F(e(s), u_f(s)) \leq 0, \forall\ s \in [0, h]$.

Then, the closed loop system (5.11)*, under the control input* (5.19)*, converges to the set* $\mathcal{E}_f$ *for* $t \to \infty$.

Proof. As usual in predictive control the proof consists of two parts: in the first part it is established that initial feasibility implies feasibility afterwards. Based on this result it is then shown that the error $e(t)$ converges to the terminal set $\mathcal{E}_f$.

Feasibility Analysis: Consider any sampling time instant t_i for which a solution exists. In between t_i and t_{i+1}, the optimal control input $\hat{u}^\star(s; e(t_i)), \forall s \in [t_i, t_{i+1})$ is implemented. According to (5.21), it holds that:

$$e(t_{i+1}) = \hat{e}(t_{i+1}; \hat{u}^\star(\cdot; e(t_i)), e(t_i)).$$

The remaining piece of the optimal control input $\hat{u}^\star(s; e(t_i)), s \in [t_{i+1}, t_i + T_p]$ satisfies the state and input constraints E, U, respectively. Furthermore,

$$\hat{e}(t_i + T_p; \hat{u}^\star(\cdot; e(t_i)), e(t_i)) \in \mathcal{E}_f,$$

and we know from Assumption 2b of Theorem 1 that for all $e(t) \in \mathcal{E}_f$, there exists at least one control input $u_f(\cdot)$ that renders the set $\mathcal{E}_f$ invariant over h. Picking any such input, a feasible control input $\bar{u}(\cdot; e(t_{i+1}))$, at time instant t_{i+1}, may be the following:

$$\bar{u}(s; e(t_{i+1})) := \begin{cases} \hat{u}^\star(s; e(t_i)), & s \in [t_{i+1}, t_i + T_p), \\ u_f(\hat{e}(t_i + T_p; u^\star(\cdot), e(t_i))), & s \in [t_i + T_p, t_{i+1} + T_p]. \end{cases} \tag{5.23}$$

Thus, from feasibility of $\hat{u}^\star(s, e(t_i))$ and the fact that $u_f(e(t)) \in U$, for all $e(t) \in \mathcal{E}_f$, it follows that:

$$\bar{u}(s; e(t_{i+1})) \in U, \forall\ s \in [t_{i+1}, t_i + T_p].$$

Hence, the feasibility at time t_i implies feasibility at time t_{i+1}. Therefore, if the OCP (5.16a) - (5.16d) is feasible at time $t = 0$, it remains feasible for every $t \geq 0$.

Convergence Analysis: The second part involves proving convergence of the state e in the terminal set $\mathcal{E}_f$. In order to prove this, it must be shown that a proper value function is decreasing along the solution trajectories starting at a sampling time t_i. Consider the optimal value function $J^\star(e(t_i))$, as is given in (5.20). Consider also the cost of the feasible control input, indicated by:

$$\bar{J}(e(t_{i+1})) := \bar{J}(e(t_{i+1}), \bar{u}(\cdot; e(t_{i+1}))). \tag{5.24}$$

Define:

$$\begin{aligned} u_1(s) &\coloneqq \bar{u}(s; e(t_{i+1})), \\ e_1(s) &\coloneqq \bar{e}(s; u_1(s), e(t_{i+1})), s > t_{i+1}, \end{aligned} \tag{5.25}$$

where $e_1(s)$ stands for the predicted state e at time s, based on the measurement of the state e at time t_{i+1}, while using the feasible control input $\bar{u}(s; e(t_{i+1}))$. Let us also define the following terms:

$$\begin{aligned} u_2(s) &= \hat{u}^\star(s; e(t_i)), \\ e_2(s) &= \hat{e}(s; u_2(s), e(t_i)), s > t_{i+1}. \end{aligned} \tag{5.26}$$

(5.25), (5.26) form convenient notations for the readability of the proof hereafter.

By employing (5.16a), (5.20) and (5.24), the difference between the optimal and feasible cost is given by:

$$\begin{aligned} &\bar{J}(e(t_{i+1})) - J^\star(e(t_i)) = V(e_1(t_{i+1} + T_p)) + \int_{t_{i+1}}^{t_{i+1}+T_p} \Big[F(e_1(s), u_1(s))\Big] ds \\ &- V(e_2(t_i + T_p)) - \int_{t_i}^{t_i+T_p} \Big[F(e_2(s), u_2(s))\Big] ds \\ &= V(e_1(t_{i+1} + T_p)) + \int_{t_{i+1}}^{t_i+T_p} \Big[F(e_1(s), u_1(s))\Big] ds - V(e_2(t_i + T_p)) + \\ &\int_{t_i+T_p}^{t_{i+1}+T_p} \Big[F(e_1(s), u_1(s))\Big] ds - \int_{t_i}^{t_{i+1}} \Big[F(e_2(s), u_2(s))\Big] ds \\ &- \int_{t_{i+1}}^{t_i+T_p} \Big[F(e_2(s), u_2(s))\Big] ds. \end{aligned} \tag{5.27}$$

Note that, from (5.23), the following holds:

$$\bar{u}(s; e(t_{i+1})) = \hat{u}^\star(s; e(t_i)), \forall\ s \in [t_{i+1}, t_i + T_p]. \tag{5.28}$$

By combining (5.25), (5.26) and (5.28), it yields that:

$$u_1(s) = u_2(s) = \bar{u}(s), \forall\ s \in [t_{i+1}, t_i + T_p], \tag{5.29}$$

which implies that:

$$e_1(s) = e_2(s), \forall\ s \in [t_{i+1}, t_i + T_p]. \tag{5.30}$$

The combination of (5.29) and (5.30) implies that:

$$F(e_1(s), u_1(s)) = F(e_1(s), u_1(s)), \forall\ s \in [t_{i+1}, t_i + T_p].$$

which implies that:

$$\int_{t_{i+1}}^{t_i+T_p} \Big[F(e_1(s), u_1(s))\Big] ds = \int_{t_{i+1}}^{t_i+T_p} \Big[F(e_2(s), u_2(s))\Big] ds. \tag{5.31}$$

By employing (5.31), (5.27) becomes:

$$\bar{J}(e(t_{i+1})) - J^{\star}(e(t_i)) = V(e_1(t_{i+1} + T_p)) + \int_{t_i+T_p}^{t_{i+1}+T_p} \Big[F(e_1(s), u_1(s))\Big] ds \\ - V(e_2(t_i + T_p)) - \int_{t_i}^{t_{i+1}} \Big[F(e_2(s), u_2(s))\Big] ds. \tag{5.32}$$

Due to the fact that $t_{i+1} + T_p - (t_i + T_p) = t_{i+1} - t_i = h$, and that Assumption 3b of Theorem 5.1 holds for one sampling period h, we obtain:

$$\begin{aligned}
&\int_{t_i+T_p}^{t_{i+1}+T_p} \Big[\frac{\partial V}{\partial e} f_e(e_1(s), u_1(s)) + F(e_1(s), u_1(s))\Big] ds \leq 0 \\
\Leftrightarrow &\int_{t_i+T_p}^{t_{i+1}+T_p} \Big[\dot{V}(e_1(s))\Big] ds + \int_{t_i+T_p}^{t_{i+1}+T_p} \Big[F(e_1(s), u_1(s))\Big] ds \leq 0 \\
\Leftrightarrow &V(e_1(t_{i+1} + T_p)) - V(e_1(t_i + T_p)) + \int_{t_i+T_p}^{t_{i+1}+T_p} \Big[F(e_1(s), u_1(s))\Big] ds \leq 0 \\
\Leftrightarrow &V(e_1(t_{i+1} + T_p)) - V(e_1(t_i + T_p)) + \int_{t_i+T_p}^{t_{i+1}+T_p} \Big[F(e_1(s), u_1(s))\Big] ds \\
&\quad \leq V(e_2(t_i + T_p)) - V(e_2(t_i + T_p)) \\
\Leftrightarrow &V(e_1(t_{i+1} + T_p)) + \int_{t_i+T_p}^{t_{i+1}+T_p} \Big[F(e_1(s), u_1(s))\Big] ds - V(e_2(t_i + T_p)) \\
&\quad \leq V(e_1(t_i + T_p)) - V(e_2(t_i + T_p)).
\end{aligned}$$

By employing the property $y \leq |y|, \forall y \in \mathbb{R}$, we get:

$$V(e_1(t_{i+1} + T_p)) + \int_{t_i+T_p}^{t_{i+1}+T_p} \Big[F(e_1(s), u_1(s))\Big] ds - V(e_2(t_i + T_p)) \\ \leq \left|V(e_1(t_i + T_p)) - V(e_2(t_i + T_p))\right|. \tag{5.33}$$

By employing Lemma 5.1, we have that:

$$|V(e_1(t_i + T_p)) - V(e_2(t_i + T_p))| \leq L_V \|e_1(t_i + T_p) - e_2(t_i + T_p)\|. \tag{5.34}$$

By combining (5.33) and (5.34) we get:

$$V(e_1(t_{i+1} + T_p)) + \int_{t_i+T_p}^{t_{i+1}+T_p} \Big[F(e_1(s), u_1(s))\Big] ds - V(e_2(t_i + T_p)) \\ \leq L_V \|e_1(t_i + T_p) - e_2(t_i + T_p)\| \tag{5.35}$$

For $s = t_i + T_p$, (5.30) gives:

$$e_1(t_i + T_p) = e_2(t_i + T_p). \tag{5.36}$$

By combining (5.36) and (5.35) we have:

$$V(e_1(t_{i+1}+T_p)) + \int_{t_i+T_p}^{t_{i+1}+T_p} \Big[F(e_1(s),u_1(s))\Big] ds - V(e_2(t_i+T_p)) \le 0. \tag{5.37}$$

By combining (5.32) with (5.37), the following holds:

$$\bar{J}(e(t_{i+1})) - J^\star(e(t_i)) \le -\int_{t_i}^{t_{i+1}} \Big[F(e_2(s),u_2(s))\Big] ds. \tag{5.38}$$

By substituting $e = e_2(s), u = u_2(s)$ in (5.18) we get:

$$F(e_2(s),u_2(s)) \ge m\|e_2(s)\|^2$$

or equivalently:

$$\begin{aligned}&\int_{t_i}^{t_{i+1}} \Big[F(e_2(s),u_2(s))\Big] ds \ge m\int_{t_i}^{t_{i+1}} \|e_2(s)\|^2 ds \\ \Leftrightarrow -&\int_{t_i}^{t_{i+1}} \Big[F(e_2(s),u_2(s))\Big] ds \le -m\int_{t_i}^{t_{i+1}} \|e_2(s)\|^2 ds.\end{aligned} \tag{5.39}$$

By combining (5.38) and (5.39) we finally get:

$$\bar{J}(e(t_{i+1})) - J^\star(e(t_i)) \le -m\int_{t_i}^{t_{i+1}} \|e_2(s)\|^2 ds. \tag{5.40}$$

It is clear that the optimal solution at time t_{i+1}, i.e., $J^\star(e(t_{i+1}))$, will not be worse than the feasible one at the same time i.e. $\bar{J}(e(t_{i+1}))$. Therefore, (5.40) implies:

$$J^\star(e(t_{i+1})) - J^\star(e(t_i)) \le -m\int_{t_i}^{t_{i+1}} \|e_2(s)\|^2 s \le 0, \tag{5.41}$$

or, by using the fact that $\displaystyle\int_{t_0}^{t_i} \|e_2(s)\|^2 ds = \sum_{j=0}^{i-1}\int_{t_j}^{t_{j+1}} \|e_2(s)\|^2 ds$, equivalently, we obtain:

$$J^\star(e(t_{i+1})) - J^\star(e(t_i)) \le -m\int_{t_0}^{t_{i+1}} \|e_2(s)\|^2 ds + m\sum_{j=0}^{i-1}\int_{t_j}^{t_{j+1}} \|e_2(s)\|^2 ds. \tag{5.42}$$

By using induction and the fact that $t_i = h \cdot i, t_{i+1} = h\cdot(i+1), \forall i \ge 0$, from (5.15), (5.42) is written as:

$$J^\star(e(t_i)) - J^\star(e(t_0)) \le -m\int_{t_0}^{t_i} \|e_2(s)\|^2 ds. \tag{5.43}$$

Since $t_0 = 0$ we obtain:

$$J^\star(e(t_i)) \le J^\star(e(0)) - m \int_0^{t_i} \|e_2(s)\|^2 ds. \tag{5.44}$$

which implies that:

$$J^\star(e(t_i)) \le J^\star(e(0)). \tag{5.45}$$

By combining (5.41), (5.45), we obtain:

$$J^\star(e(t_{i+1})) \le J^\star(e(t_i)) \le J^\star(e(0)), \forall\ t_i = i \cdot h, i \ge 0. \tag{5.46}$$

Therefore, the value function $J^\star(e(t_i))$ has proven to be non-increasing for all the sampling times. Let us define the function:

$$\widetilde{V}(e(t)) := J^\star(e(s)) \le J^\star(e(0)), t \in \mathbb{R}_{\ge 0}, \tag{5.47}$$

where $s = \max\{t_i : t_i \le t\}$. Since $J^\star(e(0))$ is bounded, (5.47) implies that $\widetilde{V}$ is bounded. Since the signals $e(t), u(t)$ are bounded ($e(t) \in E, u(t) \in U$), according to (5.11), it holds that $\dot{e}(t)$ is also bounded. From (5.44) we have that:

$$\widetilde{V}(e(t)) = J^\star(e(s)) \le J^\star(e(0)) - m \int_0^{s} \|e_2(s)\|^2 ds.$$

which due to the fact that $s \le t$, is equivalent to:

$$\widetilde{V}(e(t)) \le J^\star(e(0)) - m \int_0^{t} \|e_2(s)\|^2 ds, t \in \mathbb{R}_{\ge 0}. \tag{5.48}$$

From (5.48), we get:

$$\int_0^{t} \|e_2(s)\|^2 ds \le \frac{1}{m} \left[J^\star(e(0)) - \widetilde{V}(e(t)) \right], t \in \mathbb{R}_{\ge 0}. \tag{5.49}$$

Since $J^\star(e(0)), V(e(t))$ has been proven to be bounded, the term $\displaystyle\int_0^t \|e_2(s)\|^2 ds$ is also bounded. Therefore, by employing Lemma 2.2, we have that $\|e_2(t)\| \to 0$, as $t \to \infty$. The latter implies that:

$$\lim_{t\to\infty} \|e(t)\| = 0 \Rightarrow \lim_{t\to\infty} \|e(t)\| \in \mathcal{E}_f,$$

and leads to the conclusion of the proof. □

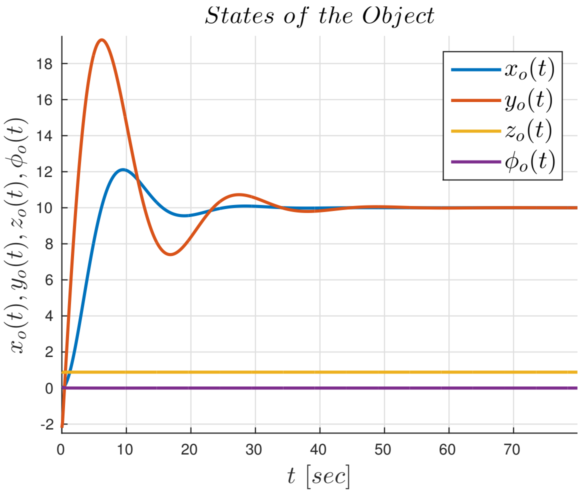


Figure 5.2: The states of the object.

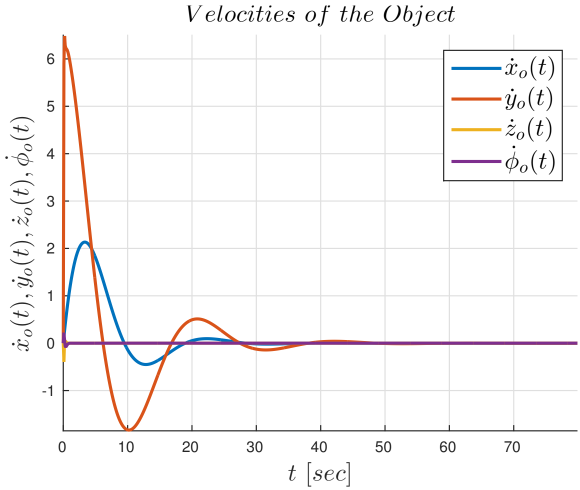


Figure 5.3: The velocities of the object.

5.2.3 Simulation Results

To demonstrate the efficiency of the proposed control protocol, we consider two simulation scenarios.

Scenario 1: Consider $N = 2$ ground vehicles equipped with 2 DOF manipulators, rigidly grasping an object with $n_1 = n_2 = 4, n = n_1 + n_2 = 8$. From (5.8) we have that $x = [x_O^\top, v_O^\top, q^\top]^\top \in \mathbb{R}^{16}, u \in \mathbb{R}^8$, with $x_O = [p_O^\top, \eta_O]^\top \in \mathbb{R}^4$, $v_O = [\dot{p}_O^\top, \omega_{x_O}]^\top \in \mathbb{R}^4$, $p_O = [x_O, y_O, z_O]^\top \in \mathbb{R}^3, q = [q_1^\top, q_2^\top]^\top \in \mathbb{R}^8$, $q_i = [p_{B_i}^\top, \alpha_i^\top]^\top \in \mathbb{R}^4$, $p_{B_i} = [x_{B_i}, y_{B_i}]^\top \in \mathbb{R}^2$, $\alpha_i = [\alpha_{i_1}, \alpha_{i_2}]^\top \in \mathbb{R}^2, i \in \{1, 2\}$. The manipulators become singular when $\sin(\alpha_{i_1}) = 0\}, i \in \{1, 2\}$, thus the state constraints for the manipulators are set to:

$$\varepsilon < \alpha_{1_1} < \frac{\pi}{2} - \varepsilon, -\frac{\pi}{2} + \varepsilon < \alpha_{1_2} < \frac{\pi}{2} - \varepsilon,$$
$$-\frac{\pi}{2} + \varepsilon < \alpha_{2_1} < -\varepsilon, -\frac{\pi}{2} + \varepsilon < \alpha_{2_2} < \frac{\pi}{2} - \varepsilon.$$

We also consider the input constraints:

$$-10 \leq u_{i,j}(t) \leq 10, i \in \{1, 2\}, j \in \{1, \ldots, 4\}.$$

The initial conditions are set to:

$$x_O(0) = \left[0, -2.2071, 0.9071, \frac{\pi}{2}\right]^\top, v_O(0) = [0, 0, 0, 0]^\top,$$
$$q_1(0) = \left[0, 0, \frac{\pi}{4}, \frac{\pi}{4}\right]^\top, q_2(0) = \left[0, -4.4142, -\frac{\pi}{4}, -\frac{\pi}{4}\right]^\top.$$

The desired goal states are set to:

$$x_{O,\text{des}} = \left[10, 10, 0.9071, \frac{\pi}{2}\right]^\top, v_{O,\text{des}} = [0, 0, 0, 0]^\top,$$
$$q_{1,\text{des}} = \left[10, 12.2071, \frac{\pi}{4}, \frac{\pi}{4}\right]^\top, q_{2,\text{des}} = \left[10, 7.7929, -\frac{\pi}{4}, -\frac{\pi}{4}\right]^\top.$$

We set an obstacle between the initial and the desired pose of the object, that is spherical with center $[5, 5, 1]$ and radius 2. The sampling time is $h = 0.1$ seconds, the horizon is set to $T_p = 0.3$ seconds, and the total simulation time is 80 seconds; The matrices P, Q, R are set to:

$$P = Q = 10 I_{16\times16}, R = 2 I_{8\times8}.$$

The simulation results are depicted in Fig. 5.2- Fig. 5.7, which shows that the states of the agents as well as the states of the object converge to the desired ones while guaranteeing that the obstacle is avoided and all state and input constraints are met.

Scenario 2: Consider $N = 3$ ground vehicles equipped with 2 DOF manipulators, rigidly grasping an object with $n_1 = n_2 = n_3 = 4, n = n_1 + n_2 + n_3 = 12$. From (5.8) we have that $x = [x_O^\top, v_O^\top, q^\top]^\top \in \mathbb{R}^{20}$, $u \in \mathbb{R}^{12}$, with $x_O = [p_O^\top, \eta_O]^\top \in \mathbb{R}^4$, $v_O = [\dot{p}_O^\top, \omega_{x_O}]^\top \in \mathbb{R}^4$, $p_O = [x_O, y_O, z_O]^\top \in \mathbb{R}^3$, $q = [q_1^\top, q_2^\top, q_3^\top]^\top \in \mathbb{R}^{12}$, $q_i = [p_{B_i}^\top, \alpha_i^\top]^\top \in \mathbb{R}^4$, $p_{B_i} = [x_{B_i}, y_{B_i}]^\top \in \mathbb{R}^2$, $\alpha_i = [\alpha_{i_1}, \alpha_{i_2}]^\top \in \mathbb{R}^2, i \in \{1, 2\}$. The manipulators become singular when $\sin(\alpha_{i_1}) = 0\}, i \in \{1, 2, 3\}$, thus the state constraints for the manipulators are set to:

$$\varepsilon < \alpha_{1_1} < \frac{\pi}{2} - \varepsilon, -\frac{\pi}{2} + \varepsilon < \alpha_{1_2} < \frac{\pi}{2} - \varepsilon,$$
$$-\frac{\pi}{2} + \varepsilon < \alpha_{2_1} < -\varepsilon, -\frac{\pi}{2} + \varepsilon < \alpha_{2_2} < \frac{\pi}{2} - \varepsilon.$$

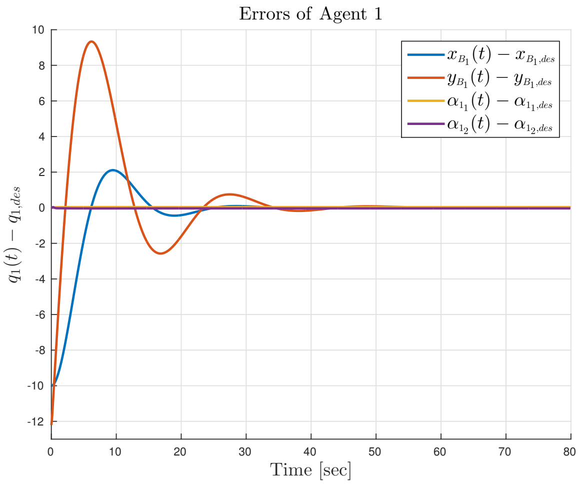


Figure 5.4: The errors of vehicle 1 as well as the errors of the manipulator.

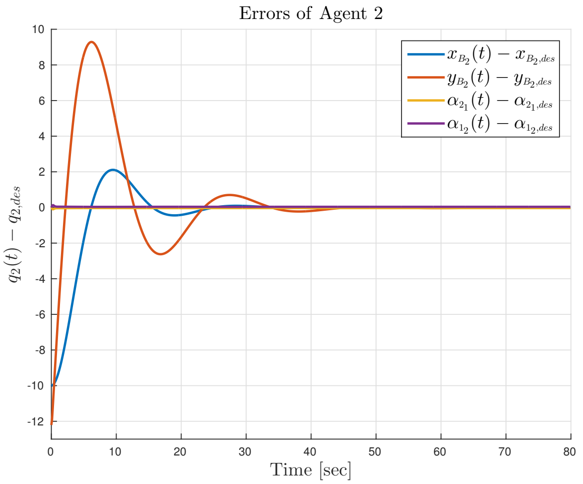


Figure 5.5: The errors of vehicle 2 as well as the errors of the manipulator.

We also consider the input constraints:

$$-10 \leq u_{i,j}(t) \leq 10, i \in \{1, 2\}, j \in \{1, \ldots, 4\}.$$

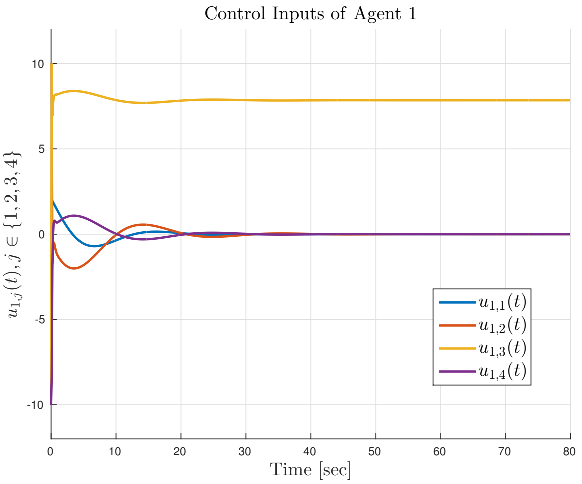


Figure 5.6: The control inputs of the actuators of agent 1.

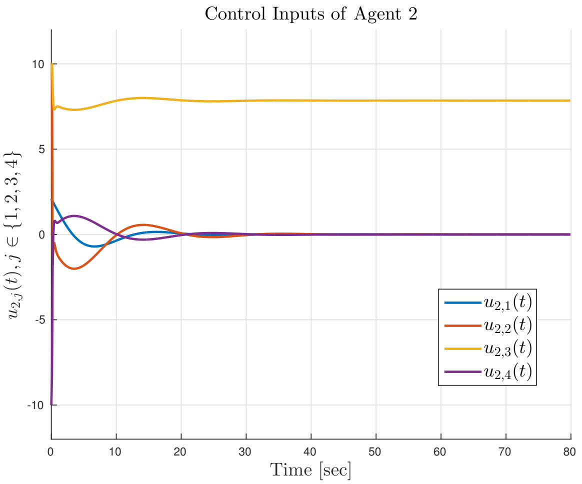


Figure 5.7: The control inputs of the actuators of agent 2.

The initial conditions are set to:

$$
\begin{aligned}
&x_O(0) = \left[0, -2.2071, 0.9071, \frac{\pi}{2}\right]^\top, v_O(0) = [0,0,0,0]^\top, \\
&q_1(0) = \left[0.5, 0, \frac{\pi}{4}, \frac{\pi}{4}\right]^\top, q_2(0) = \left[0, -4.4142, -\frac{\pi}{4}, -\frac{\pi}{4}\right]^\top, \\
&q_3(0) = \left[-0.5, 0, \frac{\pi}{4}, \frac{\pi}{4}\right]^\top.
\end{aligned}
$$

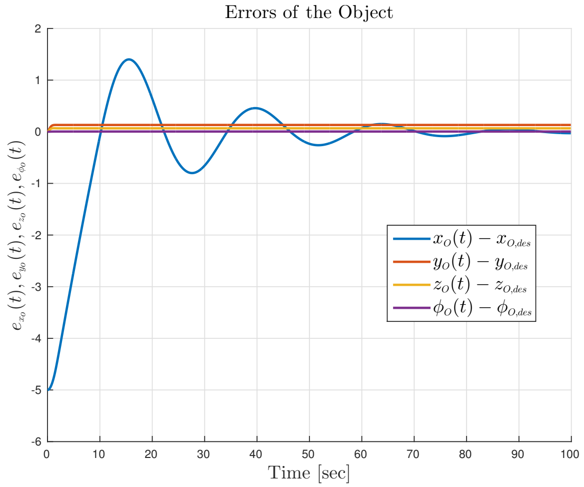


Figure 5.8: The errors of the object.

The desired goal states are set to:

$$x_{O,\text{des}} = \left[5, -2.2071, 0.9071, \frac{\pi}{2}\right]^\top, v_{O,\text{des}} = [0,0,0,0]^\top,$$
$$q_{1,\text{des}} = \left[5.5, 0, \frac{\pi}{4}, \frac{\pi}{4}\right]^\top, q_{2,\text{des}} = \left[5, -4.4142, -\frac{\pi}{4}, -\frac{\pi}{4}\right]^\top,$$
$$q_{3,\text{des}} = \left[4.5, 0.0, \frac{\pi}{4}, \frac{\pi}{4}\right]^\top.$$

The sampling time is $h = 0.1$ seconds, the horizon is set to $T_p = 0.5$ seconds, and the total simulation time is 100 seconds; The matrices P, Q, R are set to:

$$P = Q = 0.5I_{20\times 20}, R = 0.5I_{12\times 12}.$$

The terminal set is taken as a ball of radius 0.1 around 0 for both scenarios. The simulation results are depicted in Fig. 5.8- Fig. 5.15, which shows that the states of the agents as well as the states of the object converge to the desired ones while guaranteeing that all state and input constraints are met. The simulation scenarios were carried out by using the NMPC toolbox given in [119] and they took 23500, 45547 seconds for Scenario 1 and 2, respectively, in MATLAB Environment on a desktop computer with 8 cores, 3.60 GHz CPU and 16GB of RAM.

5.3 Decentralized Cooperative Transportation

The controller synthesize in the previous section is centralized, i.e., a central computer unit calculated all the MPC-based input signals for the agents. In this section, we aim at extending the results to a decentralized framework, where each agent calculates its own control signal.

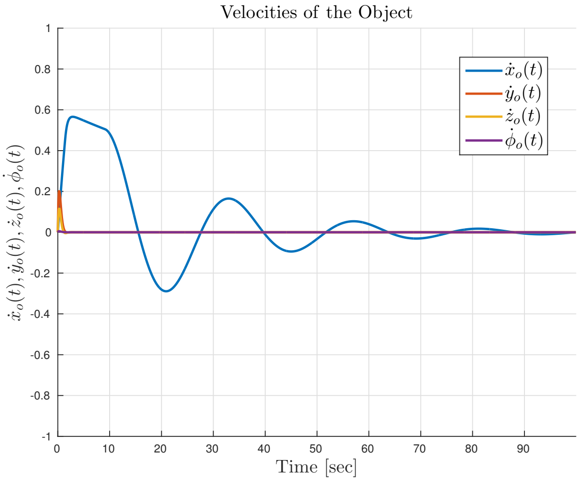


Figure 5.9: The velocities of the object.

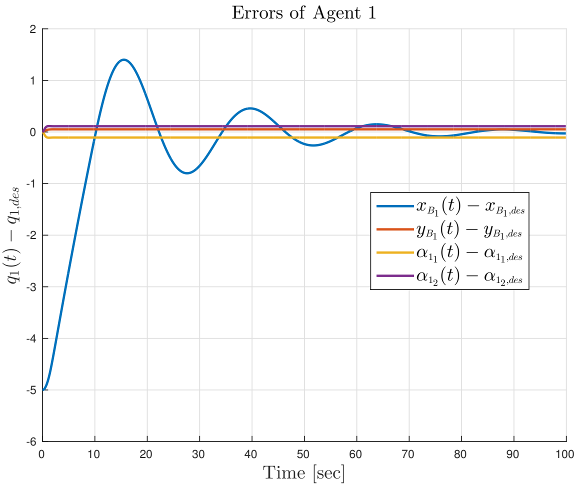


Figure 5.10: The errors of vehicle 1 as well as the errors of the manipulator.

5.3.1 Problem Formulation

The formulation we adopt here follows from the previous section, so we omit the derivations of the agents' and the object's dynamic modeling.

The grasping rigidity along with (5.7a) yields

$$v_O = v_{O_i}(q_i, \dot{q}_i) := J_{i_O}(q_i) v_i(q_i, \dot{q}_i), \tag{5.50}$$

for every $i \in \mathcal{N}$, where $J_{i_O}(q_i) := [J_{O_i}(q_i)]^{-1}$, with $J_{O_i}(q_i)$ as defined in (4.10).

Consider now the constants c_i, with $0 < c_i < 1$ and $\sum\limits_{i \in \mathcal{N}} c_i = 1$ that play

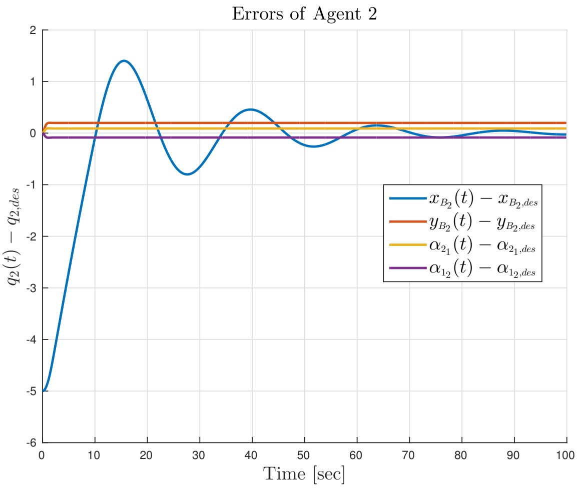


Figure 5.11: The errors of vehicle 2 as well as the errors of the manipulator.

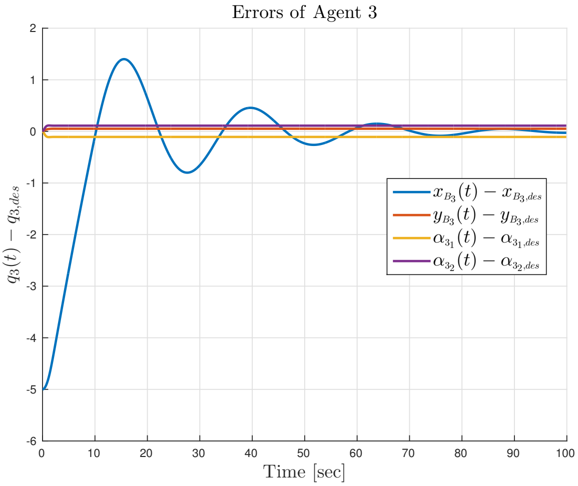


Figure 5.12: The errors of vehicle 3 as well as the errors of the manipulator.

the role of load sharing coefficients for the agents. Then (5.5b) can be written as: $\sum\limits_{i\in\mathcal{N}} c_i\Big\{M_O(x_{O_i}(q_i))\dot{v}_{O_i}(q_i,\dot{q}_i) + g_O(x_{O_i}(q_i))C_O(x_{O_i}(q_i), v_{O_i}(q_i,\dot{q}_i))v_{O_i}(q_i,\dot{q}_i)\Big\} = \sum\limits_{i\in\mathcal{N}}[J_{O_i}(q_i)]^\top\lambda_i$, from which, by employing the grasp coupling (see (4.14)), (5.2), (5.50), and after straightforward algebraic manipulations, we obtain the coupled dynamics

$$\sum_{i\in\mathcal{N}}\Big\{\widetilde{M}_i(q_i)\ddot{q}_i + \widetilde{C}_i(q_i,\dot{q}_i)\dot{q}_i + \widetilde{g}_i(q_i)\Big\} = \sum_{i\in\mathcal{N}}[J_{O_i}(q_i)]^\top u_i, \tag{5.51}$$

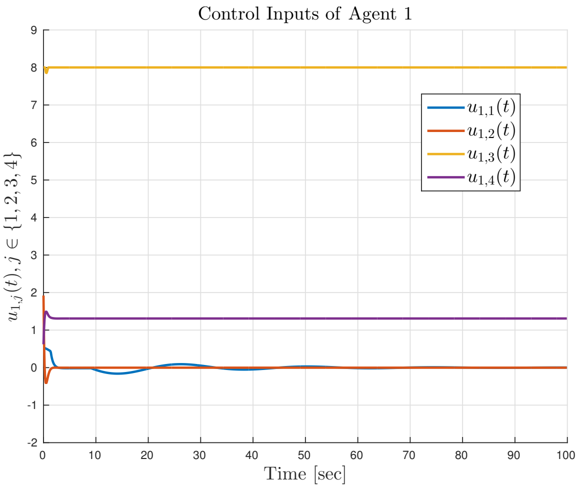


Figure 5.13: The control inputs of the actuators of agent 1.

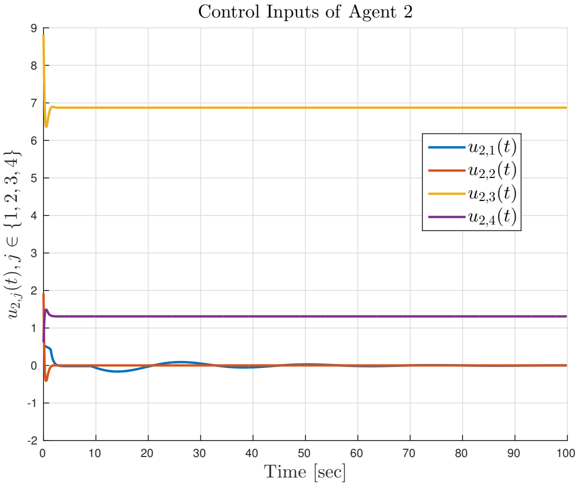


Figure 5.14: The control inputs of the actuators of agent 3.

where:

$$
\begin{aligned}
&\widetilde{M}_i(q_i) := c_i M_O(x_{O_i}(q_i)) J_{i_O}(q_i) J_i(q_i) + [J_{O_i}(q_i)]^\top M_i(q_i) J_i(q_i),\\
&\widetilde{C}_i(q_i,\dot{q}_i) := [J_{O_i}(q_i)]^\top \Big(M_i(q_i)\dot{J}_i(q_i) + C_i(q_i,\dot{q}_i) J_i(q_i) \Big) +\\
&c_i M_O(x_{O_i}(q_i)) J_{i_O}(q_i) \dot{J}_i(q_i) + c_i M_O(x_{O_i}(q_i)) \dot{J}_{i_O}(q_i) J_i(q_i),\\
&+ c_i C_O(x_{O_i}(q_i), v_{O_i}(q_i,\dot{q}_i)),\\
&\widetilde{g}_i(q_i) := c_i g_O(x_{O_i}(q_i)) + [J_{O_i}(q_i)]^\top g_i(q_i), i \in \mathcal{N}.
\end{aligned}
$$

The problem in hand in this section is the same as Problem 5.1, with the extra

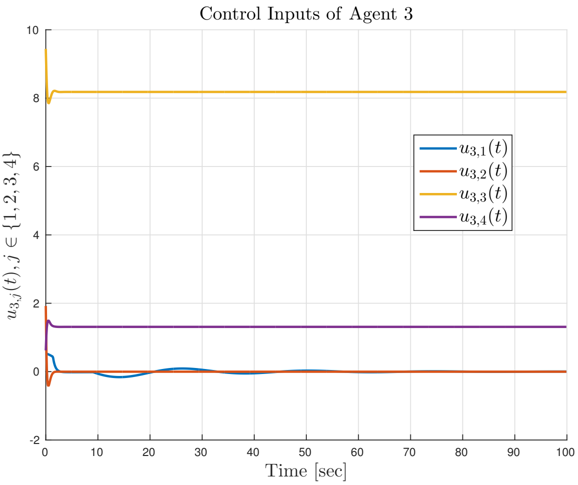


Figure 5.15: The control inputs of the actuators of agent 3.

constraint that the control design needs to be decentralized. For that, we need the following assumption regarding the agent communication:

Assumption 5.2. (Sensing and communication capabilities) Each agent $i \in \mathcal{N}$ is able to continuously measure the other agents' state $q_j, \dot{q}_j$, $j \in \mathcal{N}\backslash\{i\}$. Moreover, each agent $i \in \mathcal{N}$ is able to communicate with the other agents $j \in \mathcal{N}\backslash\{i\}$ without any delays.

Note that the aforementioned sensing assumption is reasonable, since in cooperative manipulation tasks, the agents are sufficiently close to each other, and therefore potential sensing radii formed by realistic sensors are large enough to cover them. Moreover, each agent $i \in \mathcal{N}$ can construct at every time instant the set-valued functions $\mathcal{A}_j(q_j)$, $\forall j \in \mathcal{N}\backslash\{i\}$, whose structure can be transmitted off-line to all agents.

Along with the sets $\mathcal{S}_{i,O}$, $\mathcal{S}_{i,A}$ defined in the previous section, we also define $\mathcal{S}_{O_i} := \{q_i \in \mathbb{R}^{n_i} : \mathcal{C}_O(x_{O_i}(q_i)) \cap \mathcal{O}_z \neq \emptyset\}$, $\forall i \in \mathcal{N}$, as well as the projection sets for agent i $\widetilde{\mathcal{S}}_{i,A}([q_\ell]_{\ell \in \mathcal{N}\backslash\{i\}}) := \{q_i \in \mathbb{R}^{n_i} : q \in \mathcal{S}_{i,A}\}$, $\forall i \in \mathcal{N}$, where the notation $[q_\ell]_{\ell \in \mathcal{N}\backslash\{i\}}$ stands for the stack vector of all $q_\ell, \ell \in \mathcal{N}\backslash\{i\}$.

5.3.2 Main Results

In this section, a systematic solution to Problem 1 is introduced, based on NMPC. The proposed methodology is decentralized, since we do not consider a centralized system that calculates all the control signals and transmits them to the agents, like in the previous section. To achieve that, we employ a leader-follower perspective. More specifically, as will be explained in the sequel, at each sampling time, a leader agent solves part of the coupled dynamics (5.51) via an NMPC scheme, and transmits

its predicted variables to the rest of the agents. Assume, without loss of generality, that the leader corresponds to agent $i = 1$. Loosely speaking, the proposed solution proceeds as follows: agent 1 solves, at each sampling time step, the receding horizon model predictive control subject to the forward dynamics:

$$\widetilde{M}_1(q_1)\ddot{q}_1 + \widetilde{C}_1(q_1, \dot{q}_1)\dot{q}_1 + \widetilde{g}(q_1) = [J_{O_1}(q_1)]^\top u_1, \tag{5.52}$$

and a number of inequality constraints, as will be clarified later. After obtaining a control input sequence and a set of predicted variables for $q_1, \dot{q}_1$, denoted as $\hat{q}_1, \hat{\dot{q}}_1$, it transmits the corresponding predicted state for the object $x_{O_1}(\hat{q}_1), v_{O_1}(\hat{q}_1, \hat{\dot{q}}_1)$ for the control horizon to the other agents $\{2, \dots, N\}$. Then, the followers solve the receding horizon NMPC subject to the forward dynamics:

$$\widetilde{M}_i(q_i)\ddot{q}_i + \widetilde{C}_i(q_i, \dot{q}_i)\dot{q}_i + \widetilde{g}(q_i) = [J_{O_i}(q_i)]^\top u_i, \tag{5.53}$$

the state equality constraints:

$$x_{O_i}(q_i) = x_{O_1}(\hat{q}_1), v_{O_i}(q_i, \dot{q}_i) = v_{O_1}(\hat{q}_1, \hat{\dot{q}}_1), \tag{5.54}$$

$i \in \{2, \dots, N\}$ as well as a number of inequality constraints that incorporate obstacle and inter-agent collision avoidance. More specifically, we consider that there is a priority sequence among the agents, which we assume, without loss of generality, that is defined by $\{1, \dots, N\}$. Each agent, after solving its optimization problem, transmits its calculated predicted variables to the agents of lower priority, which take them into account for collision avoidance. Note that the coupled object-agent dynamics are implicitly taken into account in equations (5.52), (5.53) in the following sense. Although the coupled model (5.51) does not imply that each one of these equations is satisfied, by forcing each agent to comply with the specific dynamics through the optimization procedure, we guarantee that (5.51) is satisfied, since it's the result of the addition of (5.52) and (5.53), for $i = 1$ and every $i \in \{2, \dots, N\}$, respectively. Intuitively, the leader agent is the one that determines the path that the object will navigate through, and the rest of the agents are the followers that contribute to the transportation. Moreover, the equality constraints (5.54) guarantee that the predicted variables of the agents $\{2, \dots, N\}$ will comply with the rigidity at the grasping points through the equality constraints (5.54).

By using the notation $x_i := [x_{i_1}^\top, x_{i_2}^\top]^\top := [q_i^\top, \dot{q}_i^\top]^\top \in \mathbb{R}^{2n_i}$, $i \in \mathcal{N}$, the nonlinear dynamics of each agent can be written as:

$$\dot{x}_i = \widetilde{f}_i(x_i, u_i) := \begin{bmatrix} \widetilde{f}_{i_1}(x_i) \\ \widetilde{f}_{i_2}(x_i, u_i) \end{bmatrix}, \tag{5.55}$$

where $\widetilde{f}_i : E_i \times \mathbb{R}^6 \to \mathbb{R}^{2n_i}$ is the locally Lipschitz function: $\widetilde{f}_{i_1}(x_i, u_i) := x_{i_2}$, $\widetilde{f}_{i_2}(x_i, u_i) := \widehat{M}_i(q_i)\Big([J_{O_i}(q_i)]^\top u_i - \widetilde{C}_i(q_i, \dot{q}_i)\dot{q} - \widetilde{g}_i(q_i)\Big)$, $i \in \mathcal{N}$, where $\widehat{M}_i : \mathbb{R}^{n_i} \backslash \mathcal{Q}_i \to \mathbb{R}^{n_i \times 6}$, is the pseudo-inverse $\widehat{M}_i(q_i) := [\widetilde{M}_i]^\top(q_i)\Big(\widetilde{M}_i(q_i)[\widetilde{M}_i(q_i)]^\top\Big)^{-1}$, and $E_i :=$

$\mathbb{R}^{n_i}\backslash\mathcal{Q}_i \times \mathbb{R}^{n_i}$, $\forall i \in \mathcal{N}$. It can be proved that in the set $\mathbb{R}^{n_i}\backslash\mathcal{Q}_i$ the matrix $\widetilde{M}_i(q_i)[\widetilde{M}_i(q_i)]^\top$ has full rank and hence, $\widehat{M}_i(q_i)$ is well defined for all $q \in \mathbb{R}^{n_i}\backslash\mathcal{Q}_i$. By abusing the notation with respect to the proof of Theorem 5.1, we define the error vector $e_1 : E_1 \to \mathbb{M} \times \mathbb{R}^6$, as:

$$e_1(x_1) := \begin{bmatrix} x_{O_1}(q_1) - x_{\text{des}} \\ v_{O_1}(q_1, \dot{q}_1), \end{bmatrix}$$

which gives us the *error dynamics*:

$$\dot{e}_1 = g_1(x_1, u_1), \tag{5.56}$$

with $g_1 : E_1 \times \mathbb{R}^6 \to \mathbb{R}^{2n_i}$:

$$g_1(x_1, u_1) := \begin{bmatrix} [J_O(\eta_{O_1}(q_1))]J_{1_O}(q_1)J_1(q_1)\dot{q}_1 \\ J_{1_O}(q_1)J_1(q_1)f_{1_2}(x_1, u_1) + \Big(J_{1_O}\dot{J}_1(q_1) + \dot{J}_{1_O}(q_1)J_1(q_1)\Big)\dot{q}_1, \end{bmatrix}$$

where we employed (5.56) and the object dynamics.

Remark 5.4. It can be concluded that $g_1(\cdot, u_1)$ is *Lipschitz continuous* in E_1 since it is continuously differentiable in its domain. Thus, for every $x_1, x_1' \in E_1$, with $x_1 \neq x_1'$, there exists a Lipschitz constant L_g such that: $|g(x_1, u) - g(x_1', u)| \leq L_g \|x_1 - x_1'\|$.

The time derivative of joint space inputs is given by: $\dot{\tau}_i = [\dot{J}_i(q_i)]^\top u_i + [J_i(q_i)]^\top \dot{u}_i$. Hence, the constraints for τ_{i_k} and $\dot{\tau}_{i_k}$, $k \in \mathbb{R}^{n_i}$,$i \in \mathcal{N}$, can be written as coupled state-input constraints: $\|\tau_i\| \leq \bar{\tau}_i \Leftrightarrow \|[J(q_i)]^\top u_i\| \leq \bar{\tau}_i$, $\|\dot{\tau}_i\| \leq \bar{\dot{\tau}}_i \Leftrightarrow \|[\dot{J}_i(q_i)]^\top u_i + [J_i(q_i)]^\top \dot{u}_i\| \leq \bar{\dot{\tau}}_i$. Let us now define the following sets $U_i \subseteq \mathbb{R}^{6 \times 6 \times (2n_i)}$:

$$\begin{aligned} U_i := \Big\{&(u_i, \dot{u}_i, x_i) \in \mathbb{R}^{6 \times 6 \times (2n_i)} : \|[J(q_i)]^\top u_i\| \leq \bar{\tau}_i, \\ &\|[\dot{J}_i(q_i)]^\top u_i + [J_i(q_i)]^\top \dot{u}_i\| \leq \bar{\dot{\tau}}_i\Big\}, i \in \mathcal{N}, \end{aligned} \tag{5.57}$$

as the sets that capture the control input constraints of (5.55) (note that, compared to the previous section, these sets capture also constraints of the input rate), as well as their projections

$$U_{i,u} := \big\{u_i \in \mathbb{R}^6 : (u_i, \dot{u}_i, x_i) \in U_i\big\}, i \in \mathcal{N}. \tag{5.58}$$

Define also the set-valued functions $X_i : \mathbb{R}^{n-n_i} \rightrightarrows \mathbb{R}^{2n_i}, i \in \mathcal{N}$, by:

$$\begin{aligned} &X_1([q_\ell]_{\ell \in \{2,\ldots,N\}}) := \Big\{x_1 \in \mathbb{R}^{2n_1} : \theta_{O_1}(q_1) \in [-\bar{\theta}, \bar{\theta}], \theta_{B_1} \in [-\bar{\theta}, \bar{\theta}], |\dot{q}_{k_1}| \leq \bar{\dot{q}}_1, \\ &q_1 \in \widetilde{\mathcal{Q}}_1\backslash\Big(\mathcal{S}_{1,O} \cup \widetilde{\mathcal{S}}_{1,A}([q_\ell]_{\ell \in \{2,\ldots,N\}})\Big), x_{O_1}(q_1) \in \mathbb{R}^3\backslash S_{O_1}\Big\} \\ &X_i([q_\ell]_{\ell \in \mathcal{N}\backslash\{i\}}) := \Big\{x_i \in \mathbb{R}^{2n_i} : \theta_{B_i} \in [-\bar{\theta}, \bar{\theta}], |\dot{q}_{k_i}| \leq \bar{\dot{q}}_i, \\ &q_i \in \widetilde{\mathcal{Q}}_i\backslash\left(\mathcal{S}_{i,O} \cup \mathcal{S}_{i,A}([q_\ell]_{\ell \in \mathcal{N}\backslash\{i\}})\right)\Big\}, \end{aligned}$$

$\forall i \in \{2, \dots, N\}$. Note that $q_i \in X_i([q_\ell]_{\ell \in \mathcal{N}\setminus\{i\}}) \implies q_i \notin \mathcal{Q}_i, \forall i \in \mathcal{N}$.

The sets X_i capture all the state constraints of the system dynamics (5.55), i.e., representation- and singularity-avoidance, collision avoidance among the agents and the obstacles, as well as collision avoidance of the object with the obstacles, which is assigned to the leader agent only. We further define the set-valued functions $\mathcal{E}_1 : \mathbb{R}^{n-n_1} \rightrightarrows \mathbb{M} \times \mathbb{R}^6$ as $\mathcal{E}_1([q_\ell]_{\ell\in\{2,\dots,N\}}) \coloneqq \{e_1(x_1) \in \mathbb{M}\times\mathbb{R}^6 : x_1 \in X_1([q_\ell]_{\ell\in\{2,\dots,N\}})\}$, which now represent the constraints sets for the NMPC scheme.

The main problem at hand is the design of a *feedback control law* $u_1 \in U_1$ for agent 1 which guarantees that the error signal e_1 with dynamics given in (5.56), satisfies $\lim_{t\to\infty} \|e_1(x_1(t))\| \to 0$, while ensuring singularity avoidance, collision avoidance between the agents, between the agents and the obstacles as well as the object and the obstacles. The role of the followers $\{2, \dots, N\}$ is, through the load-sharing coefficients $c_2, \dots, c_N$ in (5.51), to contribute to the object trajectory execution, as derived by the leader agent 1. In order to solve the aforementioned problem, we propose a NMPC scheme, that is presented hereafter.

Consider a sequence of sampling times $\{t_j\}$, $j \in \mathbb{N}$ as defined in the previous section. For agent 1, the open-loop input signal applied in between the sampling instants is given by the solution of the following FHOCP:

$$\min_{\hat{u}_1(\cdot)} J_1(e_1(x_1(t_j)), \hat{u}_1(\cdot)) = \min_{\hat{u}_1(\cdot)} \Bigg\{ V_1(e_1(\hat{x}_1(t_j+T_p))) + \int_{t_j}^{t_j+T_p} \Big[F_1(e_1(\hat{x}_1(s)), \hat{u}_1(s))\Big] ds \Bigg\} \tag{5.59a}$$

subject to:

$$\dot{e}_1(\hat{x}_1(s)) = g_1(\hat{x}_1(s), \hat{u}_1(s)),\ e_1(\hat{x}_1(t_j)) = e_1(x_1(t_j)), \tag{5.59b}$$

$$e_1(\hat{x}_1(s)) \in \mathcal{E}_1([q_\ell(t_j)]_{\ell\in\{2,\dots,N\}}), s \in [t_j, t_j+T_p], \tag{5.59c}$$

$$(\hat{u}_1(s), \dot{\hat{u}}_1(s), \hat{x}_1(s)) \in U_1, s \in [t_j, t_j+T_p], \tag{5.59d}$$

$$e_1(\hat{x}_1(t_j+T_p)) \in \mathcal{F}_1([q_\ell]_{\ell\in\{2,\dots,N\}}). \tag{5.59e}$$

At a generic time t_j then, agent 1 solves the aforementioned FHOCP. In the following, we use the notation $\mathcal{E}_1(\cdot)$ instead of $\mathcal{E}_1([q_\ell]_{\ell\in\{2,\dots,N\}})$ for brevity. The functions $F_1 : \mathcal{E}_1(\cdot)\times U_{1,u} \to \mathbb{R}_{\geq 0}$, $V_1 : \mathcal{E}_1(\cdot) \to \mathbb{R}_{\geq 0}$ stand for the *running cost* and the *terminal penalty cost*, respectively, and they are defined as: $F_1\big(e_1, u_1\big) \coloneqq e_1^\top Q_1 e_1 + u_1^\top R_1 u_1$, $V_1\big(e_1\big) \coloneqq e_1^\top P_1 e_1$; $R_1 \in \mathbb{R}^{6\times 6}$ and $P_1 \in \mathbb{R}^{(2n_1)\times(2n_1)}$ are symmetric and positive definite controller gain matrices to be appropriately tuned; $Q_1 \in \mathbb{R}^{(2n_1)\times(2n_1)}$ is a symmetric and positive semi-definite controller gain matrix to be appropriately tuned. The bounded *terminal* set is defined here as $\mathcal{F}_1$. Note that, similarly to (5.18), there exists class K_∞ functions α_1, α_2 such that $\alpha_1(\|[e_1^\top, u_1^\top]^\top\|) \leq F_1(e_1, u_1) \leq \alpha_2(\|[e_1^\top, u_1^\top]^\top\|)$, $\forall [e_1^\top, u_1^\top]^\top \in \mathcal{E}_1(\cdot)\times U_{1,u}$.
The *terminal set* $\mathcal{F}_1(\cdot)$ is chosen as: $\mathcal{F}_1([q_\ell]_{\ell\in\{2,\dots,N\}}) \coloneqq \{e_1 \in \mathcal{E}_1([q_\ell]_{\ell\in\{2,\dots,N\}}) : V_1(e_1) \leq \epsilon_1\}$, where $\epsilon_1 \in \mathbb{R}_{>0}$ is an arbitrarily small constant to be appropriately tuned. Moreover, similarly to the case of 5.2, it can be proved that the terminal

penalty function V_1, is Lipschitz continuous in $\mathcal{F}_1(\cdot)$, and it holds that: $\big|V_1(e_1) - V_1(e_1')\big| \leq L_{V_1}\|e_1 - e_1'\|, \forall e_1, e_1' \in \mathcal{F}_1(\cdot)$, where $L_{V_1} := 2\sigma_{\max}(P_1)\sup\{\|e_1\| : e_1 \in \mathcal{F}_1\}\|e_1\|$.

The solution to FHOCP (5.59a) - (5.59e) at time t_j provides an optimal control input, denoted by $\hat{u}_1^{\star}(s;\ e_1(x_1(t_j)), x_1(t_j))$, $s \in [t_j, t_j + T_p]$. This control input is then applied to the system until the next sampling instant t_{j+1}:

$$u_1\left(s;\ x_1(t_j), e_1(x_1(t_j))\right) = \hat{u}_1^{\star}\left(s;\ x_1(t_j), e_1(x_1(t_j))\right), \tag{5.60}$$

for every $s \in [t_j, t_j + h)$. At time $t_{j+1} = t_j + h$ a new FHOCP is solved in the same manner, leading to a receding horizon approach. The control input $u_1(\cdot)$ is of feedback form, since it is recalculated at each sampling instant based on the then-current state. The solution of (5.56) at time s, $s \in [t_j, t_j + T_p]$, starting at time t_j, from an initial condition $x_1(t_j), e_1(x_1(t_j))$, by application of the control input $u_1 : [t_j, s] \to U_{1,u}$ is denoted by $e_1\big(x_1(s);\ u_1(\cdot);\ x_1(t_j), e_1(x_1(t_j))\big)$, $s \in [t_j, t_j + T_p]$. The *predicted* state of the system (5.59b) at time s, $s \in [t_j, t_j + T_p]$ based on the measurement of the state at time t_j, $x_1(t_j)$, by application of the control input $u_1\big(t;\ x_1(t_j), e_1(x_1(t_j))\big)$ as in (5.60), is denoted by $\hat{x}_1\big(s;\ u_1(\cdot);\ x_1(t_j), e_1(x_1(t_j))\big)$, and the corresponding predicted error by $e_1(\hat{x}_1(\cdot);\ u_1(\cdot);\ x_1(t_j), e_1(x_1(t_j))\big)$, $s \in [t_j, t_j + T_p]$.

After the solution of the FHOCP and the calculation of the predicted states $\hat{x}_1\big(s;\ u_1(\cdot), e_1(x_1(t_j)), x_1(t_j)\big)$, $s \in [t_j, t_j + T_p]$ at each time instant t_j, agent 1 transmits the values $\hat{q}_1(s, \cdot)$, $\dot{\hat{q}}_1(s, \cdot)$ as well as $x_{O_1}(\hat{q}_1(s, \cdot))$ and $v_{O_1}(\hat{q}_1(s, \cdot), \dot{\hat{q}}_1(s, \cdot))$, as computed by (5.7), (5.50), $\forall s \in [t_j, t_j + T_p]$ to the rest of the agents $\{2, \ldots, N\}$. The rest of the agents then proceed as follows. Each agent $i \in \{2, \ldots, N\}$, solves the following FHOCP:

$$\min_{\hat{u}_i(\cdot)} J_i(x_i(t_j)), \hat{u}_i(\cdot)) \tag{5.61a}$$

subject to:

$$\dot{x}_i = \widetilde{f}_i(x_i(s), u_i(s)), \tag{5.61b}$$

$$x_i(s) \in X_i\Big([q_\ell(t_j)]_{\ell \in \{i+1, \ldots, N\}}\Big), \tag{5.61c}$$

$$x_i(s) \in X_i\Big([\hat{q}_\ell(s, \cdot)]_{j \in \{1, \ldots, i-1\}}\Big), \tag{5.61d}$$

$$x_{O_i}(q_i(s)) = x_{O_1}(\hat{q}_1(s; \cdot)), \tag{5.61e}$$

$$v_{O_i}(q_i(s), \dot{q}_i(s)) = v_{O_1}(\hat{q}_1(s; \cdot), \dot{\hat{q}}_1(s; \cdot)), \tag{5.61f}$$

$$(u_i(s), \dot{u}_i(s), x_i(s)) \in U_i, s \in [t_j, t_j + T_p], \tag{5.61g}$$

at every sampling time t_j. Note that, through the equality constraints (5.61e), (5.61f), the follower agents must comply with the trajectory computed by the leader $\hat{q}_1(s, \cdot), \dot{\hat{q}}_1(s, \cdot)$. This can be problematic in the sense that this trajectory might drive the followers to collide with an obstacle or among each other. Resolution of such cases is not in the scope of this paper and constitutes part of future research. We state that with the following assumption:

Assumption 5.3. The sets $\{(q,s) \in \mathbb{R}^n \times [t_j, t_j + T_p] : x_{O_i}(q_i(s)) = x_{O_1}(\hat{q}_1(s;\cdot)),$ $v_{O_i}(q_i(s), \dot{q}_i(s)) = v_{O_1}(\hat{q}_1(s;\cdot), \hat{\dot{q}}_1(s;\cdot))\} \ \cap \ \mathcal{S}_{i,O} \ \cap \ \widetilde{\mathcal{S}}_{i,A}([q_\ell(t_j)]_{\ell \in \{i+1,\dots,N\}}) \ \cap$ $\widetilde{\mathcal{S}}_{i,A}([q_\ell(s)]_{\ell \in \{1,\dots,i-1\}})$ are nonempty, $\forall i \in \{2,\dots,N\}$.

Next, similarly to the leader agent $i = 1$, it calculates the predicted states $\hat{q}_i(s,\cdot), \hat{\dot{q}}_i(s,\cdot), s \in [t_j, t_j + T_p]$, which then transmits to the agents $\{i+1,\dots,N\}$. In that way, at each time instant t_j, each agent $i \in \{2,\dots,N\}$ measures the other agents' states (as stated in Assumption 5.2), incorporates the constraint (5.61c) for the agents $\{i+1,\dots,N\}$, receives the predicted states $\hat{q}_\ell(s,\cdot), \hat{\dot{q}}_\ell(s,\cdot)$ from the agents $\ell \in \{2,\dots,i-1\}$ and incorporates the collision avoidance constraint (5.61d) for the entire horizon. Loosely speaking, we consider that each agent $i \in \mathcal{N}$ takes into account the first state of the next agents in priority ($q_\ell(t_j), \ell \in \{i+1,\dots,N\}$), as well as the transmitted predicted variables $\hat{q}_\ell(s,\cdot), \ell \in \{1,\dots,i-1\}$ of the previous agents in priority, for collision avoidance. Intuitively, the leader agent executes the planning for the followed trajectory of the object's center of mass (through the solution of the FHOCP (5.59a)-(5.59e)), the follower agents contribute in executing this trajectory through the load sharing coefficients c_i (as indicated in the coupled model (5.51)), and the agents low in priority are responsible for collision avoidance with the agents of higher priority. Moreover, the aforementioned equality constraints (5.61e), (5.61f) as well as the forward dynamics (5.61a) guarantee the compliance of all the followers with the model (5.51). For the followers, the cost $J_i(x_i(t_j), \hat{u}_i(\cdot))$ can be selected as any function of x_i, u_i, $\forall i \in \{2,\dots,N\}$.

Therefore, given the constrained FHOCP (5.61a)-(5.61g), the solution of problem lies in the capability of the leader agent to produce a state trajectory that guarantees $x_{O_1}(q_1(t)) \to x_{\text{des}}$, by solving the FHOCP (5.59a)-(5.59e), which is discussed in Theorem 5.2.

Remark 5.5. Note that, if the satisfaction of the equality constraints (5.61e), (5.61f) guarantees that there is no collision among the agents (e.g., in the case that two agents grasp a large object from two symmetrical - with respect to the object's center of mass - grasping points), then the transmission of the predicted variables among the follower agents $\{2,\dots,N\}$ is not needed. In that case, the followers can solve the problem (5.61a) - (5.61g) simultaneously, reducing thus the overall computation time.

We redefine now the admissible control input, in order to be consistent with this section's notation, and provide the theorem that summarizes the main results.

Definition 5.2. A control input $u_1 : [t_j, t_j + T_p] \to \mathbb{R}^m$ for a state $e_1(x_1(t_j))$ is called *admissible* for the FHOCP (5.59a)-(5.59e) if the following hold: 1) $u_1(\cdot)$ is piecewise continuous; 2) $u_1(s) \in U_{1,u}, \forall s \in [t_j, t_j + T_p]$; 3) $e_1\big(x_1(s);\ u_1(\cdot);\ x_1(t_j), e_1(x_1(t_j))\big) \in \mathcal{E}_1(\cdot), \forall\ s \in [t_j, t_j + T_p]$, and 4) $e_1\big(x_1(t_j + T_p);\ u_1(\cdot);\ x_1(t_j), e_1(x_1(t_j))\big) \in \mathcal{F}_1(\cdot)$.

Theorem 5.2. *Suppose that:* 1) *Assumption 5.1 - 5.3 hold;* 2) *The FHOCP* (5.59a)-(5.59e) *is feasible for the initial time* $t = 0$*;* 3) *There exists an admissible control* *input* $\kappa_1 : [t_j + T_p, t_{j+1} + T_p] \to U_1$ *such that for all* $e_1 \in \mathcal{F}_1(\cdot)$ *and for every* $s \in [t_j + T_p, t_{j+1} + T_p]$ *it holds that:* $e_1(x_1(s)) \in \mathcal{F}_1(\cdot)$ *and* $\dfrac{\partial V_1}{\partial e_1} g_1(e_1(x_1(s)), \kappa_1(s)) + F_1(e_1(x_1(s)), h_1(s)) \leq 0$*. Then, the system* (5.56)*, under the control input* (5.60)*, converges to the set* $\mathcal{F}_1(\cdot)$ *when* $t \to \infty$*.*

Proof. The proof is similar to the proof of Theorem 5.1 and is omitted. □

5.3.3 Simulation Results

To demonstrate the efficiency of the proposed control protocol, we consider a simulation example with $N = 3$ ground vehicles equipped with 2 DOF manipulators, rigidly grasping an object with $n_1 = n_2 = n_3 = 4$, $n = n_1 + n_2 + n_3 = 12$. The states of the agents are given as: $q_i = [p_{B_i}^\top, \alpha_i^\top]^\top \in \mathbb{R}^4$, $p_{B_i} = [x_{B_i}, y_{B_i}]^\top \in \mathbb{R}^2$, $\alpha_i = [\alpha_{i_1}, \alpha_{i_2}]^\top \in \mathbb{R}^2$, $i \in \{1, 2, 3\}$. The state of the object is $x_O = [p_O^\top, \eta_O]^\top \in \mathbb{R}^4$ and it is calculated though the states of the agents. The manipulators become singular when $\sin(\alpha_{i_1}) = 0\}, i \in \{1, 2\}$, thus the state constraints for the manipulators are set to: $\varepsilon < \alpha_{1_1} < \frac{\pi}{2} - \varepsilon$, $-\frac{\pi}{2} + \varepsilon < \alpha_{1_2} < \frac{\pi}{2} - \varepsilon$, $-\frac{\pi}{2} + \varepsilon < \alpha_{2_1} < -\varepsilon$, $-\frac{\pi}{2} + \varepsilon < \alpha_{2_2} < \frac{\pi}{2} - \varepsilon$. We also consider the input constraints: $-8.5 \leq u_{i,j}(t) \leq 8.5$, $i \in \{1, 2\}$, $j \in \{1, \ldots, 4\}$. The initial conditions of agents and the object are set to: $q_1(0) = [0.5, 0, \frac{\pi}{4}, \frac{\pi}{4}]^\top$, $q_2(0) = [0, -4.4142, -\frac{\pi}{4}, -\frac{\pi}{4}]^\top$, $q_3(0) = [-0.50, -4.4142, -\frac{\pi}{4}, -\frac{\pi}{4}]^\top$, $\dot{q}_1(0) = \dot{q}_2(0) = \dot{q}_3(0) = [0, 0, 0, 0]^\top$ and $x_O(0) = [0, -2.2071, 0.9071, \frac{\pi}{2}]^\top$. The desired goal state the object is set to $x_{O,\mathrm{des}} = [5, -2.2071, 0.9071, \frac{\pi}{2}]^\top$, which, due to the structure of the considered robots, corresponding uniquely to $q_{1,\mathrm{des}} = [5.5, 0, \frac{\pi}{4}, \frac{\pi}{4}]^\top$, $q_{2,\mathrm{des}} = [5, -4.4142, -\frac{\pi}{4}, -\frac{\pi}{4}]^\top$, $q_{3,\mathrm{des}} = [4.5, 0, -\frac{\pi}{4}, -\frac{\pi}{4}]^\top$, $\dot{q}_{3,\mathrm{des}} = [0, 0, 0, 0]^\top$ and $\dot{q}_{1,\mathrm{des}} = \dot{q}_{2,\mathrm{des}} = \dot{q}_{3,\mathrm{des}} = [0, 0, 0, 0]^\top$. We set an obstacle between the initial and the desired pose of the object. The obstacle is spherical with center $[2.5, -2.2071, 1]$ and radius $\sqrt{0.2}$. The sampling time is $h = 0.1$ seconds, the horizon is $T_p = 0.5$ seconds, and the total simulation time is 60 seconds; The matrices P_i, Q_i, R_i are set to: $P_i = Q_i = 0.5 I_{8 \times 8}$, $R_i = 0.5 I_{4 \times 4}$, $\forall i \in \{1, 2, 3\}$, and the load sharing coefficients as $c_1 = 0.3$, $c_2 = 0.5$, and $c_3 = 0.2$. The simulation results are depicted in Fig. 5.16-Fig. 5.23; Fig. 5.16, Fig. 5.17 and Fig. 5.18 show the error states of agent 1, 2 and 3, respectively, which converge to 0; Fig. 5.19 depicts the states of the objects; Fig. 5.23 shows the collision-avoidance constraint with the obstacle; Fig. 5.20 - Fig. 5.22 depict the control inputs of the three agents. Note that the different load-sharing coefficients produce slightly different inputs. The simulation was carried out by using the NMPC toolbox given in [119] and it took 13450 sec in MATLAB Environment on a desktop computer with 8 cores, 3.60 GHz CPU and 16GB of RAM. Note the significant time difference with respect to the centralized case of the previous section.

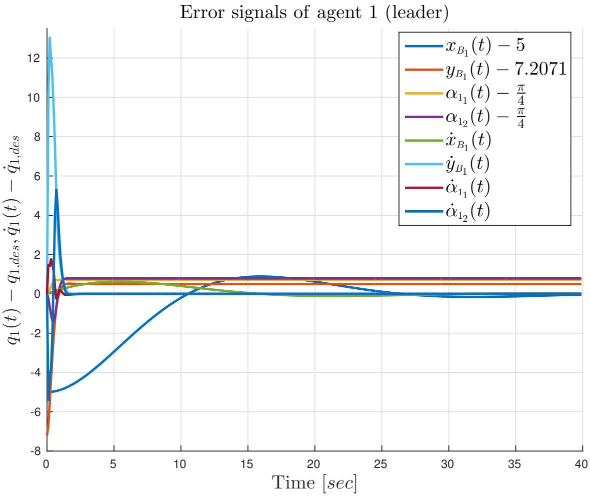


Figure 5.16: The error states of agent 1.

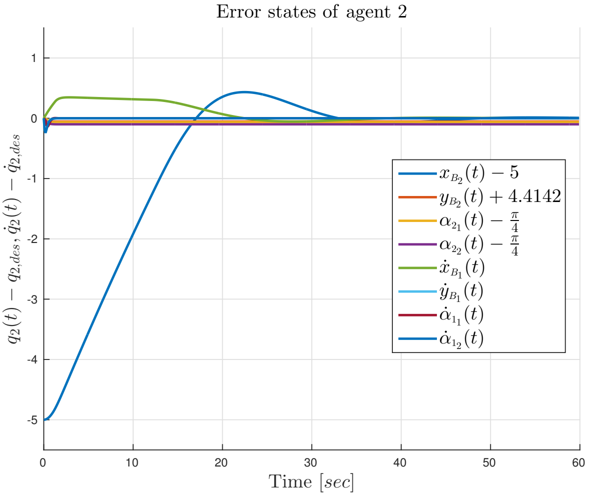


Figure 5.17: The error states of agent 2.

5.4 Conclusions and Future Work

In this work we proposed two NMPC schemes for the cooperative transportation of an object rigidly grasped by N robotic agents. The proposed control scheme deals with singularities of the agents, inter-agent collision avoidance as well as collision avoidance between the agents and the object with the workspace obstacles. We proved the feasibility and convergence analysis of the proposed methodology and simulation results verified the efficiency of the approach. Future efforts will be

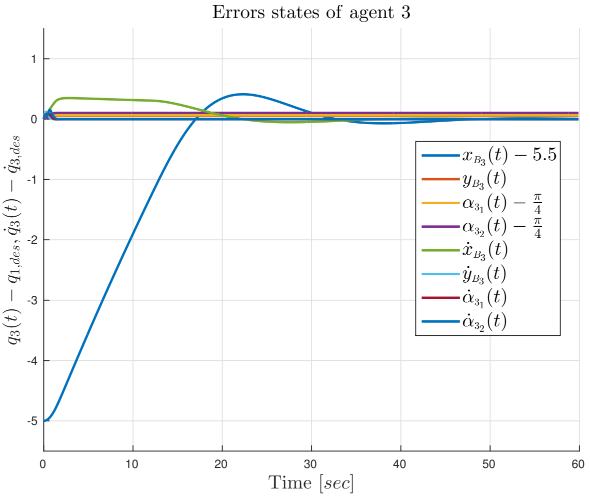


Figure 5.18: The error states of agent 3.

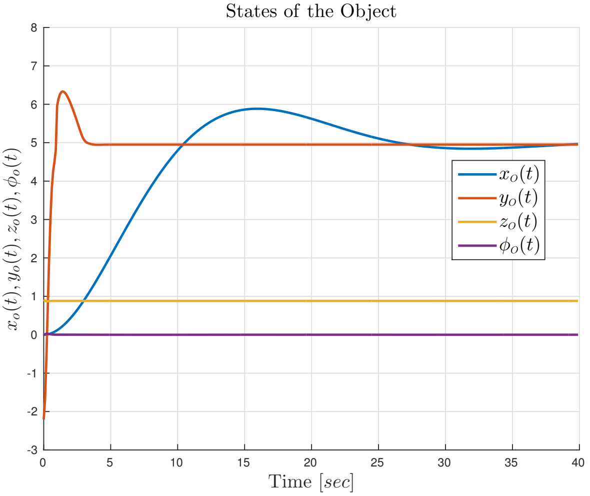


Figure 5.19: The states of object converging to the desired ones.

devoted towards reconfiguration in case of task infeasibility for the followers, event-triggered communication between the agents so as to reduce the communication burden that is required for solving the FHOCP at every sampling time, and real-time experiments.

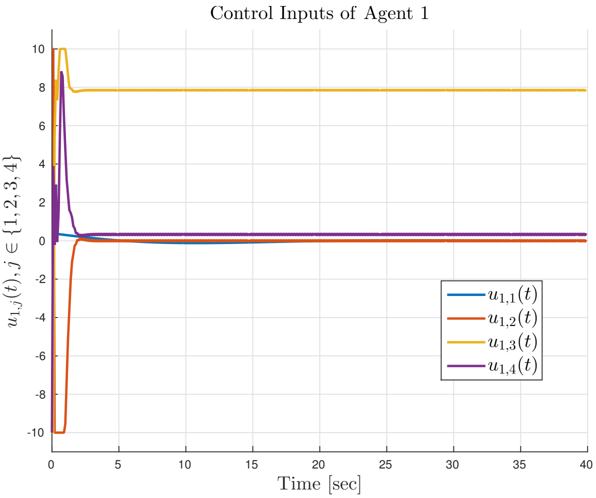


Figure 5.20: The control inputs of agent 1 with $-8.5 \leq u_{1,j}(t) \leq 8.5$.

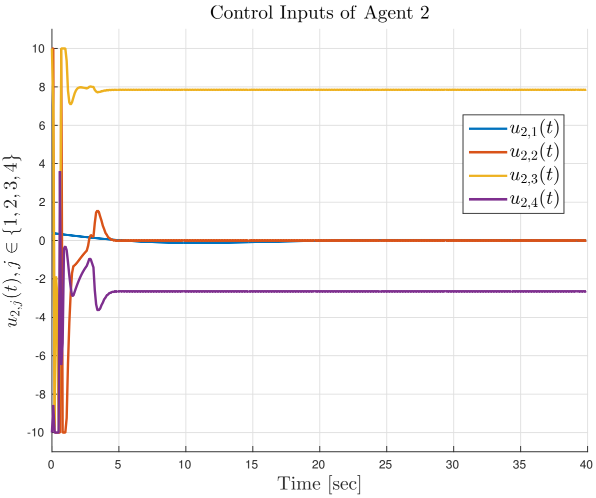


Figure 5.21: The control inputs of agent 2 with $-8.5 \leq u_{2,j}(t) \leq 8.5$.

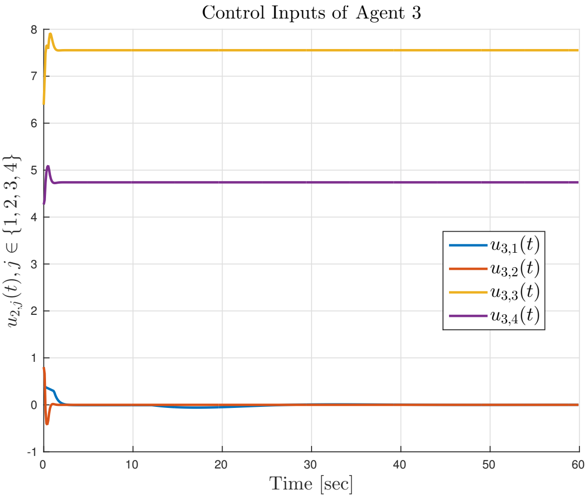


Figure 5.22: The control inputs of agent 3 with $-8.5 \leq u_{3,j}(t) \leq 8.5$.

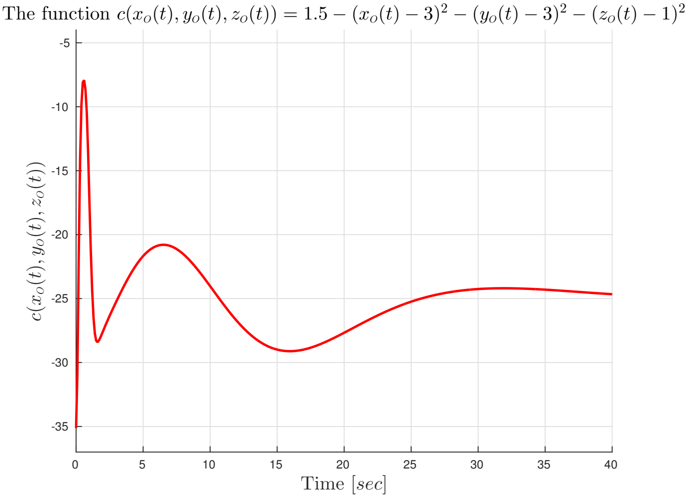


Figure 5.23: The function $c(x_O(t), y_O(t), z_O(t)) = 0.2 - (x_O - 2.5)^2 - (y_O + 2.2071)^2 - (z_O - 1)^2$ is always negative, indicating collision avoidance.

Chapter 6

Abstractions for Multi-Agent Manipulator-Endowed Systems

This chapter addresses the motion planning problem for a team of manipulator-endowed robotic agents under high level goals. We propose a hybrid control strategy that guarantees the accomplishment of each agent's local goal specification, which is given as a temporal logic formula, while guaranteeing inter-agent collision avoidance and connectivity maintenance. The overall approach is based on abstraction of the continuous systems to discrete transition systems, which we accomplish by designing suitable decentralized continuous controllers based on previous work on navigation functions. Next, given specific high-level tasks encoded by temporal logic formulas, we employ standard formal verification techniques and we derive high-level control algorithms that satisfy the agents' specifications. Simulation and experimental results verify the validity of the proposed methods.

6.1 Introduction

The importance of using multi-agent systems, highlighted in the previous chapters, is evident when the multi-agent system consists of manipulator-endowed agents, such as, e.g., mobile/aerial manipulators or unmanned aerial vehicles (UAVs). These type of systems are widely used for inspection, surveillance as well as manipulation tasks, such as pick-and-place tasks or object transportation.

In the case of aerial vehicles, the multi-agent problems are well studied in the related literature. The standard problem of formation control for a team of aerial vehicles is addressed in [125–130], whereas [131–135] consider leader-follower formation approaches, where the latter also treats the problem of collision avoidance with static obstacles in the environment; [136], [137] and [138] employ dynamic programming, Model Predictive Control and reachable set algorithms, respectively, for inter-agent collision avoidance. In [139] the cooperative evader pursuit problem is treated. Regarding the related literature that concerns cooperative robotic manipulation tasks, we refer the reader to the previous chapters.

Ultimately, we would like the robotic agents to execute more complex high-level tasks, involving combinations of safety (”never enter a dangerous regions”), surveillance (”keep visiting regions A and B infinitely often”) or sequencing (”collect data in region C and upload it in region D”) properties. Temporal logic languages offer a means to express the aforementioned specifications, since they can describe complex planning objectives in a more efficient way than the well-studied navigation algorithms. A recent direction in the multi-agent control and robotics field is the use of temporal logic languages for motion and/or action planning, since they provide a fully-automated correct-by-design controller synthesis approach for autonomous robots. Temporal logics, such as linear temporal logic (LTL), computation tree logic (CTL) or metric-interval temporal logic (MITL), provide formal high-level languages that can describe planning objectives more complex than the usual navigation techniques. The task specification is given as a temporal logic formula with respect to a discretized abstraction of the robot motion modeled as a finite transition system, and then, a high-level discrete plan is found by off-the-shelf model-checking algorithms, given the finite transition system and the task specification [38].

There exists a wide variety of works that employ temporal logic languages for single- and multi-agent systems, e.g., [140–156]. Regarding aerial vehicles, [157] addresses the vehicle routing problem using MTL specifications and [158] approaches the LTL motion planning using MILP optimization techniques, both in a centralized manner. Markov Decision Processes are used for the LTL planning in [159]. The aforementioned works, however, consider discrete agent models and do not take into account their continuous dynamics. The discretization of a multi-agent system to an abstracted finite transition system necessitates the design of appropriate continuous-time controllers for the transition of the agents among the states of the transition system [38]. Most works in the related literature, however, including the aforementioned ones, either assume that there *exist* such continuous controllers or adopt single- and double-integrator models, ignoring the actual dynamics of the agents. Discretized abstractions, including design of the discrete state space and/or continuous-time controllers, have been considered in [160–164] for general systems and [165, 166] for multi-agent systems.

Another drawback of the majority of works in the related literature of temporal logic-based motion planning is the point-agent assumption (as, e.g. in [143, 147, 148]), which does not take into account potential collisions between the robotic agents. The latter is a crucial safety property in real-time scenarios, where actual vehicles are used in the motion planning framework.

The contribution of this chapter is the design of well-defined abstractions for a multi-agent manipulator-endowed system.

1. Firstly, we propose a novel decentralized control protocol for the motion planning of a team of aerial vehicles under LTL specifications with simultaneous inter-agent collision avoidance. In particular, we extend previous work on decentralized navigation functions [37] to abstract the motion of each agent as a finite transition system. Then, we employ standard formal-verification

techniques to derive plans that satisfy each agent's LTL specification. The proposed control protocol is decentralized in the sense that each agent has limited sensing information and derives and executes its desired path without communicating with the other agents or knowing their respective goals/specifications. Simulation and experimental results with quadrotors verify the effectiveness of the proposed framework.

2. Secondly, we design robust continuous-time controllers for the navigation of a team of 2nd order mobile manipulators among predefined regions of interest. The proposed methodology is decentralized, since each agent uses only local information based on limited sensing capabilities. Moreover, we guarantee (i) inter-agent collision avoidance by introducing a novel transformation-free ellipsoid-based strategy, (ii) connectivity maintenance for a subset of the initially connected agents, which might be important for potential cooperative tasks, and (iii) kinematic singularity avoidance of the robotic agents.

The rest of the chapter consists of two main parts: Firstly, Section 6.3 is devoted to the abstraction derivation of a team of aerial vehicles, with Sections 6.3.1, 6.3.2, 6.3.3, and 6.3.4 defining and solving the problem in hand, and providing simulation and experimental results, respectively. Similarly, Section 6.4 presents the decentralized abstraction for the team of mobile manipulators with Sections 6.4.1, 6.4.2, and 6.4.3 providing the corresponding problem formulation, solution and simulation results, respectively. Finally, Section 6.5 concludes the chapter.

6.2 Preliminaries

6.2.1 Cubic Equations and Ellipsoid Collision

Proposition 6.1. Consider the cubic equation $f(\lambda) = c_3\lambda^3 + c_2\lambda^2 + c_1\lambda + c_0 = 0$ with $c_\ell \in \mathbb{R}, \forall \ell \in \{0, \dots, 3\}$ and roots $(\lambda_1, \lambda_2, \lambda_3) \in \mathbb{C}^3$, with $f(\lambda_1) = f(\lambda_2) = f(\lambda_3) = 0$. Then, given its discriminant $\Delta = (c_3)^4 \prod_{\substack{i \in \{1,2\} \\ j \in \{i+1,\dots,3\}}} (\lambda_i - \lambda_j)^2$, the following hold:

(i) $\Delta = 0 \Leftrightarrow \exists i, j \in \{1, 2, 3\}$, with $i \neq j$, such that $\lambda_i = \lambda_j$, i.e., at least two roots are equal,

(ii) $\Delta > 0 \Leftrightarrow \lambda_i \in \mathbb{R}, \forall i \in \{1, 2, 3\}$, and $\lambda_i \neq \lambda_j, \forall i, j \in \{1, 2, 3\}$, with $i \neq j$, i.e., all roots are real and distinct.

Proposition 6.2. [167] Consider two planar ellipsoids $\mathcal{A} = \{z \in \mathbb{R}^3 \text{ s.t. } z^\top A(t) z \leq 0\}$, $\mathcal{B} = \{z \in \mathbb{R}^3 \text{ s.t. } z^\top B(t) z \leq 0\}$, with $z = [p^\top 1]^\top$ being the homogeneous coordinates of $p \in \mathbb{R}^2$, and $A, B : \mathbb{R}_{\geq 0} \to \mathbb{R}^{3\times 3}$ terms that describe their motion in 2D space. Given their characteristic polynomial $f : \mathbb{R} \to \mathbb{R}$ with $f(\lambda) = \det(\lambda A - B)$, which has degree 3, the following hold:

(i) $\exists \lambda^* \in \mathbb{R}_{>0}$ s.t. $f(\lambda^*) = 0$, i.e, the polynomial $f(\lambda)$ always has one positive real root,

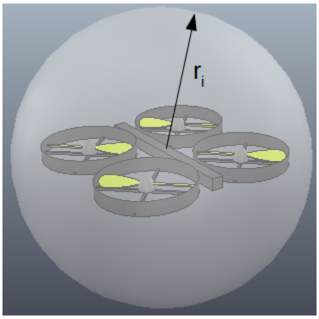


Figure 6.1: Bounding sphere of an aerial vehicle.

(ii) $\mathcal{A} \cap \mathcal{B} = \emptyset \Leftrightarrow \exists \lambda_1^*, \lambda_2^* \in \mathbb{R}_{<0}$, with $\lambda_1^* \neq \lambda_2^*$, and $f(\lambda_1^*) = f(\lambda_2^*) = 0$, i.e., $\mathcal{A}$ and $\mathcal{B}$ are disjoint if and only if the characteristic equation $f(\lambda) = 0$ has two distinct negative roots.

(iii) $\mathcal{A} \cap \mathcal{B} \neq \emptyset$ and $\mathring{\mathcal{A}} \cap \mathring{\mathcal{B}} = \emptyset \Leftrightarrow \exists \lambda_1^*, \lambda_2^* \in \mathbb{R}_{<0}$, with $\lambda_1^* = \lambda_2^*$, and $f(\lambda_1^*) = f(\lambda_2^*) = 0$, i.e., $\mathcal{A}$ and $\mathcal{B}$ touch externally if and only if the characteristic equation $f(\lambda) = 0$ has a negative double root.

6.3 Decentralized Motion Planning with Collision Avoidance for a Team of UAVs under High Level Goals

6.3.1 System and Problem Formulation

Consider N aerial agents operating in a static workspace that is bounded by a large sphere in 3-D space $\mathcal{W} = \mathcal{B}_{r_0}(p_0) = \{p \in \mathbb{R}^3 \text{ s.t. } \|p - p_0\| \leq r_0\}$, where $p_0 \in \mathbb{R}^3$ and $r_0 > 0$ are the center and radius of $\mathcal{W}$. Within $\mathcal{W}$ there exist K smaller spheres around points of interest, which are described by $\pi_k = \mathcal{B}_{r_{\pi_k}}(p_{\pi_k}) = \{p \in \mathbb{R}^3 \text{ s.t. } \|p - p_{\pi_k}\| \leq r_{\pi_k}\} \subset \mathcal{W}$, where $p_{\pi_k} \in \mathbb{R}^3, r_{\pi_k} > 0$ are the central point and radius, respectively, of π_k. We denote the set of all π_k as $\Pi = \{\pi_1, \ldots, \pi_K\}$. For the workspace partition to be valid, we consider that the regions of interest are sufficiently distant from each other and from the workspace boundary, i.e., $d_3(p_{\pi_k}, p_{\pi_{k'}}) > 4 \max_{k \in \{1,\ldots,K\}}(r_{\pi_k})$ and $d_3(p_{\pi_k}, p_0) < r_0 - 3r_{\pi_k}, \forall k, k' \in \{1, \ldots, K\}$ with $k \neq k'$. Moreover, we introduce a set of atomic propositions Ψ_i for each agent $i \in \{1, \ldots, N\}$ that indicates certain properties of interest of agent i in Π and are expressed as boolean variables. The properties satisfied at each region π_k are provided by the labeling function $\mathcal{L}_i : \Pi \to 2^{\Psi_i}$, which assigns to each region $\pi_k, k \in \{1, \ldots, K\}$ the subset of the atomic propositions Ψ_i that are true in that region.

System model

Each agent $i \in \{1, \dots, N\}$ occupies a bounding sphere: $\mathcal{B}_{r_i}(p_i(t)) = \{p(t) \in \mathcal{W}$ s.t. $\|p(t) - p_i(t)\| \leq r_i\}$, where $p_i : \mathbb{R}_{\geq 0} \to \mathbb{R}^3$ is the center and $r_i > 0$ the radius of the sphere (Fig. 6.1). We also consider that $r_i < r_{\pi_k}, \forall i \in \{1, \dots, N\}, k \in \{1, \dots, K\}$, i.e., the regions of interest are larger than the aerial vehicles. The motion of each agent is controlled via its centroid p_i through the single integrator dynamics:

$$\dot{p}_i = u_i, i \in \{1, \dots, N\}. \tag{6.1}$$

Moreover, we consider that agent i has a limited sensing range of $d_{s_i} > \max_{i,j=\{1,\dots,N\}} (r_i + r_j)$. Therefore, by defining the neighboring set $\mathcal{N}_i = \{j \in \{1, \dots, N\},$ s.t. $\|p_i - p_j\| \leq d_{s_i}\}$, agent i knows at each time instant the position of all $p_j, \forall j \in \mathcal{N}_i$ as well as its own position p_i. The workspace is perfectly known, i.e., p_{π_k}, r_{π_k} are known to all agents, for all $k \in \{1, \dots, K\}$.

With the above ingredients, we provide the following definitions:

Definition 6.1. An agent $i \in \{1, \dots, N\}$ is in a region $\pi_k, k \in \{1, \dots, K\}$ at a configuration p_i, denoted as $\mathcal{A}_i(p_i) \in \pi_k$, if and only if $\mathcal{B}_{r_i}(p_i) \subseteq \mathcal{B}_{r_{\pi_k}}(p_{\pi_k})$.

Definition 6.2. Assume that $\mathcal{A}_i(p_i(t_0)) \in \pi_k, i \in \{1, \dots, N\}, k \in \{1, \dots, K\}$ for some $t_0 \geq 0$. Then there exists a transition for agent i from region π_k to region $\pi_{k'}, k' \in \{1, \dots, K\}$, denoted as $\pi_k \to_i \pi_{k'}$, if and only if there exists a finite $t_f \geq 0$ and a bounded control trajectory u_i such that (i) $\mathcal{A}_i(p_i(t_f)) \in \pi_{k'}$, (ii) $\mathcal{B}_{r_i}(p_i(t)) \cap \mathcal{B}_{r_{\pi_m}}(p_{\pi_m}) = \emptyset$, and (iii) $\mathcal{B}_{r_i}(p_i(t)) \cap \mathcal{B}_{r_{i'}}(p_{i'}(t)) = \emptyset, \forall m \in \{1, \dots, K\}$ with $m \neq k, k', \forall i' \in \{1, \dots, N\}$ with $i' \neq i$ and $t \in [0, t_f]$.

Loosely speaking, an agent i can transit between two regions of interest π_k and $\pi_{k'}$, if there exists a bounded control trajectory u_i in (6.1) that takes agent i from π_k to $\pi_{k'}$ while avoiding entering all other regions and colliding with the other agents.

Specification

Our goal is to control the multi-agent system subject to (6.1) so that each agent's behavior obeys a given specification over its atomic propositions Ψ_i.

Given a trajectory $p_i(t)$ of agent i, its corresponding *behavior* is given by the infinite sequence $\beta_i = (p_i(t), \psi_i) = (p_{i_1}, \psi_{i_1})(p_{i_2}, \psi_{i_2}) \dots$, with $\psi_{i_m} \in 2^{\Psi_i}$ and $\mathcal{A}(p_{i_m}) \in \pi_{k_m}, \psi_{i_m} \in \mathcal{L}_i(\pi_{k_m}), k_m \in \{1, \dots, K\}, \forall m \in \mathbb{N}$.

Definition 6.3. The behavior $\beta_i = (p_i(t), \psi_i)$ satisfies an LTL formula ϕ if and only if $\psi_i \models \phi$.

Problem Formulation

The control objectives are given for each agent separately as LTL formulas ϕ_i over $\Psi_i, i \in \{1, \dots, N\}$. An LTL formula is satisfied if there exists a behavior $\beta_i = (p_i(t), \psi_i)$ of agent i that satisfies ϕ_i. Formally, the problem treated in this section is the following:

Problem 6.1. Given a set of aerial vehicles N subject to the dynamics (6.1) and N LTL formulas ϕ_i, over the respective atomic propositions $\Psi_i, i \in \{1, \ldots, N\}$, achieve behaviors β_i that (i) yield satisfaction of $\phi_i, \forall i \in \{1, \ldots, N\}$ and (ii) guarantee inter-agent collision avoidance.

6.3.2 Main Results

Continuous Control Design

The first ingredient of our solution is the development of a decentralized feedback control law that establishes a transition relation $\pi_k \to_i \pi_{k'}, \forall k, k' \in \{1, \ldots, K\}$ according to Def. 6.2. Our approach is based on the concept of *Decentralized Navigation Functions*, introduced in [37], for which an overview can be found in Section 2.3. More specifically, given that $\mathcal{A}_i(p_i(t_0))$ for some $t_0 \geq 0$, we propose a decentralized control law u_i for the transition $\pi_k \to_i \pi_{k'}$, as defined in Def. 6.2.

Initially, we define the set of "undesired" regions as $\Pi_{k,k'} = \{\pi_m \in \Pi, m \in \{1, \ldots, K\} \backslash \{k, k'\}\}$ and the corresponding free space $\mathcal{F}_{k,k'} = \mathcal{W} \backslash \{\mathcal{B}_{r_\pi}(p_\pi)\}_{\pi \in \Pi_{k,k'}}$. As the goal configuration we consider the centroid $p_{\pi_{k'}}$ of $\pi_{k'}$ and we construct the function $\gamma_{i_{k'}} : \mathcal{F}_{k,k'} \to \mathbb{R}_{\geq 0}$ with $\gamma_{i_{k'}}(p_i) = \|p_i - p_{\pi_{k'}}\|^2$. For the collision avoidance between the agents, we employ the function $G_i : \mathcal{F}_{k,k'} \times \mathbb{R}^{3(N-1)} \to \mathbb{R}$ as defined in [37].

Moreover, we also need some extra terms that guarantee that agent i will avoid the rest of the regions as well as the workspace boundary. To this end, we construct the function $\alpha_{i_{k,k'}} : \mathcal{F}_{k,k'} \to \mathbb{R}$ with $\alpha_{i_{k,k'}}(p_i) = \alpha_{i,0}(p_i) \prod_{m \in \Pi_{k,k'}} \alpha_{i,m}(p_i)$, where the function $\alpha_{i,0} : \mathcal{F}_{k,k'} \to \mathbb{R}$ is a measure of the distance of agent i from the workspace boundary $\alpha_{i,0} = (r_0 - r_i)^2 - \|p_i - p_0\|^2$ and the function $\alpha_{i,m} : \mathcal{F}_{k,k'} \to \mathbb{R}$ is a measure of the distance of agent i from the undesired regions $\alpha_{i,m} = \|p_i - p_m\|^2 - (r_i + r_m)^2$.

With the above ingredients, we construct the following navigation function $\varphi_{i_{k,k'}} : \mathcal{F}_{k,k'} \times \mathbb{R}^{3(N-1)} \to [0, 1]$:

$$\varphi_{i_{k,k'}}(p(t)) = \frac{\gamma_{i_{k'}}(p_i) + f_i(G_i)}{(\gamma_{i_{k'}}^{\lambda_i}(p_i) + G_i(p)\alpha_{i_{k,k'}}(p_i))^{1/\lambda_i}} \tag{6.2}$$

for agent i, with $\lambda_i > 0$ and the following vector field:

$$c_{i_{k,k'}}(t) = \begin{cases} -k_{g_i} \dfrac{\partial \varphi_{i_{k,k'}}(p(t))}{\partial p_i(t)}, & \text{if } \pi_k \not\equiv \pi_{k'} \\ 0 & \text{if } \pi_k \equiv \pi_{k'} \end{cases} \tag{6.3}$$

for all $t \geq t_0$, with $k_{g_i} > 0$ and $f_i(G_i)$ as defined in [37].

The navigation field (6.3) guarantees that agent i will not enter the undesired regions or collide with the other agents and $\lim_{t \to \infty} p_i(t) = p_{\pi_{k'}}$. The latter property of asymptotic convergence along with the assumption that $r_i < r_{\pi_k}, \forall i \in \{1, \ldots, N\}, k \in \{1, \ldots, K\}$, implies that there exists a finite time instant $t^f_{i,k'} \geq t_0$

such that $p_i(t^f_{i,k'}) \in \mathcal{B}_{r_{\pi_{k'}}}(p_{\pi_{k'}})$ and more specifically that $\mathcal{A}_i(p_i(t^f_{i,k'})) \in \pi_{k'}$, which is the desired behavior. The time instant $t^f_{i,k'}$ can be chosen from the set $S_{t_{k'}} = \{t \geq t_0, \mathcal{A}_i(p_i(t)) \in \pi_{k'}\}$.

Note, however, that once agent i leaves region π_k, there is no guarantee that it will not enter that region again (note that $F_{k,k'}$ includes π_k). Therefore, we define the set $\Pi_{\emptyset,k'} = \{\pi_m \in \Pi, m \in \{1,\ldots,K\}\backslash\{k'\}\}$ and the corresponding free space $\mathcal{F}_{\emptyset,k'} = \mathcal{W}\backslash\{\mathcal{B}_{r_\pi}(p_\pi)\}_{\pi\in\Pi_{\emptyset,k'}}$, and we construct the function $\varphi_{i_{\emptyset,k'}} : \mathcal{F}_{\emptyset,k'} \times \mathbb{R}^{3(N-1)} \to [0,1]$:

$$\varphi_{i_{\emptyset,k'}}(p(t)) = \frac{\gamma_{i_{k'}}(p_i) + f_i(G_i)}{(\gamma^{\lambda_i}_{i_{k'}}(p_i) + G_i(p)\alpha_{i_{\emptyset,k'}}(p_i))^{1/\lambda_i}} \tag{6.4}$$

where $\alpha_{i_{\emptyset,k'}} = \alpha_{i,0}(p_i)\prod_{m\in\Pi_{\emptyset,k'}} \alpha_{i,m}(p_i)$, with corresponding vector field:

$$c_{i_{\emptyset,k'}}(t) = -k_{g_i}\frac{\partial\varphi_{i_{\emptyset,k'}}(p(t))}{\partial p_i(t)}, \tag{6.5}$$

which guarantees that region π_k will be also avoided. Therefore, we develop a switching control protocol that employs (6.3) until agent i is out of region π_k and then switches to (6.5) until $t = t^f_{i,k'}$. Consider the following switching function:

$$s(x) = \frac{1}{2}(\text{sat}(2x-1)+1) \tag{6.6}$$

where sat : $\mathbb{R} \to [-1,1]$ is the standard saturation function ($\text{sat}(x) = x$, if $|x| \leq 1; \text{sat}(x) = x/|x|$, if $|x| > 1$), and the time instant $t'_{i,k}$ that represents the moment that agent i is out of region π_k, i.e., $t'_{i,k} = \min S_{t_{\not k}}$, where $S_{t_{\not k}} = \{t \geq t_0, \mathcal{B}_{r_i}(p_i(t)) \cap \mathcal{B}_{r_{\pi_k}}(p_{\pi_k}) = \emptyset\}$. Note that $t'_{i,k} < t^f_{i,k'}$, since $d_3(p_{\pi_k}, p_{\pi_{k'}}) > 4\max_{k\in\{1,\ldots,K\}}(r_{\pi_k}), \forall k, k' \in \{1,\ldots,K\}$ with $k \neq k'$. Then, we propose the following switching control protocol $u_i : [t_0, t^f_{i,k'}) \to \mathbb{R}^3$:

$$u_i(t) = \begin{cases} c_{i_{k,k'}}(t), & t \in T_1 \\ (1 - s(\xi_{i,k}))c_{i_{k,k'}}(t) + s(\xi_{i,k})c_{i_{\emptyset,k'}}(t), & t \in T_2 \end{cases} \tag{6.7}$$

where $T_1 = [t_0, t'_{i,k}), T_2 = [t'_{i,k}, t^f_{i,k'})$ and $\xi_{i,k} = \dfrac{t - t'_{i,k}}{\nu_i}$, where ν_i is a design parameter indicating the time period of the switching process, with $t^f_{i,k'} - t'_{i,k} > \nu_i > 0$. Invoking the continuity of $p_i(t)$, we obtain $\lim_{t\to(t^f_{i,k'})^-} p_i(t) = p_i(t^f_{i,k'}) \in \mathcal{B}_{r_{\pi_{k'}}}(p_{\pi_{k'}})$ and hence the control protocol (6.7) guarantees, for sufficiently small ν_i, that agent i will navigate from π_k to $\pi_{k'}$ in finite time without entering any other regions or colliding with other agents and therefore establishes a transition $\pi_k \to_i \pi_{k'}$. The proof of correctness of (6.2) and (6.4) follows closely the one in [37] and is therefore omitted.

High-Level Plan Generation

The next step of our solution is the high-level plan, which can be generated using standard techniques inspired by automata-based formal verification methodologies. In Section 6.3.2, we proposed a continuous control law that allows the agents to transit between any $\pi_k, \pi_{k'} \in \Pi$ in the given workspace $\mathcal{W}$, without colliding with each other. Thanks to this and to our definition of LTL semantics over the sequence of atomic propositions, we can abstract the motion capabilities of each agent as a finite transition system $\mathcal{T}_i$ as follows [38]:

Definition 6.4. The motion of each agent $i \in \{1, \dots, N\}$ in $\mathcal{W}$ is modeled by the following Transition System (TS):

$$\mathcal{T}_i = (\Pi_i, \Pi_i^{\text{init}}, \rightarrow_i, \Psi_i, \mathcal{L}_i), \tag{6.8}$$

where $\Pi_i \subseteq \Pi$ is the set of states represented by the regions of interest that the agent can be at, according to Def. 6.1, $\Pi_i^{\text{init}} \subseteq \Pi_i$ is the set of initial states that agent i can start from, $\rightarrow_i \subseteq \Pi_i \times \Pi_i$ is the transition relation established in Section 6.3.2, abbreviated as $\pi_k \rightarrow \pi_{k'}, \pi_k, \pi_{k'} \in \Pi_i$, and $\Psi_i, \mathcal{L}_i$ are the atomic propositions and labeling function respectively, as defined in Section 6.3.1.

After the definition of $\mathcal{T}_i$, we translate each given LTL formula $\phi_i, i \in \{1, \dots, N\}$ into a Büchi automaton $\mathcal{C}_i$ and we form the product $\widetilde{\mathcal{T}_i} = \mathcal{T}_i \times \mathcal{C}_i$. The accepting runs of $\widetilde{\mathcal{T}_i}$ satisfy ϕ_i and are directly projected to a sequence of waypoints to be visited, providing therefore a desired path for agent i. Although the semantics of LTL is defined over infinite sequences of atomic propositions, it can be proven that there always exists a high-level plan that takes a form of a finite state sequence followed by an infinite repetition of another finite state sequence. For more details on the followed technique, we kindly refer the reader to the related literature, e.g., [38].

Following the aforementioned methodology, we obtain a high-level plan for each agent as sequences of regions and atomic propositions $p_i = \pi_{i_1}\pi_{i_2}\dots$ and $\psi_i = \psi_{i_1}\psi_{i_2}\dots$ with $i_m \in \{1, \dots, K\}, \psi_{i_m} \in 2^{\Psi_i}, \psi_{i_m} \in \mathcal{L}_i(\pi_{i_m}), \forall m \in \mathbb{N}$ and $\psi_i \models \phi_i, \forall i \in \{1, \dots, N\}$.

The execution of (p_i, ψ_i) produces a trajectory $p_i(t)$ that corresponds to the behavior $\beta_i = (p_i(t), \psi_i) = (p_{i_1}(t), \psi_{i_1})(p_{i_2}(t), \psi_{i_2})\dots$, with $\mathcal{A}_i(p_{i_m}) \in \pi_{i_m}$ and $\psi_{i_m} \in \mathcal{L}_i(\pi_{i_m}), \forall m \in \mathbb{N}$. Therefore, since $\psi_i \models \phi_i$, the behavior β_i yields satisfaction of the formula ϕ_i. Moreover, the property of inter-agent collision avoidance is inherent in the transition relations of $\mathcal{T}_i$ and guaranteed by the navigation control algorithm of Section 6.3.2. The previous discussion is summarized in the following theorem:

Theorem 6.1. *The individual executions of $(p_i, \psi_i), i \in \{1, \dots, N\}$, that satisfy the respective ϕ_i, produce agent behaviors $\beta_i, i \in \{1, \dots, N\}$ that (i) yield the satisfaction of all $\phi_i, i \in \{1, \dots, N\}$ and (ii) guarantee inter-agent collision avoidance, providing, therefore, a solution to Problem 6.1.*

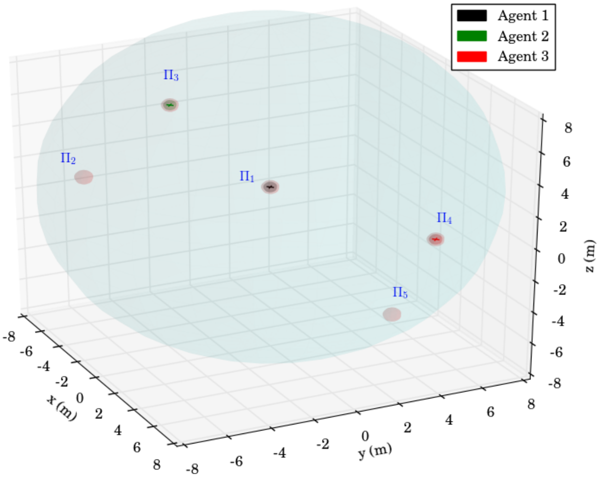


Figure 6.2: Initial workspace of the simulation studies. The grey spheres represent the regions of interest while the black, green and red crosses represent agents 1,2 and 3, respectively, along with their bounding spheres.

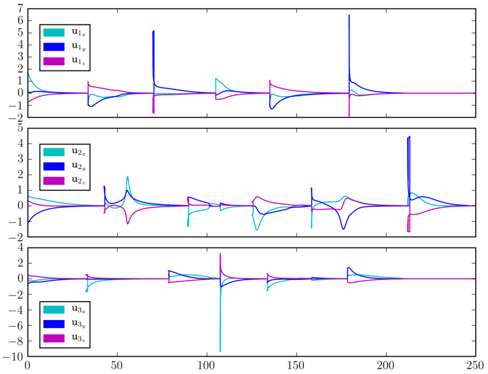
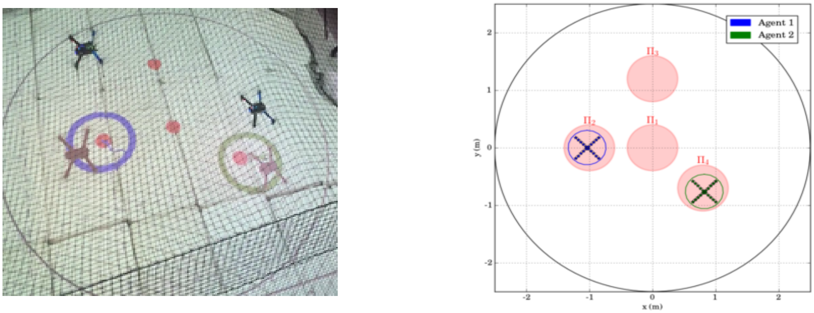


Figure 6.3: The resulting 3-dimensional control signals of the 3 agents for the simulation studies. Top: agent 1, middle: agent 2, bottom: agent 3.

Remark 6.1. The proposed control algorithm is decentralized in the sense that each agent derives and executes its own plan without communicating with the rest of the team. The only information that each agent has is the position of its neighboring agents that lie in its limited sensing radius.

6.3.3 Simulation Results

To demonstrate the efficiency of the proposed algorithm, we consider $N = 3$ aerial vehicles $\mathcal{B}_{r_i}(p_i(t))$, with $r_i = 0.3$m, $d_{s_i} = 0.65$m, $\forall i = \{1,2,3\}$, operating in a workspace $\mathcal{W} = \mathcal{B}_{r_0}(p_0)$ with $r_0 = 10$m and $p_0 = [0,0,0]^T$m. Moreover, we consider



(a) **(b)**

Figure 6.4: Initial workspace for the first real experimental scenario. (a): The UAVs with the projection of their bounding spheres, (with blue and green), and the centroids of the regions of interest (with red). (b): Top view of the described workspace. The UAVs are represented by the blue and green circled X's and the regions of interest by the red disks $\pi_1, \dots, \pi_4$.

$K = 5$ spherical regions of interest $\mathcal{B}_{r_{\pi_k}}(p_{\pi_k})$ with $r_{\pi_k} = 0.4$m, $\forall k = \{1, \dots, 5\}$ and $p_{\pi_1} = [0, 0, 2]^T$m, $p_{\pi_2} = [1, -9, 5]^T$m, $p_{\pi_3} = [-8, -1, 4]^T$m, $p_{\pi_4} = [2, 7, -2]^T$m and $p_{\pi_5} = [7.5, 2, -3]^T$m. The initial configurations of the agents are taken as $p_1(0) = p_{\pi_1}, p_2(0) = p_{\pi_3}, p_3(0) = p_{\pi_4}$ and therefore, $\mathcal{A}_1(p_1(0)) \in \pi_1, \mathcal{A}_2(p_2(0)) \in \pi_3$ and $\mathcal{A}_3(p_3(0)) \in \pi_4$. An illustration of the described workspace is depicted in Fig. 6.2.

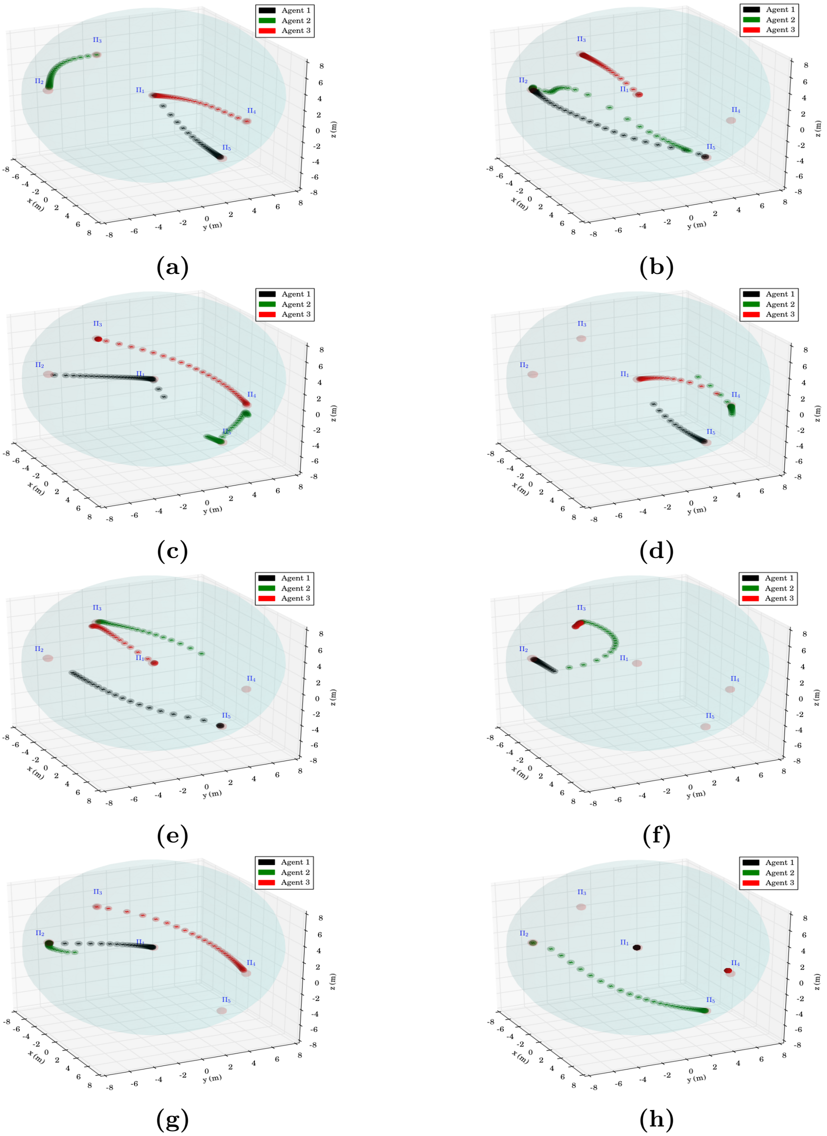
We consider that agent 2 is assigned with inspection tasks and has the atomic propositions $\Psi_2 = \{\text{“ins}_\text{a}\text{”}, \text{“ins}_\text{b}\text{”}, \text{“ins}_\text{c}\text{”}, \text{“ins}_\text{d}\text{”}, \text{“obs”}\}$ with $\mathcal{L}_2(\pi_1) = \{\text{“obs”}\}$, $\mathcal{L}_2(\pi_2) = \{\text{“ins}_\text{a}\text{”}\}, \mathcal{L}_2(\pi_3) = \{\text{“ins}_\text{b}\text{”}\}, \mathcal{L}_2(\pi_4) = \{\text{“ins}_\text{c}\text{”}\}$ and $\mathcal{L}_2(\pi_5) = \{\text{“ins}_\text{d}\text{”}\}$, where we have considered that region π_1 is an undesired (”obstacle”) region for this agent. More specifically, the task for agent 2 is the continuous inspection of the workspace while avoiding region π_1. The corresponding LTL specification is $\phi_2 = (\Box\neg\text{“obs”}) \wedge \Box(\Diamond\text{“ins}_\text{a}\text{”} \wedge \Diamond\text{“ins}_\text{b}\text{”} \wedge \Diamond\text{“ins}_\text{c}\text{”} \wedge \Diamond\text{“ins}_\text{d}\text{”})$. Agents 1 and 3 are interested in moving around resources scattered in the workspace and have propositions $\Psi_1 = \Psi_3 = \{\text{“res}_\text{a}\text{”}, \text{“res}_\text{b}\text{”}, \text{“res}_\text{c}\text{”}, \text{“res}_\text{d}\text{”}, \text{“res}_\text{e}\text{”}\}$ with $\mathcal{L}_1(\pi_1) = \mathcal{L}_3(\pi_1) = \{\text{res}_\text{a}\}, \mathcal{L}_1(\pi_2) = \mathcal{L}_3(\pi_2) = \{\text{res}_\text{b}\}, \mathcal{L}_1(\pi_3) = \mathcal{L}_3(\pi_3) = \{\text{res}_\text{c}\}, \mathcal{L}_1(\pi_4) = \mathcal{L}_3(\pi_4) = \{\text{res}_\text{d}\}$ and $\mathcal{L}_1(\pi_5) = \mathcal{L}_3(\pi_5) = \{\text{res}_\text{e}\}$. We assume that “$\text{res}_\text{a}$” is shared between the two agents whereas “res_b” and “res_e” have to be accessed only by agent 1 and “res_c” and “res_d” only by agent 3. The corresponding specifications are $\phi_1 = \Box\neg(\text{“res}_\text{c}\text{”} \vee \text{“res}_\text{d}\text{”}) \wedge \Box\Diamond(\text{“res}_\text{a}\text{”} \bigcirc \text{“res}_\text{e}\text{”} \bigcirc \text{“res}_\text{b}\text{”})$ and $\phi_3 = \Box\neg(\text{“res}_\text{b}\text{”} \vee \text{“res}_\text{e}\text{”}) \wedge \Box\Diamond(\text{“res}_\text{a}\text{”} \bigcirc \text{“res}_\text{c}\text{”} \bigcirc \text{“res}_\text{d}\text{”})$, where we have also included a specific order for the access of the resources. Next, we employ the off-the-shelf tool LTL2BA [168] to create the Büchi automata $\mathcal{C}_i, i = \{1, 2, 3\}$ and by following the procedure described in Section 6.3.2, we derive the paths $p_1 = (\pi_1\pi_5\pi_2)^\omega, p_2 = (\pi_3\pi_2\pi_5\pi_4)^\omega, p_3 = (\pi_4\pi_1\pi_3)^\omega$, whose execution satisfies ϕ_1, ϕ_2, ϕ_3. Regarding the continuous control protocol, we chose $k_{g_i} = 15, \lambda_i = 5, \forall i \in \{1, 2, 3\}$ in (6.3), (6.5) and the switching duration in (6.7) was calculated online as $\nu_i = 0.1t'_{i,k}$, where we assume that the large distance between the regions π_k (see Fig. 6.2) implies that $t^f_{i,k'} > 1.1t'_{i,k}$ and thus, $\nu_i < t^f_{i,k'} - t'_{i,k}$. The simulation results are depicted in Fig. 6.3 and 6.5. In particular, Fig. 6.5 illustrates the execution of the paths $(\pi_1\pi_5\pi_2)^2\pi_1, (\pi_3\pi_2\pi_5\pi_4)^2\pi_3\pi_2\pi_5$ and $(\pi_4\pi_1\pi_3)^2\pi_4$ by agents $1, 2$ and 3 respectively, where the superscript 2 here denotes that the corresponding paths are executed twice. Fig. 6.3 depicts the resulting control inputs $u_i, \forall i \in \{1, 2, 3\}$. The figures demonstrate the successful execution of the agents' paths and therefore, satisfaction of the respective formulas with inter-agent collision avoidance.

6.3.4 Experimental Results

The validity and efficiency of the proposed solution was also verified through real-time experiments. The experimental setup involved two remotely controlled *IRIS+* quadrotors from 3D Robotics, which we consider to have sensing range $d_{s_i} = 0.65$m, upper control input bound $|u_m| \leq 1$m/s, $m \in \{x, y, z\}$, and bounding spheres with radius $r_i = 0.3$m, $\forall i \in \{1, 2\}$. We considered two 2-dimensional scenarios in a workspace $\mathcal{W} = \{p \in \mathbb{R}^2 \text{ s.t. } \|p\| \leq 2.5\text{m}\}$, i.e. $p_0 = [0, 0]^T$ and $r_0 = 2.5$m.

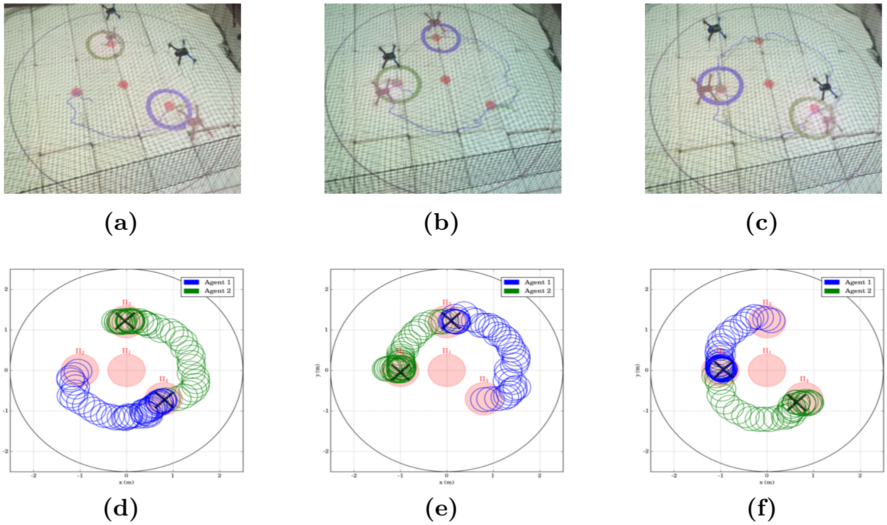
The first scenario included 4 regions of interest $\Pi = \{\pi_1, \dots, \pi_4\}$ in $\mathcal{W}$, with $r_{\pi_k} = 0.4, \forall k \in \{1, \dots, 4\}$ and $p_{\pi_1} = [0, 0]^T$m, $p_{\pi_2} = [-1, 0]^T$m, $p_{\pi_3} = [0, 1.25]^T$m and $p_{\pi_4} = [0.8, -0.7]^T$m. The initial positions of the agents were taken such that $\mathcal{A}_1(p_1(0)) \in \pi_2$ and $\mathcal{A}_2(p_2(0)) \in \pi_4$ (see Fig. 6.4). We also defined the atomic propositions $\Psi_1 = \Psi_2 = \{\text{“obs”}, \text{“}a\text{”}, \text{“}b\text{”}, \text{“}c\text{”}\}$ with $L_1(\pi_1) = L_2(\pi_1) = \{\text{“obs”}\}, L_1(\pi_2) = L_2(\pi_2) = \{\text{“}a\text{”}\}, L_1(\pi_3) = L_2(\pi_3) = \{\text{“}b\text{”}\}, L_1(\pi_4) = L_2(\pi_4) = \{\text{“}c\text{”}\}$. In this scenario, we were interested in area inspection while avoiding the ”obstacle” region, and thus, we defined the individual specifications with the following LTL formulas: $\phi_1 = \phi_2 = \square\neg\text{“obs”} \wedge \square\lozenge(\text{“}a\text{”} \bigcirc \text{“}c\text{”} \bigcirc \text{“}b\text{”})$. By following the procedure described in Section 6.3.2, we obtained the paths $p_1 = (\pi_2\pi_4\pi_3)^\omega, p_2 = (\pi_4\pi_2\pi_3)^\omega$. Fig. 6.6 depicts the execution of the paths $(\pi_2\pi_4\pi_3)^1$ and $(\pi_4\pi_2\pi_3)^1$ by agents 1 and 2, respectively, and Fig. 6.7 shows the corresponding input signals, which do not exceed the control bounds 1m/s. It can be deduced by the figures that the agents successfully satisfy their individual formulas, without colliding with each other.

The second experimental scenario included 3 regions of interest $\Pi = \{\pi_1, \dots, \pi_3\}$ in $\mathcal{W}$, with $r_{\pi_k} = 0.4, \forall k \in \{1, \dots, 3\}$ and $p_{\pi_1} = [-1, -1.7]^T$m, $p_{\pi_2} = [-1.3, 1.3]^T$m and $p_{\pi_3} = [1.2, 0]^T$m. The initial positions of the agents were taken such that $\mathcal{A}_1(p_1(0)) \in \pi_1$ and $\mathcal{A}_i(p_2(0)) \in \pi_2$ (see Fig. 6.8). We also defined the atomic propositions $\Psi_1 = \Psi_2 = \{\text{“res}_\text{a}\text{”}, \text{“res}_\text{b}\text{”}, \text{“base”}\}$, corresponding to a base and several resources in the workspace, with $L_1(\pi_1) = L_2(\pi_1) = \{\text{“res}_\text{a}\text{”}\}, L_1(\pi_2) = L_2(\pi_2) = \{\text{“base”}\}, L_1(\pi_3) = L_2(\pi_3) = \{\text{“res}_\text{b}\text{”}\}$. We considered that the agents had to transfer the resources to the ”base” in π_2; both agents were responsible for “res$_\text{a}$” but only agent 1 should access “res$_\text{b}$”. The specifications were translated to the formulas $\phi_1 = \square(\lozenge(\text{“res}_\text{a}\text{”} \bigcirc \text{“base”}) \wedge \lozenge(\text{“res}_\text{b}\text{”} \bigcirc \text{“base”})), \phi_2 = \square\neg\text{“res}_\text{b}\text{”} \wedge \square\lozenge(\text{“res}_\text{a}\text{”} \bigcirc \text{“base”})$ and the derived paths were $p_1 = (\pi_1\pi_2\pi_3\pi_2)^\omega$ and $p_2 = (\pi_1\pi_2)^\omega$. The execution of



(a) (b) (c) (d) (e) (f) (g) (h)

Figure 6.5: Execution of the paths $(\pi_1\pi_5\pi_2)^2\pi_1, (\pi_3\pi_2\pi_5\pi_4)^2\pi_3\pi_2\pi_5$ and $(\pi_4\pi_1\pi_3)^2\pi_4$ by agents $1, 2$ and 3, respectively, for the simulation studies.



(a) (b) (c)

(d) (e) (f)

Figure 6.6: Execution of the paths $(\pi_2\pi_4\pi_3)^1$ and $(\pi_4\pi_3\pi_2)^1$ by agents 1 and 2, respectively for the first experimental scenario. (a), (d): $\pi_2 \to_1 \pi_4, \pi_4 \to_2 \pi_3$, (b), (e): $\pi_4 \to_1 \pi_3, \pi_3 \to_2 \pi_2$, (c), (f):$\pi_3 \to_1 \pi_2, \pi_2 \to_2 \pi_4$.

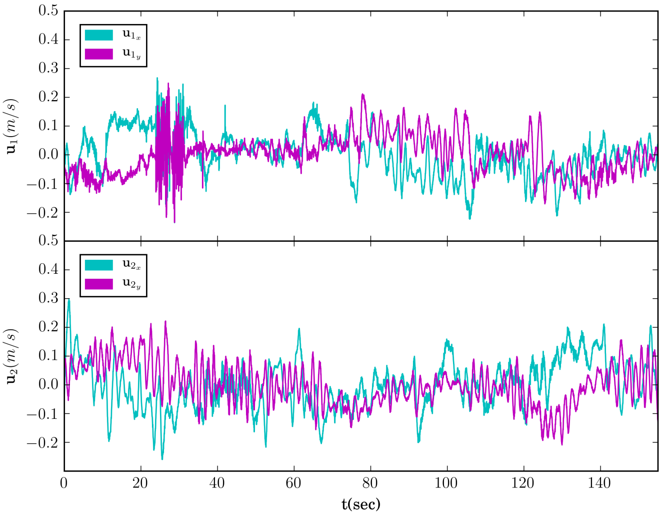
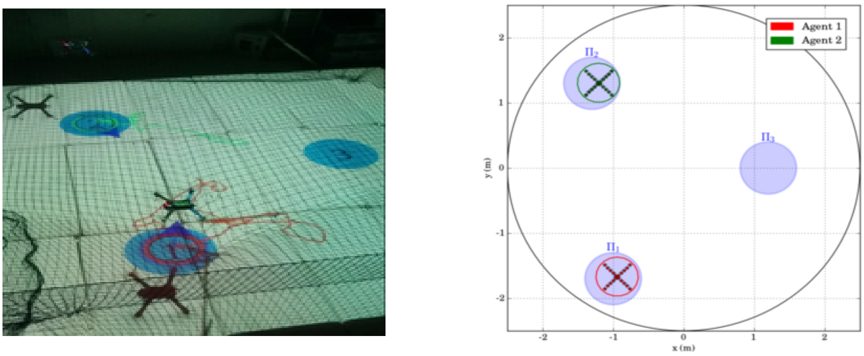


Figure 6.7: The resulting 2-dimensional control signals of the 2 agents for the first experimental scenario. Top: agent 1, bottom: agent 2.

the paths $(\pi_1\pi_2\pi_3\pi_2)^1$ and $(\pi_2\pi_1)^2$ by agents 1 and 2, respectively, are depicted in Fig. 6.10, and the corresponding control inputs are shown in Fig. 6.9. The figures



(a) **(b)**

Figure 6.8: Initial workspace for the second experimental scenario. (a): The UAVs with the projection of their bounding spheres, (with red and green), and the regions of interest (blue disks). (b): Top view of the described workspace. The UAVs are represented by the red and green circled X's and the regions of interest by the blue disks $\pi_1, \dots, \pi_3$.

demonstrate the successful execution and satisfaction of the paths and formulas, respectively, and the compliance with the control input bounds.

Regarding the continuous control protocol in the aforementioned experiments, we chose $k_{g_i} = 3, \lambda_i = 2$ in (6.3), (6.5) and the switching duration in (6.7) as $\nu_i = 0.1t'_{i,k}, \forall i \in \{1, 2\}$.

Remark 6.2. Note that, although the limited available workspace in the experiments did not satisfy all the conditions regarding the distance between regions and the workspace boundary, as introduced in Section 6.3.1, the two experimental scenarios were successfully conducted.

The simulations and experiments were conducted in Python environment using an Intel Core i7 2.4 GHz personal computer with 4 GB of RAM, and are clearly demonstrated in the video found in https://youtu.be/dO77ZYEFHlE.

6.4 Robust Decentralized Abstractions for Multiple Mobile Manipulators

6.4.1 Problem Formulation

Consider $N \in \mathbb{N}$ fully actuated agents with $\mathcal{V} := \{1, \dots, N\}, N \geq 2$, composed by a robotic arm mounted on an omnidirectional mobile base, operating in a static workspace $\mathcal{W}$ that is bounded by a large sphere in 3D space, i.e. $\mathcal{W} = \mathring{\mathcal{B}}_{p_0, r_0} = \{p \in \mathbb{R}^3 \text{ s.t. } \|p - p_0\| < r_0\}$, where $p_0 \in \mathbb{R}^3$ is the center of $\mathcal{W}$, and $r_0 \in \mathbb{R}_{\geq 0}$ is

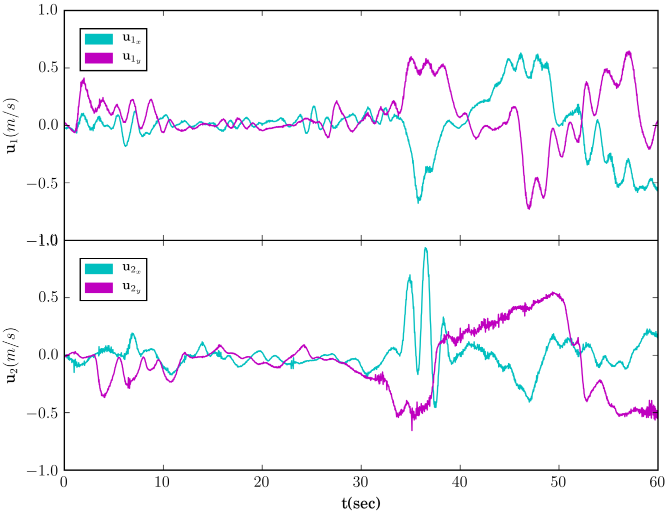
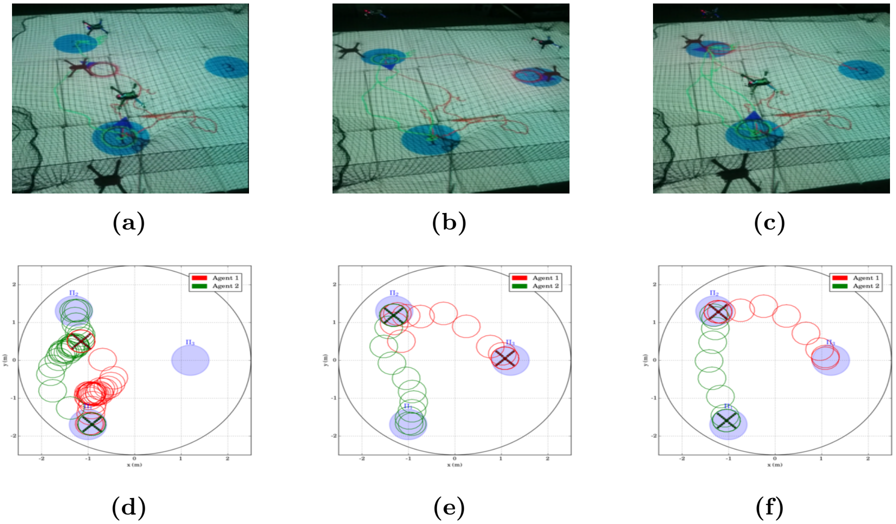


Figure 6.9: The resulting 2-dimensional control signals of the 2 agents for the second experimental scenario. Top: agent 1, bottom: agent 2.



(a) **(b)** **(c)**

(d) **(e)** **(f)**

Figure 6.10: Execution of the paths $(\pi_1\pi_2\pi_3\pi_2)^1$ and $(\pi_2\pi_1)^2$ by agents 1 and 2, respectively for the second experimental scenario. (a), (d): $\pi_1 \to_1 \pi_2, \pi_2 \to_2 \pi_1$, (b), (e): $\pi_2 \to_1 \pi_3, \pi_1 \to_2 \pi_2$, (c), (f): $\pi_3 \to_1 \pi_2, \pi_2 \to_2 \pi_1$.

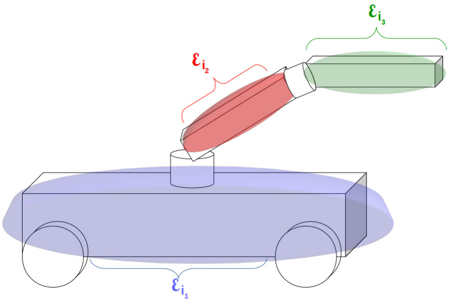


Figure 6.11: An agent that consists of $\ell_i = 3$ rigid links.

its radius. Without loss of generality, we consider that $p_0 = 0_{3\times 1}$, corresponding to an inertial frame $\{I\}$. Within $\mathcal{W}$ there exist K disjoint spheres around points of interest, which are described by $\pi_k = \mathcal{B}_{p_k,r_k} = \{p \in \mathbb{R}^3 \text{ s.t. } \|p - p_k\| \leq r_k\}, k \in \mathcal{K} := \{1, \dots, K\}$, where $p_k \in \mathbb{R}^3$ and $r_k \in \mathbb{R}_{>0}$ are the center and radius of the kth region, respectively. The regions of interest can be equivalently described by $\pi_k = \{z \in \mathbb{R}^4 \text{ s.t. } z^\top T_{\pi_k} z \leq 0\}$, where $z = [p^\top, 1]^\top$ is the vector of homogeneous coordinates of $p \in \mathbb{R}^3$, and

$$T_{\pi_k} = \begin{bmatrix} I_3 & p_k \\ 0_{3\times 1}^\top & -r_k^2 \end{bmatrix}, \forall k \in \mathcal{K}. \tag{6.9}$$

The dynamic model of each agent is given by the second-order Lagrangian dynamics:

$$M_i(q_i)\ddot{q}_i + N_{q_i}(q_i, \dot{q}_i)\dot{q}_i + g_i(q_i) + f_i(q_i, \dot{q}_i) = \tau_i, \tag{6.10}$$

$\forall i \in \mathcal{V}$, where $q_i \in \mathbb{R}^{n_i}$ is the vector of generalized coordinates (e.g., pose of mobile base and joint coordinates of the arms), and the rest of the terms as in (4.3) with a slight change of notation;$f_i(\cdot)$ here represents unmodeled nonlinearities and external disturbances. Without loss of generality, we assume that $n_i = n \in \mathbb{N}, \forall i \in \mathcal{V}$. In addition, we denote as $\{B_i\}$ the frame of the mobile base of agent i and $p_{B_i} : \mathbb{R}^n \to \mathbb{R}^3$ its inertial position. Moreover, the matrix $\dot{M}_i - 2N_i$ is skew-symmetric [169], and we further make the following assumption:

Assumption 6.1. There exist positive constants c_i such that $\|f_i(q_i, \dot{q}_i)\| \leq c_i \|q_i\| \|\dot{q}_i\|$, $\forall (q_i, \dot{q}_i) \in \mathbb{R}^n \times \mathbb{R}^n$, $i \in \mathcal{V}$.

We consider that each agent is composed by ℓ_i rigid links (see Fig. 6.11) with $\mathcal{Q}_i = \{1, \dots, \ell_i\}$ the corresponding index set. Each link of agent i is approximated by the ellipsoid set [167] $\mathcal{E}_{i_m}(q_i) = \{z \in \mathbb{R}^4 \text{ s.t. } z^\top E_{i_m}(q_i) z \leq 0\}$; $z = [p^\top, 1]^\top$ is the homogeneous coordinates of $p \in \mathbb{R}^3$, and $E_{i_m} : \mathbb{R}^n \to \mathbb{R}^{4\times 4}$ is defined as $E_{i_m}(q_i) = T_{i_m}^{-T}(q_i) \hat{E}_{i_m} T_{i_m}^{-1}(q_i)$, where $\hat{E}_{i_m} = \text{diag}\{a_{i_m}^{-2}, b_{i_m}^{-2}, c_{i_m}^{-2}, -1\}$ corresponds to the positive lengths $a_{i_m}, b_{i_m}, c_{i_m}$ of the principal axes of the ellipsoid, and

$T_{i_m} : \mathbb{R}^n \to \mathbb{R}^{4\times 4}$ is the transformation matrix for the coordinate frame $\{i_m\}$ placed at the center of mass of the m-th link of agent i, aligned with the principal axes of $\mathcal{E}_{i_m}$:

$$T_{i_m}(q_i) = \begin{bmatrix} R_{i_m}(q_i) & p_{i_m}(q_i) \\ 0_{3\times 1}^\top & 1 \end{bmatrix},$$

with $R_{i_m} : \mathbb{R}^n \to \mathbb{R}^{3\times 3}$ being the rotation matrix of the center of mass of the link, $\forall m \in \mathcal{Q}_i, i \in \mathcal{V}$. For an ellipsoid $\mathcal{E}_{i_m}, i \in \mathcal{V}, m \in \mathcal{Q}_i$, we denote as $\mathcal{E}_{i_m}^{xy}, \mathcal{E}_{i_m}^{xz}, \mathcal{E}_{i_m}^{yz}$ its projections on the planes x-y, x-z and y-z, respectively, with corresponding matrix terms $E_{i_m}^{xy}, E_{i_m}^{xz}, E_{i_m}^{yz}$. Note that the following holds for two different ellipsoids $\mathcal{E}_{i_m}$ and $\mathcal{E}_{j_l}$:

$$\begin{aligned} &\mathcal{E}_{i_m}(q_i) \cap \mathcal{E}_{j_l}(q_j) \neq \emptyset \ \wedge \ \mathring{\mathcal{E}}_{i_m}(q_i) \cap \mathring{\mathcal{E}}_{j_l}(q_j) = \emptyset \Leftrightarrow \\ &\mathcal{E}_{i_m}^s(q_i) \cap \mathcal{E}_{j_l}^s(q_j) \neq \emptyset \ \wedge \ \mathring{\mathcal{E}}_{i_m}^s(q_i) \cap \mathring{\mathcal{E}}_{j_l}^s(q_j) = \emptyset, \end{aligned}$$

$\forall s \in \{xy, xz, yz\}$, i.e., in order for $\mathcal{E}_{i_m}, \mathcal{E}_{j_l}$ to collide (touch externally), all their projections on the three planes must also collide. Therefore, a sufficient condition for $\mathcal{E}_{i_m}$ and $\mathcal{E}_{j_l}$ not to collide is $\mathcal{E}_{i_m}^s(q_i) \cap \mathcal{E}_{j_l}^s(q_j) = \emptyset$, for some $s \in \{xy, xz, yz\}$. In view of Proposition 6.2 in Chapter 2, that means that the characteristic equations $f_{i_m,j_l}^s(\lambda) := \det(\lambda \mathcal{E}_{i_m}^s(q_i) - \mathcal{E}_{j_l}^s(q_j)) = 0$ must always have one positive real root and two negative distinct roots for at least one $s \in \{xy, xz, yz\}$. Hence, be denoting the discriminant of $f_{i_m,j_l}^s(\lambda) = 0$ as Δ_{i_m,j_l}^s, Proposition 6.1 in Chapter 2 suggests that Δ_{i_m,j_l}^s must remain always positive for at least one $s \in \{xy, xz, yz\}$, since a collision would imply $\Delta_{i_m,j_l}^s = 0$, $\forall s \in \{xy, xz, yz\}$. Therefore, by defining the function $\delta : \mathbb{R} \to \mathbb{R}_{\geq 0}$ as:

$$\delta(x) = \begin{cases} \phi_\delta(x), & x > 0, \\ 0, & x \leq 0, \end{cases} \tag{6.11}$$

where ϕ_δ is an appropriate polynomial that ensures that $\delta(x)$ is twice continuously differentiable everywhere (e.g. $\phi_\delta(x) = x^3$), we can conclude that a sufficient condition for $\mathcal{E}_{i_m}$ and $\mathcal{E}_{j_l}$ not to collide is $\delta(\Delta_{i_m,j_l}^{xy}) + \delta(\Delta_{i_m,j_l}^{xz}) + \delta(\Delta_{i_m,j_l}^{yz}) > 0$, since a collision would result in $\Delta_{i_m,j_l}^s = 0 \Leftrightarrow \delta(\Delta_{i_m,j_l}^s) = 0, \forall s \in \{xy, xz, yz\}$.

Next, we define the constant $\bar{d}_{B_i}$, which is the maximum distance of the base to a point in the agent's volume over all possible configurations, i.e. $\bar{d}_{B_i} = \sup_{q_i \in \mathbb{R}^n}\{\|p_{B_i}(q_i) - p_i(q_i)\|\}, p_i \in \bigcup_{m\in\mathcal{Q}_i} \mathcal{E}_{i_m}(q_i)$. We also denote $\bar{d}_B = [\bar{d}_{B_1}, \ldots, \bar{d}_{B_N}]^\top \in \mathbb{R}_{\geq 0}^N$. Moreover, we consider that each agent has a sensor located at the center of its mobile base p_{B_i} with a sensing radius $d_{\text{con}_i} \geq 2\max_{i\in\mathcal{V}}\{\bar{d}_{B_i}\} + \varepsilon_d$, where ε_d is an arbitrarily small positive constant. Hence, each agent has the sensing sphere $\mathcal{D}_i(q_i) = \{p \in \mathbb{R}^3 \text{ s.t. } \|p - p_{B_i}(q_i)\| \leq d_{\text{con}_i}\}$ and its neighborhood set at each time instant is defined as $\mathcal{N}_i(q_i) = \{j \in \mathcal{V}\backslash\{i\} \text{ s.t. } \|p_{B_i}(q_i) - p_{B_j}(q_j)\| \leq d_{\text{con}_i}\}$.

As mentioned in Section 6.1, we are interested in defining transition systems for the motion of the agents in the workspace in order to be able to assign complex high level goals through logic formulas. Moreover, since many applications necessitate the cooperation of the agents in order to execute some task (e.g. transport an object), we consider that a nonempty subset $\widetilde{\mathcal{N}}_i \subseteq \mathcal{N}_i(q_i(0)), i \in \mathcal{V}$, of the initial neighbors of the agents must stay connected through their motion in the workspace. In addition, it follows that the transition system of each agent must contain information regarding the current position of its neighbors. The problem in hand is equivalent to designing decentralized control laws $\tau_i, i \in \mathcal{V}$, for the appropriate transitions of the agents among the predefined regions of interest in the workspace.

Next, we provide the following necessary definitions.

Definition 6.5. An agent $i \in \mathcal{V}$ is in region $k \in \mathcal{K}$ at a configuration $q_i \in \mathbb{R}^n$, denoted as $\mathcal{A}_i(q_i) \in \pi_k$, if and only if $\|p_{i_m}(q_i) - p_k\| \leq r_k - \max\{\alpha_{i_m}, \beta_{i_m}, c_{i_m}\}, \forall m \in \mathcal{Q}_i \Rightarrow \|p_{B_i}(q_i) - p_k\| \leq r_k - \bar{d}_{B_i}$.

Definition 6.6. Agents $i, j \in \mathcal{V}$, with $i \neq j$, are in *collision-free* configurations $q_i, q_j \in \mathbb{R}^n$, denoted as $\mathcal{A}_i(q_i) \not\equiv \mathcal{A}_j(q_j)$, if and only if $\mathcal{E}_{i_m}(q_i) \cap \mathcal{E}_{j_l}(q_j) = \emptyset, \forall m \in \mathcal{Q}_i, l \in \mathcal{Q}_j$.

Given the aforementioned discussion, we make the following assumptions regarding the agents and the validity of the workspace:

Assumption 6.2. The regions of interest are

(i) large enough such that all the robots can fit, i.e., given a specific $k \in \mathcal{K}$, there exist $q_i, i \in \mathcal{V}$, such that $\mathcal{A}_i(q_i) \in \pi_k$, $\forall i \in \mathcal{V}$, with $\mathcal{A}_i(q_i) \not\equiv \mathcal{A}_j(q_j)$, $\forall i, j \in \mathcal{V}$, with $i \neq j$.

(ii) sufficiently far from each other and the obstacle workspace, i.e.,

$$\begin{aligned}
\|p_k - p_{k'}\| &\geq \max_{i \in \mathcal{V}}\{2\bar{d}_{B_i}\} + r_k + r_{k'} + \varepsilon_p, \\
r_0 - \|p_k\| &\geq \max_{i \in \mathcal{V}}\{2\bar{d}_{B_i}\},
\end{aligned}$$

$\forall k, k' \in \mathcal{K}, k \neq k'$, where ε_p is an arbitrarily small positive constant.

Next, in order to proceed, we need the following definition.

Definition 6.7. Assume that $\mathcal{A}_i(q_i(t_0)) \in \pi_k, i \in \mathcal{V}$, for some $t_0 \in \mathbb{R}_{\geq 0}, k \in \mathcal{K}$, with $\mathcal{A}_i(q_i(t_0)) \not\equiv \mathcal{A}_j(q_j(t_0)), \forall j \in \mathcal{V} \backslash \{i\}$. There exists a transition for agent i between π_k and $\pi_{k'}, k' \in \mathcal{K}$, denoted as $(\pi_k, t_0) \xrightarrow{i} (\pi_{k'}, t_f)$, if and only if there exists a finite time $t_f \geq t_0$, such that $\mathcal{A}_i(q_i(t_f)) \in \pi_{k'}$ and $\mathcal{A}_i(q_i(t)) \not\equiv \mathcal{A}_j(q_j(t))$, $\mathcal{E}_{i_m}(q_i(t)) \cap \mathcal{E}_{i_\ell}(q_i(t))$, $\mathcal{E}_{i_m}(q_i(t)) \cap \pi_z = \emptyset, \forall m, \ell \in \mathcal{Q}_i, m \neq \ell, j \in \mathcal{V} \backslash \{i\}, z \in \mathcal{K} \backslash \{k, k'\}, t \in [t_0, t_f]$.

Given the aforementioned definitions, the treated problem is the design of decentralized control laws for the transitions of the agents between two regions of interest in the workspace, while preventing collisions of the agents with each other, the workspace boundary, and the remaining regions of interest. More specifically, we aim to design a finite transition system for each agent of the form [38]

$$\mathcal{T}_i = (\Pi, \Pi_{i,0}, \xrightarrow{i}, \mathcal{AP}_i, \mathcal{L}_i, \mathcal{F}_i), \tag{6.12}$$

where $\Pi = \{\pi_1, \dots, \pi_K\}$ is the set of regions of interest that the agents can be at, according to Def. 6.5, $\Pi_{i,0} \subseteq \Pi$ is a set of initial regions that each agent can start from, $\xrightarrow{i} \subset (\Pi \times \mathbb{R}_{\geq 0})^2$ is the transition relation of Def. 6.7, $\mathcal{AP}_i$ is a set of given atomic propositions, represented as boolean variables, that hold in the regions of interest, $\mathcal{L}_i : \Pi \to 2^{\mathcal{AP}_i}$ is a labeling function, and $\mathcal{F}_i : \Pi \to \Pi^{|\widetilde{\mathcal{N}}_i|}$ is a function that maps the region that agent i occupies to the regions the initial neighbors $\widetilde{\mathcal{N}}_i$ of agent i are at. Therefore, the treated problem is the design of bounded controllers τ_i for the establishment of the transitions $\xrightarrow{i}$. Moreover, as discussed before, the control protocol should also guarantee the connectivity maintenance of a subset of the initial neighbors $\widetilde{\mathcal{N}}_i, \forall i \in \mathcal{V}$. Another desired property important in applications involving robotic manipulators, is the nonsingularity of the Jacobian matrix $J_i : \mathbb{R}^n \to \mathbb{R}^{6\times n}$, that transforms the generalized coordinate rates of agent $i \in \mathcal{V}$ to generalized velocities [169]. That is, the set $\mathbb{S}_i = \{q_i \in \mathbb{R}^n \text{ s.t. } \det(J_i(q_i)[J_i(q_i)]^\top) = 0\}$ should be avoided, $\forall i \in \mathcal{V}$.

Formally, we define the problem treated in this section as follows:

Problem 6.2. Consider N mobile manipulators with dynamics (6.10) and K regions of interest $\pi_k, k \in \mathcal{K}$, with $\dot{q}_i(t_0) < \infty, \mathcal{A}_i(q_i(t_0)) \in \pi_{k_i}, k_i \in \mathcal{K}, \forall i \in \mathcal{V}$ and $\mathcal{A}_i(q_i(t_0)) \not\equiv \mathcal{A}_j(q_j(t_0)), \mathcal{E}_{i_m}(q_i(t_0)) \cap \mathcal{E}_{i_\ell}(q_i(t_0)) = \emptyset, \forall i, j \in \mathcal{V}, i \neq j, m, \ell \in \mathcal{Q}_i, m \neq \ell$. Given nonempty subsets of the initial edge sets $\widetilde{\mathcal{N}}_i \subseteq \mathcal{N}_i(q_i(0)) \subseteq \mathcal{V}, \forall i \in \mathcal{V}$, the fact that $\det(J_i(q_i(t_0))[J_i(q_i(t_0))]^\top) \neq 0, \forall i \in \mathcal{V}$, as well as the indices $k'_i \in \mathcal{K}, i \in \mathcal{V}$, such that $\|p_{k'_i} - p_{k'_j}\| + r_{k'_i} + r_{k'_j} \leq d_{\text{con}_i}, \forall j \in \widetilde{\mathcal{N}}_i, i \in \mathcal{V}$, design decentralized controllers τ_i such that, for all $i \in \mathcal{V}$:

1. $(\pi_{k_i}, t_0) \xrightarrow{i} (\pi_{k'_i}, t_{f_i})$, for some $t_{f_i} \geq t_0$,
2. $r_0 - (\|p_{B_i}(t)\| + \bar{d}_{B_i}) > 0, \forall t \in [t_0, t_{f_i}]$,
3. $j_i^* \in \mathcal{N}_i(q_i(t)), \forall j_i^* \in \widetilde{\mathcal{N}}_i, t \in [t_0, t_{f_i}]$,
4. $q_i(t) \in \mathbb{R}^n \backslash \mathbb{S}_i, \forall t \in [t_0, t_{f_i}]$.

The aforementioned specifications concern 1) the agent transitions according to Def. 6.7, 2) the confinement of the agents in $\mathcal{W}$, 3) the connectivity maintenance between a subset of initially connected agents and 4) the agent singularity avoidance. Moreover, the fact that the initial edge sets $\widetilde{\mathcal{N}}_i$ are nonempty implies that the sensing radius of each agent i covers the regions π_{k_j} of the agents in the neighboring set $\widetilde{\mathcal{N}}_i$. Similarly, the condition $\|p_{k'_i} - p_{k'_j}\| + r_{k'_i} + r_{k'_j} \leq d_{\text{con}_i}, \forall j \in \widetilde{\mathcal{N}}_i$, is a feasibility condition for the goal regions, since otherwise it would be impossible for two initially connected agents to stay connected. Intuitively, the sensing radii d_{con_i} should be large enough to allow transitions of the multi-agent system to the entire workspace.

6.4.2 Main Results

Continuous Control Design

To solve Problem 6.2, we denote as $\varphi_i : \mathbb{R}^{Nn} \to \mathbb{R}_{\geq 0}$ a *decentralized potential function*, with the following properties:

(i) The function $\varphi_i(q)$ is not defined, i.e., $\varphi_i(q) = \infty$, $\forall i \in \mathcal{V}$, when a collision or a connectivity break occurs,

(ii) The critical points of φ_i where the vector field $\nabla_{q_i}\varphi_i(q)$ vanishes, i.e., the points where $\nabla_{q_i}\varphi_i(q) = 0$, consist of the goal configurations and a set of configurations whose region of attraction (by following the negated vector field curves) is a set of measure zero.

(iii) It holds that $\nabla_{q_i}\varphi_i(q) + \sum_{j\in\mathcal{N}_i(q_i)} \nabla_{q_i}\varphi_j(q) = 0 \Leftrightarrow \nabla_{q_i}\varphi_i(q) = 0$ and $\sum_{j\in\mathcal{N}_i(q_i)} \nabla_{q_i}\varphi_j(q) = 0$, $\forall i \in \mathcal{N}, q \in \mathbb{R}^{Nn}$.

More specifically, $\varphi_i(q)$ is a function of two main terms, a *goal function* $\gamma_i : \mathbb{R}^n \to \mathbb{R}_{\geq 0}$, that should vanish when $\mathcal{A}_i(q_i) \in \pi_{k_i'}$, and an *obstacle function*, $\beta_i : \mathbb{R}^n \to \mathbb{R}_{\geq 0}$ is a bounded that encodes inter-agent collisions, collisions between the agents and the obstacle boundary/undesired regions of interest, connectivity losses between initially connected agents and singularities of the Jacobian matrix $J_i(q_i)$;

Next, we provide an analytic construction of the goal and obstacle terms. However, the construction of the function φ_i is out of the scope of this work. Examples can be found in [37][1] and [170].

γ_i **- Goal Function**

Function γ_i encodes the control objective of agent i, i.e., reach the region of interest $\pi_{k_i'}$. Hence, we define $\gamma_i : \mathbb{R}^n \to \mathbb{R}_{\geq 0}$ as

$$\gamma_i(q_i) = \|q_i - q_{k_i'}\|^2, \tag{6.13}$$

where $q_{k_i'}$ is a configuration such that $r_{k'} - \|p_{B_i}(q_{k_i'}) - p_{k_i'}\| \leq \bar{d}_{B_i} - \varepsilon$, for an arbitrarily small positive constant ε, which implies $\mathcal{A}_i(q_{k_i'}) \in \pi_{k_i'}$, $\forall i \in \mathcal{V}$. In case that multiple agents have the same target, i.e., there exists at least one $j \in \mathcal{V}\backslash\{i\}$ such that $\pi_{k_j'} = \pi_{k_i'}$, then we assume that $\mathcal{A}_i(q_{k_i'}) \not\equiv \mathcal{A}_j(q_{k_j'})$.

β_i **- Collision/Connectivity/Singularity Function**

The function β_i encodes all inter-agent collisions, collisions with the boundary of the workspace and the undesired regions of interest, connectivity between initially connected agents and singularities of the Jacobian matrix $J_i(q_i), \forall i \in \mathcal{V}$.

Consider the function $\Delta_{i_m,j_l} : \mathbb{R}^{2n} \to \mathbb{R}_{\geq 0}$, with $\Delta_{i_m,j_l}(q_i, q_j) = \delta(\Delta^{xy}_{i_m,j_l}(q_i, q_j)) + \delta(\Delta^{xz}_{i_m,j_l}(q_i, q_j)) + \delta(\Delta^{yz}_{i_m,j_l}(q_i, q_j))$, where $\Delta^s_{i_m,j_l} : \mathbb{R}^{2n} \to \mathbb{R}_{\geq 0}$ is the discriminant of the cubic equation $\det\{\lambda E^s_{i_m}(q_i) - E^s_{j_l}(q_j)\} = 0, \forall s \in \{xy, xz, yz\}$, for two given ellipsoids $\mathcal{E}_{i_m}$ and $\mathcal{E}_{j_l}, m \in \mathcal{Q}_i, l \in \mathcal{Q}_j, i, j, \in \mathcal{V}$, and δ as defined in (6.11). As

[1] In that case, we could choose $\varphi_i = \frac{1}{1-\phi_i}$, where ϕ_i is the proposed function of [37]

discussed in Section 6.4.1, a sufficient condition for the ellipsoids $\mathcal{E}_{i_m}$ and $\mathcal{E}_{j_l}$ not to collide, is $\Delta_{i_m,j_l}(q_i(t), q_j(t)) > 0, \forall t \in \mathbb{R}_{\geq 0}$.

Additionally, we define the greatest lower bound of the Δ_{i_m,j_l} when the point p_{j_l} is on the boundary of the sensing radius $\partial D_i(q_i)$ of agent i, as $\widetilde{\Delta}_{i_m,j_l} = \inf_{(q_i,q_j)\in\mathbb{R}^{2n}}\{\Delta_{i_m,j_l}(q_i,q_j)\}$ s.t. $\|p_{B_i}(q_i) - p_{j_l}(q_j)\| = d_{\text{con}_i}, \forall m \in \mathcal{Q}_i, l \in \mathcal{Q}_j, i, j \in \mathcal{V}$. Since $d_{\text{con}_i} > 2\max_{i\in\mathcal{V}}\{\bar{d}_{B_i}\} + \varepsilon_d$, it follows that there exists a positive constant ε_Δ such that $\widetilde{\Delta}_{i_m,j_l} \geq \varepsilon_\Delta > 0, \forall m \in \mathcal{Q}_i, l \in \mathcal{Q}_j, i, j \in \mathcal{V}, i \neq j$.

Moreover, we define the function $\Delta_{i_m,\pi_k} : \mathbb{R}^n \to \mathbb{R}_{\geq 0}$, with $\Delta_{i_m,\pi_k}(q_i) = \delta(\Delta^{xy}_{i_m,\pi_k}(q_i)) + \delta(\Delta^{xz}_{i_m,\pi_k}(q_i)) + \delta(\Delta^{yz}_{i_m,\pi_k}(q_i))$, where $\Delta^{s}_{i_m,\pi_k} : \mathbb{R}^n \to \mathbb{R}$ is the discriminant of the cubic equation $\det(\lambda E^s_{i_m}(q_i) - T^s_{\pi_k})$, with $T^s_{\pi_k}$ the projected version of T_{π_k} in (6.9), $s \in \{xy, xz, yz\}$, and δ as given in (6.11). A sufficient condition for $\mathcal{E}_{i_m}$ and region $\pi_k, k \in \mathcal{K}$ not to collide is $\Delta_{i_m,\pi_k}(q_i(t)) > 0, \forall t \in \mathbb{R}_{\geq 0}, m \in \mathcal{Q}_i, i \in \mathcal{V}$.

We further define the function $\eta_{ij,c} : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$, with $\eta_{ij,c}(q_i, q_j) = d^2_{\text{con}_i} - \|p_{B_i}(q_i) - p_{B_j}(q_j)\|^2$, and the distance functions $\beta_{i_m,j_l} : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}, \beta_{ij,c} : \mathbb{R} \to \mathbb{R}_{\geq 0}, \beta_{iw} : \mathbb{R}_{\geq 0} \to \mathbb{R}$ as

$$\beta_{i_m,j_l}(\Delta_{i_m,j_l}) = \begin{cases} \phi_{i,a}(\Delta_{i_m,j_l}), & 0 \leq \Delta_{i_m,j_l} < \bar{\Delta}_{i_m,j_l}, \\ \bar{\Delta}_{i_m,j_l}, & \bar{\Delta}_{i_m,j_l} \leq \Delta_{i_m,j_l}, \end{cases}$$

$$\beta_{ij,c}(\eta_{ij,c}) = \begin{cases} 0, & \eta_{ij,c} < 0, \\ \phi_{i,c}(\eta_{ij,c}), & 0 \leq \eta_{ij,c} < d^2_{\text{con}_i}, \\ d^2_{\text{con}_i}, & d^2_{\text{con}_i} \leq \eta_{ij,c}, \end{cases}$$

$$\beta_{iw}(\|p_{B_i}\|^2) = (r_w - \bar{d}_{B_i})^2 - \|p_{B_i}\|^2,$$

where $\bar{\Delta}_{i_m,j_l}$ is a constant satisfying $0 < \bar{\Delta}_{i_m,j_l} \leq \widetilde{\Delta}_{i_m,j_l}, \forall m \in \mathcal{Q}_i, l \in \mathcal{Q}_j, i, j \in \mathcal{V}, i \neq j$, and $\phi_{i,a}, \phi_{i,c}$ are *strictly increasing* polynomials appropriately selected to guarantee that the functions β_{i_m,j_l}, and $\beta_{ij,c}$, respectively, are twice continuously differentiable everywhere, with $\phi_{i,a}(0) = \phi_{i,c}(0) = 0, \forall i \in \mathcal{V}$. Note that the functions defined above use only local information in the sensing range d_{con_i} of agent i. The function β_{i_m,j_l} becomes zero when ellipsoid $\mathcal{E}_{i_m}$ collides with ellipsoid $\mathcal{E}_{j_l}$, whereas $\beta_{ij,c}$ becomes zero when agent i loses connectivity with agent j. Similarly, β_{iw} encodes the collision of agent i with the workspace boundary.

Finally, we choose the function $\beta_i : \mathbb{R}^{Nn} \to \mathbb{R}_{\geq 0}$ as

$$\begin{aligned}\beta_i(q) =& (\det(J_i(q_i)[J_i(q_i)]^\top))^2 \beta_{iw}(\|p_{B_i}\|^2) \prod_{j\in\widetilde{\mathcal{N}}_i} \beta_{ij,c}(\eta_{ij,c}) \\ & \prod_{(m,j,l)\in\widetilde{T}} \beta_{i_m,j_l}(\Delta_{i_m,j_l}) \prod_{(m,k)\in\widetilde{L}} \Delta_{i_m,\pi_k}(q_i),\end{aligned} \tag{6.14}$$

$\forall i \in \mathcal{V}$, where $\widetilde{T} = \mathcal{Q}_i \times \mathcal{V} \times \mathcal{Q}_j, \widetilde{L} = \mathcal{Q}_i \times (\mathcal{K}\backslash\{k_i, k'_i\})$, and we have omitted the dependence on q for brevity. Note that we have included the term $(\det(J_i J_i^\top))^2$ to also account for singularities of $J_i, \forall i \in \mathcal{V}$ and the term $\prod_{(m,j,l)\in\widetilde{T}} \beta_{i_m,j_l}(\Delta_{i_m,j_l})$ takes into account also the collisions between the ellipsoidal rigid bodies of agent i.

With the introduced notation, the properties of the functions φ_i are:

(i) $\beta_i(q) \to 0 \Leftrightarrow (\varphi_i(q) \to \infty), \forall i \in \mathcal{V}$,

(ii) $\nabla_{q_i}\varphi_i(q)|_{q_i=q_i^\star} = 0, \forall q_i^\star \in \mathbb{R}^n$ s.t. $\gamma_i(q_i^\star) = 0$ and the regions of attraction of the points $\{q \in \mathbb{R}^{Nn} : \nabla_{q_i}\varphi_i(q)\big|_{q_i=\widetilde{q}_i} = 0, \gamma_i(\widetilde{q}_i) \neq 0\}, i \in \mathcal{V}$, are sets of measure zero.

By further denoting $\mathbb{D}_i = \{q \in \mathbb{R}^{Nn} : \beta_i(q) > 0\}$, we are ready to state the main theorem, that summarizes the main results of this work.

Theorem 6.2. *Under the Assumptions 6.1-6.2, the decentralized control laws $\tau_i : \mathbb{D}_i \times \mathbb{R}^n \to \mathbb{R}^n$, with*

$$\tau_i(q, \dot{q}_i) = g_i(q_i) - \nabla_{q_i}\varphi_i(q) - \sum_{j\in\mathcal{N}_i(q_i)} \nabla_{q_i}\varphi_j(q) - \hat{c}_i(q_i, \dot{q}_i)\|q_i\|\dot{q}_i - \lambda_i\dot{q}_i, \quad (6.15)$$

$\forall i \in \mathcal{V}$, along with the adaptation laws $\dot{\hat{c}}_i : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$:

$$\dot{\hat{c}}_i(q_i, \dot{q}_i) = \sigma_i\|\dot{q}_i\|^2\|q_i\|, \quad (6.16)$$

with $\hat{c}_i(q_i(t_0), \dot{q}_i(t_0)) < \infty, \sigma_i \in \mathbb{R}_{\geq 0}$, $\forall i \in \mathcal{V}$, guarantee the transitions $(\pi_{k_i}, t_0) \xrightarrow{i} (\pi_{k_i'}, t_{f_i})$ for finite $t_{f_i}, i \in \mathcal{V}$ for almost all initial conditions, while ensuring $\beta_i > 0, \forall i \in \mathcal{V}$, as well as the boundedness of all closed loop signals, providing, therefore, a solution to Problem 6.2.

Proof. The closed loop system of (6.10) is written as:

$$M_i(q_i)\ddot{q}_i + N_i(q_i, \dot{q}_i)\dot{q}_i + f_i(q_i, \dot{q}_i) = -\nabla_{q_i}\varphi_i(q_i) - \lambda_i\dot{q}_i - \hat{c}(q_i, \dot{q}_i)\|q_i\|\dot{q}_i - \sum_{j\in\mathcal{N}_i(q_i)} \nabla_{q_i}\varphi_j(q), \quad (6.17)$$

$\forall i \in \mathcal{V}$. Due to Assumption 6.2, the domain where the functions $\varphi_i(q)$ are well-defined (i.e., where $\beta_i > 0$) is connected. Hence, consider the Lyapunov-like function $V : \mathbb{R}^N \times \mathbb{R}^{Nn} \times \mathbb{R}^N \times \mathbb{D}_1 \times \cdots \times \mathbb{D}_N \to \mathbb{R}_{\geq 0}$, with

$$V(\varphi, \dot{q}, \widetilde{c}, q) = \sum_{i\in\mathcal{V}} \varphi_i(q) + \frac{1}{2}[\dot{q}_i^\top M_i(q_i)\dot{q}_i + \frac{1}{\sigma_i}\widetilde{c}_i^2]$$

where φ and $\widetilde{c}$ are the stack vectors containing all φ_i and $\widetilde{c}_i$, respectively, $i \in \mathcal{V}$, and $\widetilde{c}_i : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$, with $\widetilde{c}_i(q_i, \dot{q}_i) = \hat{c}_i(q_i, \dot{q}_i) - c_i, \forall i \in \mathcal{V}$. Note that, since there are no collision or singularities at t_0, the functions $\beta_i(q), i \in \mathcal{V}$, are strictly positive at t_0 which implies the boundedness of V at t_0. Therefore, since $\dot{q}_i(t_0) < \infty$ and

$\hat{c}_i(t_0) < \infty, \forall i \in \mathcal{V}$, there exists a positive and finite constant $M < \infty$ such that $V_0 := V(\varphi(q(t_0)), \dot{q}(t_0), \tilde{c}(q(t_0), q(t_0)) \leq M$.

By differentiating V, substituting the dynamics (6.10), employing the skew symmetry of $\dot{M}_i - 2N_i$ as well as $\sum_{i\in\mathcal{V}}([\nabla_{q_i}\varphi_i(q)]^\top \dot{q}_i + \sum_{j\in\mathcal{N}_i(q_i)}[\nabla_{q_j}\varphi_i(q)]^\top \dot{q}_j) = \sum_{i\in\mathcal{V}}([\nabla_{q_i}\varphi_i(q)]^\top + \sum_{j\in\mathcal{N}_i(q_i)}[\nabla_{q_i}\varphi_j(q)]^\top)\dot{q}_i$, we obtain

$$\dot{V} = \sum_{i\in\mathcal{V}} \left\{ \dot{q}_i^\top \left(\nabla_{q_i}\varphi_i(q) + \sum_{j\in\mathcal{N}_i(q_i)} \nabla_{q_i}\varphi_j(q) + \tau_i - g_i(q_i) \right) - \dot{q}_i^\top f_i(q_i, \dot{q}_i) + \frac{1}{\sigma_i}\widetilde{c}_i \dot{\hat{c}}_i \right\}, \tag{6.18}$$

which, by substituting the control and adaptation laws (6.15), (6.16) becomes:

$$\begin{aligned} \dot{V} =& \sum_{i\in\mathcal{V}} \{-\lambda_i\|\dot{q}_i\|^2 - \hat{c}_i\|\dot{q}_i\|^2\|q_i\| - \dot{q}_i^\top f_i(q_i, \dot{q}_i) + \widetilde{c}_i\|\dot{q}_i\|^2\|q_i\| \\ \leq& \sum_{i\in\mathcal{V}} \{-\lambda_i\|\dot{q}_i\|^2 - (\hat{c}_i - c_i - \widetilde{c}_i)\|\dot{q}_i\|^2\|q_i\| \end{aligned} \tag{6.19}$$

where we have used the property $\|f_i(q_i, \dot{q}_i)\| \leq c_i\|q_i\|\|\dot{q}_i\|$. Since $\widetilde{c}_i = \hat{c}_i - c_i$, we obtain $\dot{V} \leq -\sum_{i\in\mathcal{V}} \lambda_i\|\dot{q}_i\|^2$, which implies that V is non-increasing along the trajectories of the closed loop system. Hence, we conclude that $V(t) \leq V_0 \leq M$, as well as the boundedness of $\widetilde{c}_i, \varphi_i, \dot{q}_i$ and hence of $\hat{c}_i, \forall i \in \mathcal{V}, t \geq t_0$. Therefore, we conclude that $\beta_i(q(t)) > 0, \forall t \geq t_0, i \in \mathcal{V}$.

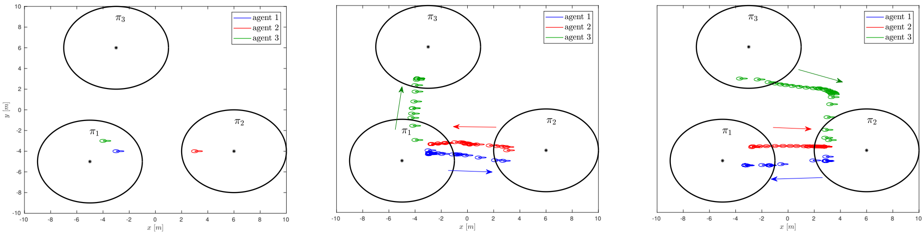
Hence, inter-agent collisions, collision with the undesired regions and the obstacle boundary, connectivity losses between the subsets of the initially connected agents and singularity configurations are avoided.

Moreover, by invoking LaSalle's Invariance Principle, the system converges to the largest invariant set contained in

$$S = \{(q, \dot{q}) \in \mathbb{D}_1 \times \cdots \times \mathbb{D}_N \times \mathbb{R}^{Nn} \text{ s.t. } \dot{q} = 0_{Nn\times 1}\}. \tag{6.20}$$

For S to be invariant, we require that $\ddot{q}_i = 0_{n\times 1}, \forall i \in \mathcal{V}$, and thus we conclude for the closed loop system (6.17) that $\nabla_{q_i}\varphi_i(q) = 0_{n\times 1}, \forall i \in \mathcal{V}$, since $\|f_i(q_i, 0_{n\times 1})\| \leq 0, \forall q_i \in \mathbb{R}^n$, in view of Assumption 6.1. Therefore, by invoking the properties of $\varphi_i(q)$, each agent $i \in \mathcal{V}$ will converge to a critical point of φ_i, i.e., all the configurations where $\nabla_{q_i}\varphi_i(q) = 0_{n\times 1}, \forall i \in \mathcal{V}$. However, due to properties of $\varphi_i(q)$, the initial conditions that lead to configurations $\widetilde{q}_i$ such that $\nabla_{q_i}\varphi_i(q)|_{q_i=\widetilde{q}_i} = 0_{n\times 1}$ and $\gamma_i(\widetilde{q}_i) \neq 0$ are sets of measure zero in the configuration space [35]. Hence, the agents will converge to the configurations where $\gamma_i(q_i) = 0$ from almost all initial conditions, i.e., $\lim_{t\to\infty} \gamma_i(q_i(t)) = 0$. Therefore, since $r_{k'} - \|p_{B_i}(q_{k'_i}) - p_{k'_i}\| \leq \bar{d}_{B_i} - \varepsilon$, it can be concluded that there exists a finite time instance t_{f_i} such that $\mathcal{A}_i(q_i(t_{f_i})) \in \pi_{k'}$, $\forall i \in \mathcal{V}$ and hence, each agent i will be at its goal region $p_{k'_i}$ at time $t_{f_i}, \forall i \in \mathcal{V}$. In addition, the boundedness of $q_i, \dot{q}_i$ implies the boundedness of the adaptation laws $\dot{\hat{c}}_i, \forall i \in \mathcal{V}$. Hence, the control laws (6.15) are also bounded.

□



(a) **(b)** **(c)**

Figure 6.12: (a): The initial position of the agents in the workspace of the simulation example. (b): The first transition of the agents in the workspace. Agent 1 transits from π_1 to π_2, agent 2 from π_2 to π_1, and agent 3 from π_1 to π_3. (c): The second transition of the agents in the workspace. Agent 1 transits from π_2 to π_1, agent 2 from π_1 to π_2, and agent 3 from π_3 to π_2.

Remark 6.3. Note that the design of the obstacle functions (6.14) renders the control laws (6.15) decentralized, in the sense that each agent uses only local information with respect to its neighboring agents, according to its limited sensing radius. Each agent can obtain the necessary information to cancel the term $\sum_{j\in\mathcal{N}_i(q_i)} \nabla_{q_i}\varphi_j(q)$ from its neighboring agents.

Finally, note that the considered dynamic model (6.10) applies for more general manipulation robots (e.g. underwater or aerial manipulators), without limiting the proposed methodology to mobile ones.

Hybrid Control Framework

Due to the proposed continuous control protocol of Section 6.4.2, the transitions $(\pi_{k_i}, t_0) \xrightarrow{i} (\pi_{k'_i}, t_{f_i})$ of Problem 6.2 are well-defined, according to Def. 6.7. Moreover, since all the agents $i \in \mathcal{V}$ remain connected with the subset of their initial neighbors $\widetilde{\mathcal{V}}_i$ and there exist finite constants t_{f_i}, such that $\mathcal{A}_i(q_i(t_{f_i})) \in \pi_{k'_i}, \forall i \in \mathcal{V}$, all the agents are aware of their neighbors state, when a transition is performed. Hence, the transition system (6.12) is well defined, $\forall i \in \mathcal{V}$. Consider, therefore, that $\mathcal{A}_i(q_i(0)) \in \pi_{k_{i,0}}, k_{i,0} \in \mathcal{K}, \forall i \in \mathcal{V}$, as well as a given desired path for each agent, that does not violate the connectivity condition of Problem 6.2. Then, the iterative application of the control protocol (6.15) for each transition of the desired path of agent i guarantees the successful execution of the desired paths, with all the closed loop signals being bounded.

Remark 6.4. Note that, according to the aforementioned analysis, we implicitly assume that the agents start executing their respective transitions at the same time (we do not take into account individual control jumps in the Lyapunov analysis, i.e., it is valid only for one transition). Intuition suggests that if the regions of interest are sufficiently far from each other, then the agents will be able to perform the sequence of their transitions independently. Detailed technical analysis of such cases is part of our future goals.

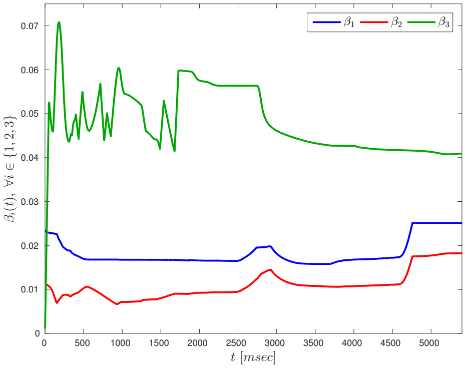


Figure 6.13: The obstacle functions $\beta_i, i \in \{1, 2, 3\}$, which remain strictly positive.

6.4.3 Simulation Results

To demonstrate the validity of the proposed methodology, we consider the simplified example of three agents in a workspace with $r_0 = 12$ and three regions of interest, with $r_k = 4, \forall k \in \{1, 2, 3\}$ m. Each agent consists of a mobile base and a rigid link connected with a rotational joint, with $\bar{d}_{B_i} = 1$m, $\forall i \in \{1, 2, 3\}$. We also choose $p_1 = [-5, -5]$m, $p_2 = [6, -4]$m, $p_3 = [-3, 6]$m. The initial base positions are taken as $p_{B_1} = [-3, -4]^\top$m$, p_{B_2} = [3, -4]^\top$m$, p_{B_3} = [-4, -5]^\top$m with $\bar{d}_{B_i} = 1.25$m$, \forall i \in \{1, 2, 3\}$, which imply that $\mathcal{A}_1(q_1(0)), \mathcal{A}_3(q_3(0)) \in \pi_1$ and $\mathcal{A}_2(q_2(0)) \in \pi_2$ (see Fig. 6.12(a). The control inputs for the agents are the 2D force acting on the mobile base, and the joint torque of the link. We also consider a sensing radius of $d_{\text{con}_i} = 8$m and the subsets of initial neighbors as $\widetilde{\mathcal{N}}_1 = \{2\}, \widetilde{\mathcal{N}}_2 = \{1, 3\}$, and $\widetilde{\mathcal{N}}_3 = \{2\}$, i.e., agent 1 has to stay connected with agent 2, agent 2 has to stay connected with agents 1 and 3 and agent 3 has to stay connected with agent 2. The agents are required to perform two transitions. Regarding the first transition, we choose $\pi_{k'_1} = \pi_2$ for agent 1, $\pi_{k'_2} = \pi_1$ for agent 2, and $\pi_{k'_3} = \pi_3$, for agent 3. Regarding the second transition, we choose $\pi_{k'_1} = \pi_1, \pi_{k'_2} = \pi_2$, and $\pi_{k'_3} = \pi_2$. The control parameters and gains where chosen as $k_i = 5, \lambda_i = 10, \rho_i = 1$, and $\sigma_i = 0.01, \forall i \in \{1, 2, 3\}$. We employed the potential field from [37]. The simulation results are depicted in Fig. 6.12-6.15. In particular, Fig. 6.12(b) and 6.12(c) illustrate the two consecutive transitions of the agents. Fig. 6.13 depicts the obstacle functions β_i which are strictly positive, $\forall i \in \{1, 2, 3\}$. Finally, the control inputs are given in Fig. 6.14 and the parameter errors $\widetilde{c}$ are shown in Fig. 6.15, which indicates their boundedness. As proven in the theoretical analysis, the transitions are successfully performed while satisfying all the desired specifications.

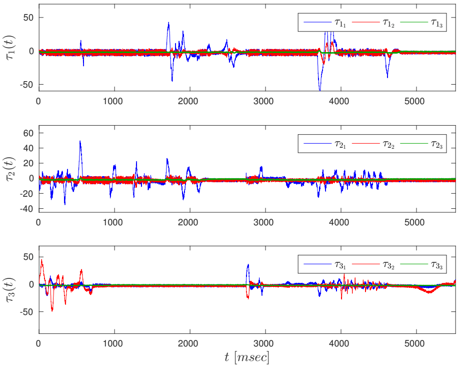


Figure 6.14: The resulting control inputs $\tau_i, \forall i \in \{1,2,3\}$ for the two transitions.

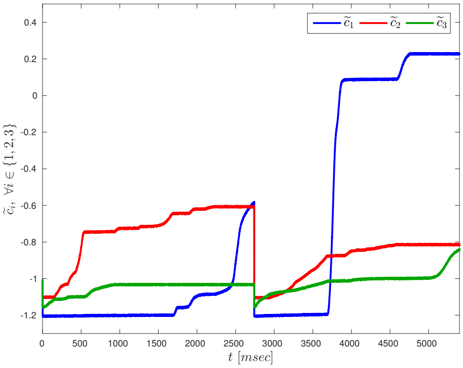


Figure 6.15: The parameter deviations $\widetilde{c}_i, \forall i \in \{1,2,3\}$, which are shown to be bounded.

6.5 Conclusion and Future Work

In this work, we proposed hybrid control strategies for the motion planning of teams of aerial vehicles as well as mobile manipulators under LTL specifications. By using decentralized navigation functions that guarantee inter-agent collision avoidance as well as connectivity maintenance, we abstracted each agent's motion as a finite transition system between regions of interest. Each agent then can derive the plan that satisfies its given LTL formula through formal-verification techniques. Simulation studies and experimental results verified the validity of the proposed algorithms. Future efforts will be devoted towards addressing abstractions of cooperative tasks between the agents by employing hybrid control techniques as well as abstraction reconfiguration due to potential execution incapability of the transitions and partially known environments.

Chapter 7

Abstractions for Multi-Agent Cooperative-Manipulation Schemes

This chapter addresses the problem of deriving well-defined abstractions for the motion planning of a team of robotic agents and objects. In particular, we propose two methodologies for the discrete abstraction of such systems. Firstly, we consider the trajectory tracking of a cooperatively manipulated object without necessitating feedback of the contact forces/torques or inter-agent communication. By employing the prescribed performance control methodology, we pre-determine the transient and steady state of the coupled object-agents system. The latter, along with a region partition of the workspace that depends on the physical volume of the object and the agents, allows us to define timed transitions for the coupled system among the derived workspace regions. Therefore, we abstract its motion as a finite transition system and, by employing standard automata-based methodologies, we define high level complex tasks for the object that can be encoded by timed temporal logics. Secondly, we present a hybrid control framework for the motion planning of a multi-agent system including N robotic agents and M objects, under high level goals expressed as Linear Temporal Logic (LTL) formulas. We design control protocols that allow the transition of the agents as well as the cooperative transportation of the objects by the agents, among predefined regions of interest in the workspace. This allows to abstract the coupled behavior of the agents and the objects as a finite transition system and to design a high-level multi-agent plan that satisfies the agents' and the objects' specifications, given as temporal logic formulas. Simulation results verify the validity of the proposed frameworks.

7.1 Introduction

As pointed out in the previous chapter, temporal-logic based motion planning has gained significant amount of attention over the last decade, since it provides a fully automated correct-by-design controller synthesis approach for autonomous robots. Temporal logics, such as linear temporal logic (LTL), provide formal high-level languages that can describe planning objectives more complex than the well-studied navigation algorithms, and have been used extensively both in single- as well as in multi-agent setups. The objectives are given as a temporal logic formula with respect to a discretized abstraction of the system (usually a finite transition system), and then, a high-level discrete path is found by off-the-shelf model-checking algorithms, given the abstracted system and the task specification [38].

Most works in the related literature consider temporal logic-based motion planning for fully actuated, autonomous agents. Consider, however, cases where some unactuated objects must undergo a series of processes in a workspace with autonomous agents (e.g., car factories). In such cases, the agents, except for satisfying their own motion specifications, are also responsible for coordinating with each other in order to transport the objects around the workspace. When the unactuated objects' specifications are expressed using temporal logics, then the abstraction of the agents' behavior becomes much more complex, since it has to take into account the objects' goals.

In contrast to the related literature, which mainly considers the trajectory tracking of manipulated objects, we are here interested in complex tasks, possibly including time, such as "never take the object to dangerous regions" or "keep moving the object from region A to B within a predefined time interval" which must be executed via the control actions of the agents. Such tasks can be expressed by temporal logic languages. Except for Linear Temporal Logic (LTL), which is the most common language that has been incorporated in the multi-agent motion planning problem, Metric and Metric Interval Temporal Logic (MTL, MITL) [39, 40, 171], as well as Time Window Temporal Logic (TWTL) are languages that encode time specifications and were used for multi-agent motion planning in [151, 172, 173]. As highlighted in the previous chapter, in order to be able to define temporal logic objectives, the continuous-time system must be abstracted to a higher-level discrete representation that incorporates the motion and the actions of the system.

The previous chapters focused on the cooperative manipulation of an object by multiple robotic agents, as well as the multi-agent discrete abstraction and control synthesis for the satisfaction of temporal logic formulas. In this chapter, we combine these results to address the discrete abstraction for cooperative manipulation schemes, from two difference aspects. Firstly, we study the *timed* abstraction of a system comprising of N robotic agents rigidly grasping an object. By using the prescribed performance control methodology of Chapter 4, we design a distributed model-free control protocol for the trajectory tracking of the cooperatively manipulated object that allows us to model the motion of the coupled object-agents system as a weighted finite transition system. Then, by employing formal verification-based methodologies, we derive a path that satisfies a given MITL task. Secondly, we present a novel hybrid control framework for the motion planning of a team of N autonomous agents and M unactuated objects under LTL specifications. Using previous results on navigation functions (similarly to Chapter 6), we design feedback control laws for i) the navigation of the agents and ii) the cooperative transportation of the objects by the agents, among predefined regions of interest in the workspace, while ensuring

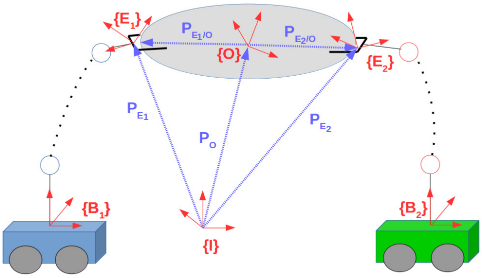


Figure 7.1: Two robotic arms rigidly grasping an object.

inter-agent collision avoidance. This allows us to model the coupled behavior of the agents and the objects with a finite transition system, which can be used for the design of high-level plans that satisfy the given LTL specifications. Simulation results verify the validity of the proposed frameworks.

The rest of the chapter consists of two main parts: Firstly, Section 7.2 presents the timed abstraction of a cooperative manipulated object, with 7.2.1 and 7.2.2 describing the problem formulation and the proposed solution and 7.2.3 providing simulation results. Secondly, Section 7.3 presents the abstraction of the multi-agent-object system, with 7.3.1, 7.3.2, and 7.3.3 providing the problem formulation, the proposed solution, and the simulation results, respectively. Finally, Section 7.4 concludes the chapter.

7.2 Timed Abstractions for Distributed Cooperative Manipulation

7.2.1 Problem Formulation

Consider a bounded workspace $\mathcal{W} \subset \mathbb{R}^3$ containing N robotic agents rigidly grasping an object, as shown in Fig. 7.1. The agents are considered to be fully actuated and they consist of a base that is able to move around the workspace (e.g., mobile or aerial vehicle) and a robotic arm. The reference frames corresponding to the i-th end-effector and the object's center of mass are denoted with $\{E_i\}$ and $\{O\}$, respectively, whereas $\{I\}$ corresponds to an inertial reference frame. The rigidity of the grasps implies that the agents can exert any force/torque along every direction to the object. We consider that each agent i knows only its own state, position and velocity, as well as its own and the object's geometric parameters. More specifically, we assume that each agent i knows the distance from its grasping point $\{E_i\}$ to the object's center of mass $\{O\}$ as well as the relative orientation offset between the two frames $\{E_i\}$ and $\{O\}$. This information can be either retrieved on-line via appropriate sensors or transmitted off-line to the agents, without the need of inter-agent on-line communication. Finally, no interaction force/torque measurements are required and the dynamic model of the object and the agents is considered unknown.

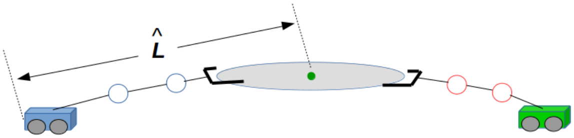


Figure 7.2: An example of the system shown in Fig. 7.1 in the configuration that produces $\hat{L}$.

System model

The modeling of the object and the agents is identical to the one of Chapter 5 and its derivation is omitted. The joint space variables of agent $i \in \mathcal{N}$ is $q_i \in \mathbb{R}^{n_i}$, with $n_i = n_{\alpha_i} + 6$, $q_i = [p_{B_i}^T, \eta_{B_i}^T, \alpha_i^T]^T$. Each agent $i \in \mathcal{N} = \{1, \dots, N\}$ has access to its own state q_i as well as $\dot{p}_{B_i}^{B_i}, \omega_{B_i}^{B_i}$, and $\dot{\alpha}_i$ via on-board sensors, and it can obtain $\dot{p}_{B_i}$, ω_{B_i} via the transformation given in Chapter 5. The task-space dynamics of the agents is given by

$$M_i(q_i)\dot{v}_i + C_i(q_i, \dot{q}_i)v_i + g_i(q_i) + f_i(q_i, \dot{q}_i) + w_i(q_i, t) = u_i - \lambda_i, \tag{7.1}$$

where we use here $f_i(\cdot)$ for modeling uncertainties and $w_i(\cdot)$ for external disturbances, which are assumed to be continuous in $q_i, \dot{q}_i$, and bounded in t. Moreover, as noted in Chapter 5, the matrix $J_{B_i}(\eta_{B_i})$, which maps $\dot{\eta}_{B_i}$ to ω_{B_i}, and the agent Jacobian matrix J_i become singular when $\theta_{B_i} = \pm\frac{\pi}{2}$ and at kinematic singularities, respectively. We assume here that the agents do not operate close to such points.

The object dynamic equations are given by

$$\dot{x}_O = J_O(\eta_O)v_O \tag{7.2a}$$
$$M_O(x_O)\dot{v}_O + C_O(x_O, v_O)v_O + g_O(x_O) + w_O(t) = \lambda_O, \tag{7.2b}$$

with the matrix $J_O(\cdot)$ and the dynamic terms as in Chapters 4, 5, and $w_{_O}(t)$ a bounded vector field representing external disturbances. The matrix $J_O(\eta_O)$ is not defined when $\theta_O = \pm\frac{\pi}{2}$, which, however, is guaranteed to be avoided by the control design. The coupled dynamics between the object and the agents is given by

$$\widetilde{M}(q)\dot{v}_O + \widetilde{C}(q, \dot{q})v_O + \widetilde{h}(q, \dot{q}) + \widetilde{w}(q, t) = G^T(q)\bar{u}, \tag{7.3}$$

with the coupled dynamic terms as in Chapter 5. We assume again here that the geometric object parameters are known and therefore each agent can compute p_O, η_O and v_O by the coupled kinematic equations, without employing any sensory data.

Workspace Partition

As mentioned in Section 7.1, we are interested in designing a well-defined abstraction of the coupled object-agents system, so that we can define MITL formulas over certain properties in a discrete set of regions of the workspace. Therefore, we provide now a partition of $\mathcal{W}$ into cell regions. We denote by $\mathcal{S}_q$ the set that consists of all points

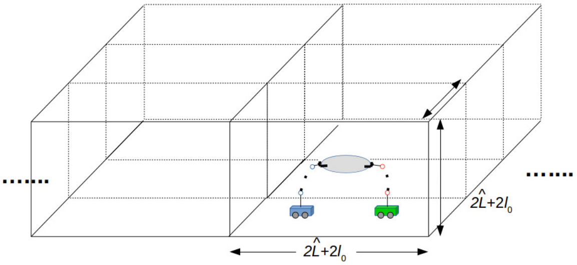


Figure 7.3: The workspace partition according to the bounding box of the coupled system.

$p_s \in \mathcal{W}$ that physically belong to the coupled system, i.e., they consist part of either the volume of the agents or the volume of the object. Note that these points depend on the actual value of q. We further define the constant $\hat{L} \geq \sup_{\substack{q\in\mathbb{R}^n \\ p_s\in\mathcal{S}_q}} d(p_s, p_O(q))$, where, with a slight abuse of notation and in view of the coupled object-agents kinematics and the forward kinematics of the agents, we express p_O as a function of q. Note that, although the explicit computation of $\mathcal{S}_q$ may not be possible, $\hat{L}$ is an upper bound of the maximum distance between the object center of mass and a point in the coupled system's volume over all possible configurations q, and thus, it can be measured. For instance, Fig. 7.2 shows $\hat{L}$ for the system of Fig. 7.1. It is straightforward to conclude that

$$p_s \in \mathcal{B}(p_O(q), \hat{L}), \forall p_s \in \mathcal{S}_q, q \in \mathbb{R}^n. \tag{7.4}$$

Next, we partition the workspace $\mathcal{W}$ into R equally sized rectangular regions $\Pi = \{\pi_1, \dots, \pi_R\}$, whose geometric centers are denoted as $p^c_{\pi_j} \in \mathcal{W}, j \in \{1, \dots, R\}$. The length of the region sides is set to $D = 2\hat{L}+2l_0$, where l_0 is an arbitrary positive constant. Hence, each region π_j can be formally defined as follows:

$$\pi_j = \{p \in \mathcal{W} \text{ s.t. } (p)_k \in [(p^c_{\pi_j})_k - \hat{L} - l_0, (p^c_{\pi_j})_k + \hat{L} + l_0), \forall k \in \{x, y, z\}\},$$

with $d(p^c_{\pi_{j+1}}, p^c_{\pi_j}) = (2\hat{L} + 2l_0), \forall j \in \{1, \dots, R-1\}$, and $(p^c_{\pi_j})_z = \hat{L} + l_0, \forall j \in \{1, \dots, R\}$; $(a)_k, k \in \{x, y, z\}$, denotes the k-th coordinate of $a = [(a)_x, (a)_y, (a)_z]^T \in \mathbb{R}^3$. An illustration of the aforementioned partition is depicted in Fig. 7.3.

Note that each π_j is a uniformly bounded and convex set and also $\pi_j \cap \pi_{j'} = \emptyset, \forall j, j' \in \{1, \dots, R\}$ with $j \neq j'$. We also define the neighborhood $\mathcal{D}$ of region π_j as the set of its adjacent regions, i.e., $\mathcal{D}(\pi_j) = \{\pi_{j'} \in \Pi \text{ s.t. } d(p^c_{\pi_j}, p^c_{\pi_{j'}}) = (2\hat{L} + 2l_0)\}$, which is symmetric, i.e., $\pi_{j'} \in \mathcal{D}(\pi_j) \Leftrightarrow \pi_j \in \mathcal{D}(\pi_{j'})$.

To proceed we need the following definitions regarding the timed transition of the coupled system between two regions $\pi_j, \pi_{j'}$:

Definition 7.1. The coupled object-agents system is in region π_j at a configuration q, denoted as $\mathcal{A}(q) \in \pi_j$, if and only if the following hold:

1. $p_s \in \pi_j, \forall p_s \in \mathcal{S}_q$
2. $d(p_O(q), p^c_{\pi_j}) < l_0$.

Definition 7.2. Assume that $\mathcal{A}(q(t_0)) \in \pi_j, j \in \{1, \dots, R\}$, for some $t_0 \in \mathbb{R}_{\geq 0}$. Then, there exists a transition for the coupled object-agents system from π_j to $\pi_{j'}, j' \in \{1, \dots, R\}$ with time duration $\delta t_{j,j'} \in \mathbb{R}_{\geq 0}$, denoted as $\pi_j \xrightarrow{\mathcal{T}} \pi_{j'}$, if and only if there exists a bounded control trajectory $\bar{u}$ in (7.3), such that the following hold:

1. $\mathcal{A}(q(t_0 + \delta t_{j,j'})) \in \pi_{j'}$,
2. $p_s \in \pi_j \cup \pi_{j'}, \forall p_s \in \mathcal{S}_q, t \in [t_0, t_0 + \delta t_{j,j'}]$.

Note that the entire system object-agents must remain in $\pi_j, \pi_{j'}$ during the transition and therefore the requirement $\pi_{j'} \in \mathcal{D}(\pi_j)$ is implicit in Definition 7.2.

Specification

Given the workspace partition, we can introduce a set of atomic propositions $\mathcal{AP}$ for the object, which are expressed as Boolean variables that correspond to properties of interest in the regions of the workspace (e.g., “Obstacle region”, “Goal region”). Formally, the labeling function $\mathcal{L} : \Pi \to 2^{\mathcal{AP}}$ assigns to each region π_j the subset of the atomic propositions $\mathcal{AP}$ that are true in π_j.

Definition 7.3. Given a time trajectory $q(t), t \geq 0$, a *timed sequence* of q is the infinite sequence $\beta = (q(t_1), t_1)(q(t_2), t_2) \dots$, with $t_m \in \mathbb{R}_{\geq 0}, t_{m+1} > t_m$ and $\mathcal{A}(q(t_m)) \in \pi_{j_m}, j_m \in \{1, \dots, R\}, \forall m \in \mathbb{N}$. The *timed behavior* of β is the infinite sequence $\sigma_\beta = (\sigma_1, t_1)(\sigma_2, t_2) \dots$, with $\sigma_m \in 2^{\mathcal{AP}}, \sigma_m \in \mathcal{L}(\pi_{j_m})$ for $\mathcal{A}(q(t_m)) \in \pi_{j_m}, j_m \in \{1, \dots, R\}$, $\forall m \in \mathbb{N}$, i.e., the set of atomic propositions that are true when $\mathcal{A}(q(t_m)) \in \pi_{j_m}$.

Definition 7.4. The timed run β satisfies an MITL formula ϕ if and only if $\sigma_\beta \models \phi$.

We are now ready to state the problem treated in this section.

Problem 7.1. Given N agents rigidly grasping an object in $\mathcal{W}$ subject to the coupled dynamics (7.3), the workspace partition Π such that $\mathcal{A}(q(0)) \in \pi_{j_0}, j_0 \in \{1, \dots, R\}$, a MITL formula ϕ over AP and the labeling function $\mathcal{L}$, derive a control strategy that achieves a timed sequence β which yields the satisfaction of ϕ.

7.2.2 Main Results

Control Design

The first ingredient of the proposed solution is the design of a decentralized control protocol $\bar{u}$ such that a transition relation between two adjacent regions according to

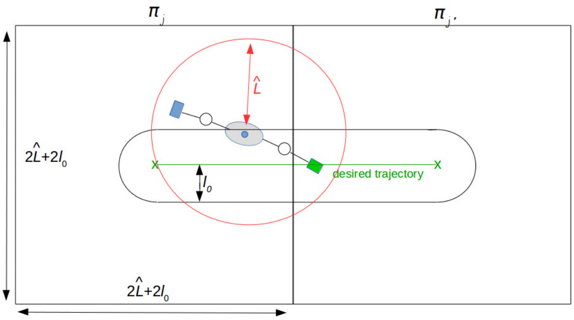


Figure 7.4: Top view of a transition between two adjacent regions π_j and $\pi_{j'}$. Since $p_O \in \mathcal{B}(p_{j,j'}(t), l_0)$, we conclude that $p_s \in \mathcal{B}(p_O, \hat{L}) \subset \mathcal{B}(p_{j,j'}(t), l_0 + \hat{L}) \subset \pi_j \cup \pi_{j'}$.

Definition 7.2 is established. Assume, therefore, that $\mathcal{A}(q(t_0)) \in \pi_j, j \in \{1, \dots, R\}$ for some $t_0 \in \mathbb{R}_{\geq 0}$. We aim to find a bounded $\bar{u}$, such that $\mathcal{A}(q(t_0 + \delta t_{j,j'})) \in \pi_{j'}$, with $\pi_{j'} \in \mathcal{D}(\pi_j)$, and $p_s \in \pi_j \cup \pi_{j'}, \forall p_s \in \mathcal{S}_q, t \in [t_0, t_0 + \delta t_{j,j'}]$, for a predefined arbitrary constant $\delta t_{j,j'} \in \mathbb{R}_{\geq 0}$ corresponding to the transition $\pi_j \xrightarrow{\mathcal{T}} \pi_{j'}$.

The first step is to associate to the transition a smooth and bounded trajectory with bounded time derivative, defined by the line segment that connects $p^c_{\pi_j}$ and $p^c_{\pi_{j'}}$, i.e. define $p_{j,j'} : [t_0, \infty) \to \mathbb{R}^3$, such that $p_{j,j'}(t_0) = p^c_{\pi_j}, p_{j,j'}(t) = p^c_{\pi_{j'}}, \forall t \geq t_0 + \delta t_{j,j'}$ and

$$\mathcal{B}(p_{j,j'}(t), \hat{L} + l_0) \subset \pi_j \cup \pi_{j'}, \quad \forall t \geq t_0. \tag{7.5}$$

An example of $p_{j,j'}$ is

$$p_{j,j'}(t) = \begin{cases} \dfrac{p^c_{\pi_{j'}} - p^c_{\pi_j}}{\delta t_{j,j'}} t + \dfrac{p^c_{\pi_j}(\delta t_{j,j'} - 1) - p^c_{\pi_{j'}}}{\delta t_{j,j'}} t_0, & t \in T_1 \\ p^c_{\pi_{j'}}, & t \in T_2 \end{cases}, \tag{7.6}$$

where $T_1 = [t_0, t_0 + \delta t_{j,j'}), T_2 = [t_0 + \delta t_{j,j'}, \infty)$. The intuition behind the solution of Problem 7.1 via the definition of $p_{j,j'}$ is the following: if we guarantee that the object's center of mass stays l_0-close to $p_{j,j'}$, i.e., $d(p_O(t), p_{j,j'}(t)) < l_0, \forall t \geq t_0$, then $d(p_O(t_0 + \delta t_{j,j'}), p^c_{\pi_{j'}}) < l_0$ and, by invoking (7.4) and (7.5), we obtain $p_s \in \mathcal{B}(p_O(t), \hat{L}) \subset \mathcal{B}(p_{j,j'}(t), \hat{L} + l_0) \subset \pi_j \cup \pi_{j'}, \forall p_s \in \mathcal{S}_q, t \geq t_0$ (and therefore $t \in [t_0, t_0 + \delta t_{j,j'}]$), and thus the requirements of Definition 7.2 for the transition relation are met. Fig. 7.4 illustrates the aforementioned reasoning.

Along with $p_{j,j'}$, we consider that the object has to comply with certain specifications associated with its orientation. Therefore, we also define a smooth and bounded orientation trajectory $\eta_{j,j'} = [\phi_{j,j'}, \theta_{j,j'}, \psi_{j,j'}]^T : [t_0, \infty) \to \mathbb{T}$ with bounded time derivative, that has to be tracked by the object's center of mass. We choose $\theta_{j,j'}(t) \in [-\theta^*, \theta^*] \subset (-\frac{\pi}{2}, \frac{\pi}{2}), \forall t \in \mathbb{R}_{\geq 0}$, with $\theta^* \in (0, \frac{\pi}{2})$, so as to ensure the singularity avoidance of $J_{O_r}(x_O)$. We form, therefore, the desired pose trajectory $x_{j,j'} : [t_0, \infty) \to \mathbb{M}$, with $x_{j,j'}(t) = [p^T_{j,j'}(t), \eta^T_{j,j'}(t)]^T$. In case of multiple consecutive transitions $\ldots \pi_h \xrightarrow{\mathcal{T}} \pi_j \xrightarrow{\mathcal{T}} \pi_{j'} \xrightarrow{\mathcal{T}} \pi_{h'} \ldots$ over the intervals $\ldots$,$\delta t_{h,j}$, $\delta t_{j,j'}$, $\delta t_{j',h'}$,$\ldots$, the desired orientation trajectories $\ldots$, $\eta_{h,j}(t)$, $\eta_{j,j'}(t)$, $\eta_{j',h'}(t)$, $\ldots$ must be continuous at the transition points, i.e., $\eta_{h,j}(t_0) = \eta_{j,j'}(t_0)$ and $\eta_{j,j'}(t_0 + \delta t_{j,j'})$ $= \eta_{j',h'}(t_0 + \delta t_{j,j'})$.

Therefore, Problem 7.1 is equivalent to a problem of trajectory tracking within certain bounds. Finally, we make the following assumption:

Assumption 7.1. The configuration of the object at $t = t_0$ does not result in a singular $J_{O_r}(x_O(t_0))$, i.e., $\theta_O(t_0) \in (-\frac{\pi}{2}, \frac{\pi}{2})$ and $\theta_{j,j'}(t)$ is chosen such that

$$-\tfrac{\pi}{2} + \theta^* < \theta_O(t_0) - \theta_{j,j'}(t_0) < \tfrac{\pi}{2} - \theta^*.$$

We can now define the associated position and orientation errors $e_p = [e_{p_1}, e_{p_2}, e_{p_3}]$ $\in \mathbb{R}^3$, $e_\eta = [e_{\eta_\phi}, e_{\eta_\theta}, e_{\eta_\psi}] \in \mathbb{T}$ as follows:

$$e_p = p_O - p_{j,j'}(t), \tag{7.7a}$$

$$e_\eta = \eta_O - \eta_{j,j'}(t), \tag{7.7b}$$

$\forall t \in [t_0, \infty)$.

A suitable methodology for the control design in hand is that of prescribed performance control, which is adapted in this work in order to achieve predefined transient and steady state response bounds for the pose errors. Following Section 2.1 and the already introduced control design of Chapter 4, prescribed performance characterizes the behavior where the aforementioned errors evolve strictly within a predefined region that is bounded by absolutely decaying functions of time, called performance functions. The mathematical expressions of prescribed performance are given by the inequalities:

$$-\rho_{p_k}(t) < e_{p_k}(t) < \rho_{p_k}(t), \quad \forall k \in \{1, 2, 3\}, \tag{7.8a}$$

$$-\rho_{\eta_\ell}(t) < e_{\eta_\ell}(t) < \rho_{\eta_\ell}(t), \quad \forall \ell \in \{\phi, \eta, \psi\}, \tag{7.8b}$$

$\forall t \in [t_0, \infty)$, where $\rho_{p_k}, \rho_{\eta_l} : [t_0, \infty) \to \mathbb{R}_{>0}$ with

$$\rho_{p_k}(t) = (\rho^0_{p_k} - \rho^\infty_{p_k}) e^{-l_{p_k}(t-t_0)} + \rho^\infty_{p_k}, \quad \forall k \in \{1, 2, 3\}, \tag{7.9a}$$

$$\rho_{\eta_\ell}(t) = (\rho^0_{\eta_\ell} - \rho^\infty_{\eta_\ell}) e^{-l_{\eta_\ell}(t-t_0)} + \rho^\infty_{\eta_\ell}, \quad \forall \ell \in \{\phi, \theta, \psi\}, \tag{7.9b}$$

are designer-specified, smooth, bounded and decreasing positive functions of time with $l_{p_k}, l_{\eta_\ell}, \rho^\infty_{p_k}, \rho^\infty_{\eta_\ell}, k \in \{1, 2, 3\}$, $\ell \in \{\phi, \theta, \psi\}$ positive parameters incorporating the desired transient and steady state performance respectively, as described in Section 2.1.

Next, we propose a state feedback control protocol that does not incorporate any information on the agents' or the object's dynamics or the external disturbances and guarantees (7.8) for all $t \in [t_0, \infty)$ and hence $[t_0, t_0 + \delta t_{j,j'}]$, which, by appropriately selecting $\rho_{p_k}(t), \rho_{\eta_\ell}(t), k \in \{1, 2, 3\}, \ell \in \{\phi, \theta,$
$\psi\}$ and given that $\mathcal{A}(q(t_0)) \in \pi_j$, guarantees a representation singularity-free (i.e.,

$\theta_O(t) \neq \frac{\pi}{2}, t \in [t_0, \infty)$) transition $\pi_j \xrightarrow{\mathcal{T}} \pi_{j'}$ with time duration of $\delta t_{j,j'}$, as will be clarified in the sequel.

Define first the stack pose error $e_s :\in \mathbb{M}$:

$$e_s = \begin{bmatrix} e_{s_1} \\ e_{s_2} \\ e_{s_3} \\ e_{s_4} \\ e_{s_5} \\ e_{s_6} \end{bmatrix} = \begin{bmatrix} e_{p_1} \\ e_{p_2} \\ e_{p_3} \\ e_{\eta_\phi} \\ e_{\eta_\theta} \\ e_{\eta_\psi} \end{bmatrix} = x_O - x_{j,j'}(t), \tag{7.10}$$

$\forall t \in [t_0, \infty)$, and perform the following steps:

Step I-a. Select the corresponding functions $\rho_{p_k}, \rho_{\eta_\ell}$ as in (7.9) with

1. $\rho^0_{p_k} = \rho_{p_k}(t_0) = l_0,\ \forall k \in \{1,2,3\},\ \ \rho^0_{\eta_\theta} = \rho_{\eta_\theta}(t_0) = \dfrac{\pi}{2} - \theta^*,\ \ \rho^0_{\eta_\ell} = \rho_{\eta_\ell}(t_0) > |e_{\eta_\ell}(t_0)|,\ \forall \ell \in \{\phi, \psi\}$,

2. $l_{p_k}, l_{\eta_\ell} \in \mathbb{R}_{>0}, \forall k \in \{1,2,3\}, \ell \in \{\phi, \theta, \psi\}$,

3. $\rho^\infty_{p_k} \in (0, \rho^0_{p_k}),\ \ \rho^\infty_{\eta_\ell} \in (0, \rho^0_{\eta_\ell}),\ \forall k \in \{1,2,3\},\ \ell \in \{\phi, \theta, \psi\}$.

Step I-b. Define the normalized errors $\xi_s \in \mathbb{R}^6$:

$$\xi_s = \begin{bmatrix} \xi_{s_1} \\ \vdots \\ \xi_{s_6} \end{bmatrix} = \rho_s^{-1}(t) e_s, \tag{7.11}$$

where $\rho_s(t) = \text{diag}\{\rho_{p_1}(t), \rho_{p_2}(t), \rho_{p_3}(t), \rho_{\eta_\phi}(t), \rho_{\eta_\theta}(t), \rho_{\eta_\psi}(t)\} \in \mathbb{R}^{6\times 6}$, and design the reference velocity vector $v_{O,\text{des}} : (-1,1)^6 \times [t_0, \infty) \to \mathbb{R}^6$, with:

$$\begin{aligned} v_{O,\text{des}}(\xi_s, t) = \begin{bmatrix} \dot{p}_{O,\text{des}}(\xi_s, t) \\ \omega_{O,\text{des}}(\xi_s, t) \end{bmatrix} &= -g_s J_{O_r}(x_O) \rho_s^{-1}(t) r_s(\xi_s) \varepsilon_s(\xi_s) \\ = -g_s J_{O_r}(\rho_s(t)\xi_s + x_{j,j'}(t)) \rho_s^{-1}(t) r_s(\xi_s) \varepsilon_s(\xi_s), \end{aligned} \tag{7.12}$$

where g_s is a positive scalar tunable gain and the signals $\varepsilon_s : (-1,1)^6 \to \mathbb{R}^6, r_s :$

$(-1,1)^6 \to \mathbb{R}^6$ are defined as:

$$\varepsilon_s(\xi_s) = \begin{bmatrix} \varepsilon_{s_1}(\xi_{s_1}) \\ \vdots \\ \varepsilon_{s_6}(\xi_{s_6}) \end{bmatrix} = \begin{bmatrix} \ln\left(\dfrac{1+\xi_{s_1}}{1-\xi_{s_1}}\right) \\ \vdots \\ \ln\left(\dfrac{1+\xi_{s_6}}{1-\xi_{s_6}}\right) \end{bmatrix}, \tag{7.13}$$

$$\begin{aligned} r_s(\xi_s) =& \mathrm{diag}\left\{\left[\frac{\partial \varepsilon_{s_m}(\xi_{s_m})}{\partial \xi_{s_m}}\right]_{m\in\{1,\ldots,6\}}\right\} \\ =& \mathrm{diag}\left\{\left[\frac{2}{(1-\xi_{s_m}^2)}\right]_{m\in\{1,\ldots,6\}}\right\}. \end{aligned} \tag{7.14}$$

Step II-a. Define the velocity error vector $e_v \in \mathbb{R}^6$ with

$$e_v = \begin{bmatrix} e_{v_1} \\ \vdots \\ e_{v_6} \end{bmatrix} = v_O - v_{O,\mathrm{des}}(\xi_s, t), \tag{7.15}$$

and select the corresponding positive performance functions $\rho_{v_m} : [t_0, \infty) \to \mathbb{R}_{>0}$ with $\rho_{v_m}(t) = (\rho^0_{v_m} - \rho^\infty_{v_m})e^{-l_{v_m}(t-t_0)} + \rho^\infty_{v_m}$, such that $\rho^0_{v_m} > |e_{v_m}(t_0)|, l_{v_m} > 0$ and $\rho^\infty_{v_m} \in (0, \rho^0_{v_m})$, $\forall m \in \{1, \ldots, 6\}$.

Step II-b. Define the normalized velocity errors $\xi_v \in \mathbb{R}^6$:

$$\xi_v = \begin{bmatrix} \xi_{v_1} \\ \vdots \\ \xi_{v_6} \end{bmatrix} = \rho_v^{-1}(t) e_v, \tag{7.16}$$

where $\rho_v(t) = \mathrm{diag}\{[\rho_{v_m}(t)]_{m\in\{1,\ldots,6\}}\}$, and design the distributed control protocol for each agent $i \in \mathcal{N}$ as $u_i : (-1,1)^6 \times (-1,1)^6 \times [t_0, \infty) \to \mathbb{R}^6$:

$$u_i(\xi_s, \xi_v, t) = -c_i g_v \Big(J_{O_i}^{-1}(q_i)\Big)^T \rho_v^{-1}(t) r_v(\xi_v) \varepsilon_v(\xi_v), \tag{7.17}$$

where J_{O_i} the coupled agent-to-object Jacobian, g_v is a positive scalar tunable gain, and c_i are predefined load sharing coefficients satisfying $\sum_{i\in\mathcal{N}} c_i = 1$ and $0 \le c_i \le 1, \forall i \in \mathcal{N}$. The signals $\varepsilon_v : (-1,1)^6 \to \mathbb{R}^6$ and $r_v : (-1,1)^6 \to \mathbb{R}^{6\times 6}$ are defined as:

$$\varepsilon_v(\xi_v) = \begin{bmatrix} \varepsilon_{v_1}(\xi_{v_1}) \\ \vdots \\ \varepsilon_{v_6}(\xi_{v_6}) \end{bmatrix} = \begin{bmatrix} \ln\left(\dfrac{1+\xi_{v_1}}{1-\xi_{v_1}}\right) \\ \vdots \\ \ln\left(\dfrac{1+\xi_{v_6}}{1-\xi_{v_6}}\right) \end{bmatrix}, \tag{7.18}$$

$$\begin{aligned} r_v(\xi_v) =& \mathrm{diag}\left\{ \left[\frac{\partial \varepsilon_{v_m}(\xi_{v_m})}{\partial \xi_{v_m}} \right]_{m \in \{1,\ldots,6\}} \right\} \\ =& \mathrm{diag}\left\{ \left[\frac{2}{(1-\xi_{v_m}^2)} \right]_{m \in \{1,\ldots,6\}} \right\}. \end{aligned} \tag{7.19}$$

The control law (7.17) can be written in vector form:

$$\begin{aligned} \bar{u}(\xi_s,\xi_v,t) =& \begin{bmatrix} u_1(\xi_s,\xi_v,t) \\ \vdots \\ u_N(\xi_s,\xi_v,t) \end{bmatrix} = U_j^{j'} \\ =& -C_g G^*(q) \rho_v^{-1}(t) r_v(\xi_v) \varepsilon_v(\xi_v), \end{aligned} \tag{7.20}$$

where $G^*(q)$ as in (4.42), $C_g = g_v \mathrm{diag}\{[c_i I_6]_{i \in \mathcal{N}}\} \in \mathbb{R}^{6N \times 6N}$, and the notation $U_j^{j'}$ stands for the transition from π_j to $\pi_{j'}$.

The aforementioned control protocol for the transition $\pi_j \xrightarrow{\mathcal{T}} \pi_{j'}$ is summarized in Algorithm 1.

Algorithm 1 Transition Algorithm

```
1: Compute trajectory $x_{j,j'}(t)$ associated to $\pi_j \xrightarrow{\mathcal{T}} \pi_{j'}$
2: Compute pose error $e_s = x_O - x_{j,j'}(t)$
3: Define pose performance functions $\rho_s(t)$
4: Define the pose normalized error $\xi_s = \rho_s^{-1}(t) e_s$
5: Define reference velocity $v_{O,\mathrm{des}}(\xi_s, t)$
6: Compute velocity error $e_v = v_O - v_{O,\mathrm{des}}(\xi_s, t)$
7: Define velocity performance functions $\rho_v(t)$
8: Define the velocity normalized error $\xi_v = \rho_v^{-1}(t) e_v$
9: Compute distributed control laws $u_i(\xi_s, \xi_v, t), i \in \mathcal{N}$
```

Remark 7.1. Notice by (7.12) and (7.17) that the proposed control protocol is distributed in the sense that each agent needs feedback only from the state of the object's center of mass, which can be obtained by the coupled kinematics. The parameters needed for the computation of $\rho_{p_k}(t), \rho_{\eta_\ell}(t), \rho_{v_m}(t), \forall k \in \{1,2,3\}$, $\ell \in \{\phi, \theta, \psi\}$,

$m \in \{1, \ldots, 6\}$ as well as c_i, g_s, g_v and $\eta_{j,j'}(t), i \in \mathcal{N}$, can be transmitted off-line to the agents. Moreover, the proposed control law does not incorporate any prior knowledge of the model nonlinearities/disturbances or force/torque measurements at the contact points. Furthermore, the proposed methodology results in a low complexity design. Notice that no hard calculations (neither analytic nor numerical) are required to output the proposed control signal, thus making its distributed implementation straightforward.

Remark 7.2. Similarly to (4.42), we can also guarantee internal force regulation by including in (7.20) a vector of desired internal forces $f_{\text{int,d}} \in \mathbb{R}^{6N}$ that belongs to the nullspace of G^T, i.e., $f_{\text{int,d}} = (I_{6N} - \frac{1}{N}G^*(q)G^T(q))\hat{f}_{\text{int,d}}$, where $\hat{f}_{\text{int,d}}$ is a constant vector that can be transmitted off-line to the agents.

The next theorem summarizes the results of this section.

Theorem 7.1. *Consider N agents rigidly grasping an object with unknown coupled dynamics* (7.3) *and $\mathcal{A}(q(t_0)) \in \pi_j, j \in \{1, \ldots, R\}$. Then, the distributed control protocol* (7.10)-(7.19) *guarantees that $\pi_j \xrightarrow{\mathcal{T}} \pi_{j'}$ with time duration $\delta t_{j,j'}$ and all closed loop signals being bounded, and thus establishes a transition relation between π_j and $\pi_{j'}$ for the coupled object-agents system, according to Definition 7.2.*

Proof. By following the proof of Theorem 4.2, we conclude that $\xi_s(t) \in (-1,1)^6$, $\xi_v(t) \in (-1,1)^6$, $\forall t \in \mathbb{R}_{\geq 0}$. Therefore, it holds that $|e_{s_m}(t)| < \rho_{s_m}(t), \forall m \in \{1, \ldots, 6\}$ and thus $|e_{p_k}(t)| < l_0, \forall p \in \{1,2,3\}$, $t \in [t_0, \infty)$, since $\rho^0_{p_k} = l_0, \forall k \in \{1,2,3\}$. Therefore, $p_O(q(t)) \in \mathcal{B}(p_{j,j'}(t), l_0), \forall t \geq t_0$ and, consequently, $p_O(q(t_0 + \delta t_{j,j'})) \in \mathcal{B}(p^c_{\pi_{j'}}, l_0)$, since $p_{j,j'}(t_0 + \delta t_{j,j'}) = p^c_{\pi_{j'}}$. Moreover, since $p_O(q(t)) \in \mathcal{B}(p_{j,j'}(t), l_0)$, we deduce that $\mathcal{B}(p_O(q(t)), \hat{L}) \subset \mathcal{B}(p_{j,j'}(t), l_0 + \hat{L})$ and invoking (7.4) and (7.5), we conclude that $p_s \in \pi_j \cup \pi_{j'}, \forall t \in [t_0, t_0 + \delta t_{j,j'}] \subset [t_0, \infty)$, and therefore a transition relation with time duration $\delta t_{j,j'}$ is successfully established. Finally, since $\rho^0_{\eta_\theta} = \rho_{\eta_\theta}(t_0) = \frac{\pi}{2} - \theta^*$ and $|e_{\eta_\theta}(t)| < \rho_{\eta_\theta}(t) \leq \rho_{\eta_\theta}(t_0), |\theta_{j,j'}(t)| < \theta^*, \forall t \in [t_0, \infty)$, we conclude that $|\theta_O(t)| < \frac{\pi}{2}, \forall t \in [t_0, \infty)$, ensuring thus the representation singularity-free transition $\pi_j \xrightarrow{\mathcal{T}} \pi_{j'}$. □

High-Level Plan Generation

The second part of the proposed solution is the derivation of a high-level plan that satisfies the given MITL formula ϕ and can be generated using standard techniques from automata-based formal verification methodologies. Thanks to our proposed control law that allows the transition $\pi_j \xrightarrow{\mathcal{T}} \pi_{j'}$ for all $\pi_j \in \Pi$ with $\pi_{j'} \in \mathcal{D}(\pi_j)$ in a predefined time interval $\delta t_{j,j'}$, we can abstract the motion of the coupled object-agents system as a finite weighted transition system (WTS) [38]

$$\mathcal{T} = \{\Pi, \Pi_0, \xrightarrow{\mathcal{T}}, \mathcal{AP}, \mathcal{L}, \gamma\}, \tag{7.21}$$

where

- Π is the set of states defined in Section 7.2.1,
- $\Pi_0 \subset \Pi$ is a set of initial states,
- $\xrightarrow{\mathcal{T}} \subseteq \Pi \times \Pi$ is a transition relation according to Definition 7.2.
- $\mathcal{AP}$ and $\mathcal{L}$ are the atomic propositions and the labeling function, respectively, as defined in Section 7.2.1, and
- $\gamma : (\xrightarrow{\mathcal{T}}) \to \mathbb{R}_{\geq 0}$ is a map that assigns to each transition its time duration, i.e., $\gamma(\pi_j \xrightarrow{\mathcal{T}} \pi_{j'}) = \delta t_{j,j'}$.

Therefore, by designing the switching protocol $U_{r_j}^{r_{j+1}}(t)$ from (7.20):

$$U_{r_j}^{r_{j+1}}(t) = -C_g G^*(q(t))\rho_v^{-1}(t) r_v(\xi_v(t))\varepsilon_v(\xi_v(t)), \forall t \in [t_j, t_j + \delta t_{r_j,r_{j+1}}), \quad (7.22)$$

$j \in \mathbb{N}$, with (i) $t_1 = 0$, (ii) $t_{j+1} = t_j + \delta t_{r_j,r_{j+1}}$ and (iii) $r_j \in \{1, \dots, R\}$, $\forall j \in \mathbb{N}$, we can define the *timed run* of the WTS as the infinite sequence $r = (\pi_{r_1}, t_1)(\pi_{r_2}, t_2)\dots$, where $\pi_{r_1} \in \Pi_0$ with $\mathcal{A}(q(0)) \in \pi_{r_1}, \pi_{r_j} \in \Pi, r_j \in \{1, \dots, R\}$ and t_j are the corresponding time stamps such that $\mathcal{A}(q(t_j)) \in \pi_{r_j}, \forall j \in \mathbb{N}$. Every timed run r generates the *timed word* $w(r) = (\mathcal{L}(\pi_{r_1}), t_1)(\mathcal{L}(\pi_{r_2}), t_2)\dots$ over $\mathcal{AP}$ where $\mathcal{L}(\pi_{r_j}), j \in \mathbb{N}$, is the subset of the atomic propositions $\mathcal{AP}$ that are true when $\mathcal{A}(q(t_j)) \in \pi_{r_j}$.

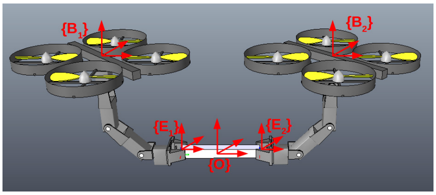
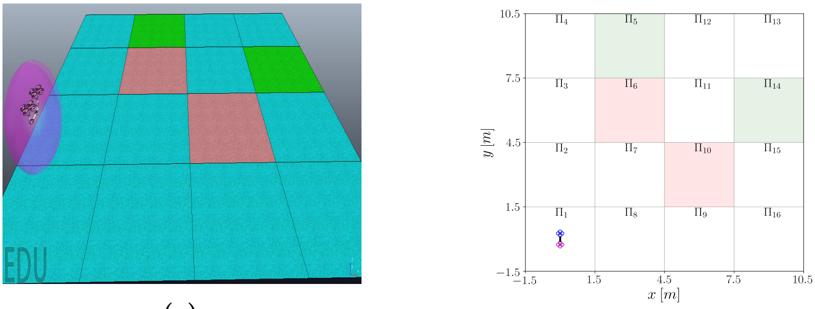


Figure 7.5: The aerial robots employed in the simulation rigidly grasping an object, with the frames $\{B_i\}, \{E_i\}, \{O\}, i \in \mathcal{N} = \{1, 2\}$.

The given MITL formula ϕ is translated into a *Timed Büchi Automaton* $\mathcal{A}_\phi^t$ [39] and the product $\mathcal{A}_p = \mathcal{T} \otimes \mathcal{A}_\phi^t$ is built [38]. The projection of the accepting runs of $\mathcal{A}_p$ onto $\mathcal{T}$ provides a *timed run* r_ϕ of $\mathcal{T}$ that satisfies ϕ; r_ϕ has the form $r_\phi = (\pi_{r_1}, t_1)(\pi_{r_2}, t_2)\dots$, i.e., an infinite[1] sequence of regions π_{r_j} to be visited at specific time instants t_j (i.e., $\mathcal{A}(q(t_j)) \in \pi_{r_j}$) with $t_1 = 0$ and $t_{j+1} = t_j + \delta t_{r_j,r_{j+1}}, r_j \in \{1, \dots, R\}, \forall j \in \mathbb{N}$. More details on the technique are beyond the scope of this work and the reader is referred to [38, 39, 172].

The execution of $r_\phi = (\pi_{r_1}, t_1)(\pi_{r_2}, t_2)\dots$ produces a trajectory $q(t), t \in \mathbb{R}_{\geq 0}$, with timed sequence $\beta_\phi = (q(t_1), t_1)$
$(q(t_2), t_2)\dots$, with $\mathcal{A}(q(t_j)) \in \pi_{r_j}, \forall j \in \mathbb{N}$. Following Definition 7.3, β_ϕ has the timed

[1] It can be proven that if such a run exists, then there also exists a run that can be always represented as a finite prefix followed by infinite repetitions of a finite suffix [38].



(a)

(b)

Figure 7.6: Illustration of the initial workspace and pose of the system object-agents in the V-REP environment (a) and in top view (b). The red cells imply obstacle regions whereas the green cells are the goal ones.

behavior $\sigma_{\beta_\phi} = (\sigma_1, t_1)(\sigma_2, t_2)\dots$ with $\sigma_j \in \mathcal{L}(\pi_{r_j})$, for $\mathcal{A}(q(t_j)) \in \pi_{r_j}, \forall j \in \mathbb{N}$. Since all π_{r_j} belong to $r_\phi, \forall j \in \mathbb{N}$, the latter implies that $\sigma_{\beta_\phi} \models \phi$ and therefore that β_ϕ satisfies ϕ. The aforementioned discussion is summarized as follows:

Theorem 7.2. *The execution of $r_\phi = (\pi_{r_1}, t_1)(\pi_{r_2}, t_2)\dots$ of $\mathcal{T}$ that satisfies ϕ guarantees a timed behavior σ_{β_ϕ} of the coupled object-agents system that yields the satisfaction of ϕ and provides, therefore, a solution to Problem 7.1.*

7.2.3 Simulation Results

The validity of the proposed framework is verified through a simulation study in the Virtual Robot Experimentation Platform (V-REP) [109]. We consider a rectangular rigid body of dimensions $0.025 \times 0.2 \times 0.025$ m^3 representing the object that is rigidly grasped by two agents. Each agent $i \in \mathcal{N} = \{1, 2\}$ consists of a quadrotor base $\{B_i\}$ and a robotic arm of two degrees of freedom $\alpha_{i_1}, \alpha_{i_2} \in [-\frac{\pi}{2}, \frac{\pi}{2}]$, as depicted in Fig. 7.5. The states of the agents are taken as $q_i = [p_{B_i}^T, \eta_{B_i}^T, \alpha_{i_1}, \alpha_{i_2}]^T \in \mathbb{R}^8$ and the control inputs as $\tau_i = [f_{B_i}^T, \mu_{B_i}^T, \tau_{\alpha_1}, \tau_{\alpha_2}]^T$, $i \in \{1, 2\}$. We consider that the quadrotor is fully actuated, as mentioned in Section 7.2.1, and there exists an embedded algorithm that translates the generalized force $\lambda_{B_i} = [f_{B_i}^T, \mu_{B_i}^T]^T$ to the actual motor inputs.

The initial conditions of the system are taken such that $p_O(0) = [0, 0, 1.5]^T$m, $\eta_O(0) = [0, 0, 0]^T$r. The workspace is partitioned into $R = 16$ regions, with $\hat{L} = 0.75$ m and $l_0 = 0.5$ m. Fig. 7.6 illustrates the aforementioned setup at $t = 0$, from which it can be deduced that $\mathcal{A}(q(0)) \in \pi_1$. We further define the atomic propositions $\mathcal{AP} = \{$“green$_1$”, “green$_2$”, “red”, “obs”$\}$, representing goal (“green$_1$”, “green$_2$”) and obstacle (‘obs”) regions with $\mathcal{L}(\pi_5) = \{$“green$_1$”$\}$, $\mathcal{L}(\pi_{14}) = \{$“green$_2$”$\}$, $\mathcal{L}(\pi_6) = \mathcal{L}(\pi_{10}) = \{$“obs”$\}$ and $\mathcal{L}(\pi_j) = \emptyset$, for the remaining regions.

We consider the MITL formula

$$\phi = (\Box_{[0,\infty)} \neg\text{“obs”}) \wedge \Diamond_{[0,60]}(\text{“green}_1\text{”} \wedge \Diamond_{[0,24]}\text{“green}_2\text{”})$$

, which describes the following behavior: the coupled system

1. must always avoid the obstacle regions,
2. must visit the first green region in the first 60 seconds and after that visit the second green region in the next 24 seconds.

By following the procedure described in Section 7.2.2, we obtain the accepting timed run

$$\begin{aligned} r_\phi =& (\pi_{r_1}, t_1)(\pi_{r_2}, t_2)\cdots = (\pi_1, 0)(\pi_2, 6)(\pi_3, 12)(\pi_4, 18)(\pi_5, 24)(\pi_{12}, 30)(\pi_{13}, 36) \\ & (\pi_{14}, 42)(\pi_{11}, 48)(\pi_{12}, 54)(\pi_5, 60). \end{aligned}$$

Regarding each transition $\pi_{r_j} \xrightarrow{\mathcal{T}} \pi_{r_{j+1}}, j \in \{1, \dots, 10\}$, we choose $\delta t_{r_j, r_{j'}} = 6$ s, $p_{r_j, r_{j'}}(t)$ as in (7.6) and $\eta_{r_j, r_{j'}}(t) = [0, 0, \frac{\pi}{4}\sin(\frac{\pi}{3}(t - t_{r_j}))]^T$ [2], where $t_{r_j} = j\delta t_{r_j, r_{j'}} = 6j$ plays the role of t_0 for each transition. Regarding the performance function parameters, we choose $\rho^0_{p_k} = \rho_{p_k}(t_{r_j}) = l_0 = 0.5$m, $l_{p_k} = 0.5$, $\rho^\infty_{p_k} = \lim\limits_{t\to\infty} \rho_{p_k}(t) = 0.1$ m, $\forall k \in \{1, 2, 3\}$, $\rho^0_{\eta_\ell} = \rho_{\eta_\ell}(t_{r_j}) = \frac{\pi}{2}$ r, $l_{\eta_\ell} = 0.5$, $\rho^\infty_{\eta_\ell} = \lim\limits_{t\to\infty} \rho_{\eta_\ell}(t) = \frac{\pi}{12}$ r, $\forall \ell \in \{\phi, \theta, \psi\}$, $\rho^0_{v_m} = \rho_{v_m}(t_{r_j}) = 2|e_{v_m}(t_{r_j})| + 0.5$, $l_{v_m} = 0.5$ and $\rho^\infty_{v_m} = \lim\limits_{t\to\infty} \rho_{v_m}(t) = 0.1$, $m \in \{1, \dots, 6\}, j \in \{1, \dots, 10\}$. The two agents contribute equally to the task by choosing $c_1 = c_2 = 0.5$. Finally, the control gains are chosen as $g_s = 1$, $g_v = 10$.

The simulation results are depicted in Figs. 7.7-7.10. More specifically, Fig. 7.7 depicts the timed transitions of the coupled object-agents system, from which it can be deduced that $p_O(t) \in \mathcal{B}(p_{r_j, r_{j'}}, l_0)$ and therefore $p_s \in \pi_{r_j} \cup \pi_{r_{j'}}$, $\forall p_s \in \mathcal{S}_q, j \in \{1, \dots, 10\}$. Moreover, Fig. 7.8 and 7.9 illustrate the errors $e_s(t)$ and $e_v(t)$ along with the performance functions $\rho_s(t), \rho_v(t)$, respectively, for all the transitions $\pi_{r_j} \to \pi_{r_{j'}}, j \in \{1, \dots, 10\}$. Finally, the resulted control inputs τ_1, τ_2 for the two agents are shown in Fig. 7.10. The aforementioned simulation paradigm is illustrated in the accompanying video.

7.3 Motion and Cooperative Transportation Planning for Multi-Agent Systems under Temporal Logic Formulas

7.3.1 System Model and Problem Formulation

Consider $N > 1$ robotic agents operating in a workspace $\mathcal{W}$ with $M > 0$ objects; $\mathcal{W}$ is a bounded open ball in 3D space, i.e., $\mathcal{W} := \mathring{\mathcal{B}}(0, r_0) = \{p \in \mathbb{R}^3 \text{ s.t. } \|p\| < r_0\}$, where

[2] Note that the nature of the quadrotors makes the whole system underactuated and values $\phi_{r_j, r_{j'}}(t), \theta_{r_j, r_{j'}}(t) \neq 0$ are not possible to be achieved without interfering with $p_O(t)$.

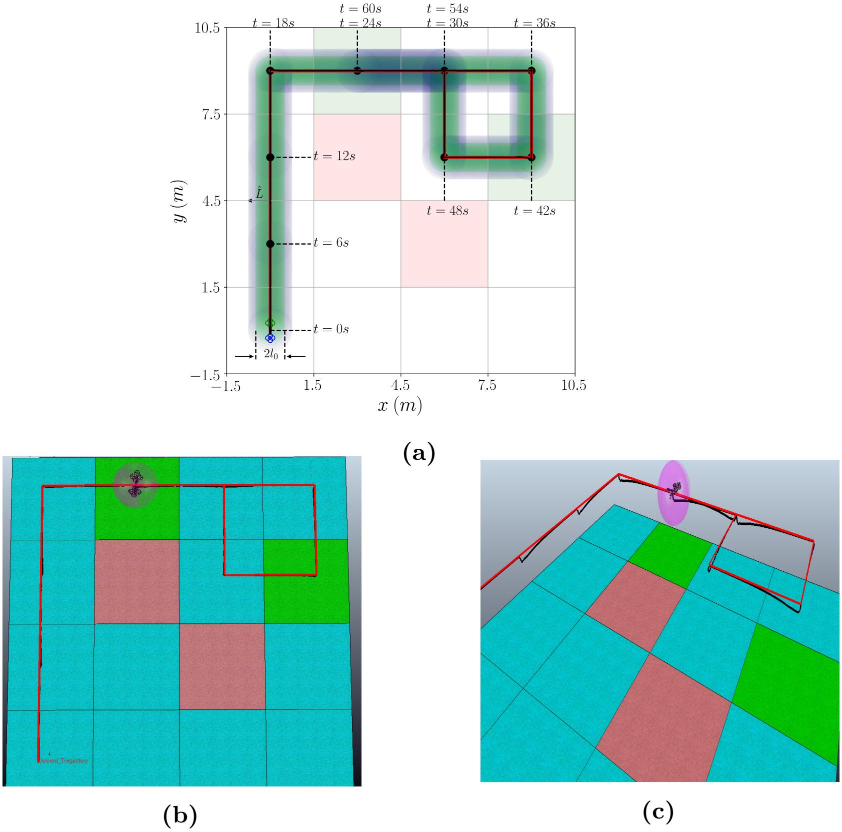


Figure 7.7: (a): The overall desired object trajectory (with red), the actual object trajectory (with black), the domain specified by $\mathcal{B}(p_{r_j,r_{j'}}(t), l_0), \forall j \in \{1, \ldots, 10\}$ (with green), and the domain specified by $\mathcal{B}(p_O(t), \hat{L})$ (with blue), for $t \in [0, 60]$ s. (b), (c): Illustration of the system at the final region at $t = 60$s in the V-REP environment along with the ball $\mathcal{B}(p_O(60), \hat{L})$. Since $p_O \in \mathcal{B}(p_{r_j,r_{j'}}(t), l_0)$, the desired timed run is successfully executed.

$r_0 \in \mathbb{R}_{>0}$ is the radius of $\mathcal{W}$. The objects are represented by rigid bodies whereas the robotic agents are fully actuated and consist of a fully actuated holonomic moving part (i.e., mobile base) and a robotic arm, having, therefore, access to the entire workspace. Within $\mathcal{W}$ there exist $K > 1$ smaller spheres around points of interest, which are described by $\pi_k := \mathcal{B}(p_{\pi_k}, r_{\pi_k}) = \{p \in \mathbb{R}^3 \text{ s.t. } \|p - p_{\pi_k}\| \leq r_{\pi_k}\}$, where $p_{\pi_k} \in \mathbb{R}^3$ is the center and $r_{\pi_k} \in \mathbb{R}_{>0}$ the radius of π_k. We denote the set of all π_k as $\Pi := \{\pi_1, \ldots, \pi_K\}$. Moreover, we introduce disjoint sets of atomic propositions Ψ_i, Ψ^O_j, expressed as boolean variables, that represent services provided to agent $i \in \mathcal{N}$ and object $j \in \mathcal{M}$ in Π. The services provided at each region π_k are given

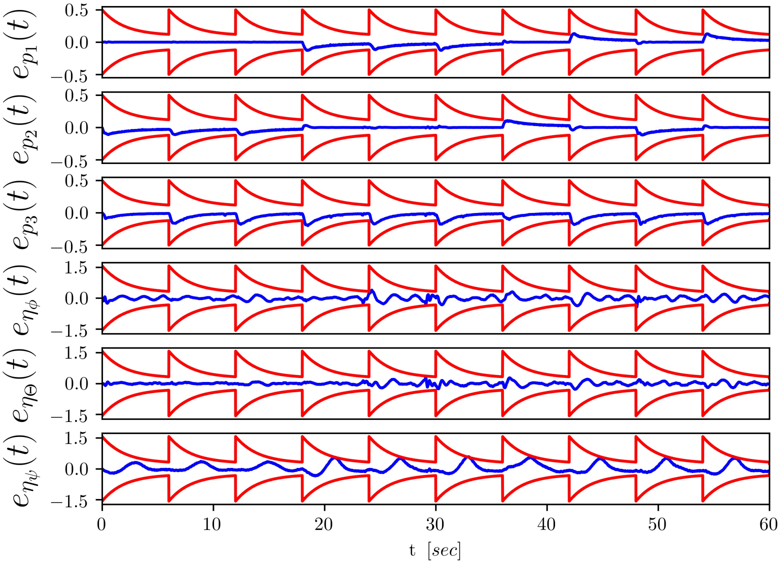


Figure 7.8: The pose errors $e_s(t)$ (with blue) along with the performance functions $\rho_s(t)$ (with red) (in m, m, m, r, r, r, respectively).

by the labeling functions $\mathcal{L}_i : \Pi \to 2^{\Psi_i}, \mathcal{L}_j^O : \Pi \to 2^{\Psi_j^O}$, which assign to each region $\pi_k, k \in \mathcal{K}$, the subset of services Ψ_i and Ψ_j^O, respectively, that can be provided in that region to agent $i \in \mathcal{N} = \{1, \dots, N\}$ and object $j \in \mathcal{M} = \{1, \dots, M\}$, respectively. In addition, we consider that the agents and the object are initially $(t = 0)$ in the regions of interest $\pi_{init(i)}, \pi_{init_O(j)}$, where the functions $init : \mathcal{N} \to \mathcal{K} = \{1, \dots, K\}$, $init_O : \mathcal{M} \to \mathcal{K}$ specify the initial region indices. We denote by $\{E_i\}$, $\{O\}$ the robotic arms' end-effector and object's center of mass frames, respectively; $\{I\}$ corresponds to an inertial frame of reference. In the following, we present the modeling of the coupled kinematics and dynamics of the object and the agents.

We denote by $q_i, \dot{q}_i \in \mathbb{R}^{n_i}$, with $n_i \in \mathbb{N}, \forall i \in \mathcal{N}$, the generalized joint-space variables and their time derivatives for agent i. The overall joint configuration is then $q := [q_1^\top, \dots, q_N^\top]^\top$, $\dot{q} := [\dot{q}_1^\top, \dots, \dot{q}_N^\top]^\top \in \mathbb{R}^n$, with $n := \sum_{i \in \mathcal{N}} n_i$. In addition, the inertial position and Euler-angle orientation of the ith end-effector, denoted by p_i and η_i, respectively, expressed in an inertial reference frame, can be derived by the forward kinematics and are smooth functions of q_i, i.e. $p_i : \mathbb{R}^{n_i} \to \mathbb{R}^3$, $\eta_i : \mathbb{R}^{n_i} \to \mathbb{T}$. The generalized velocity of each agent's end-effector $v_i := [\dot{p}_i^\top, \omega_i^\top]^\top \in \mathbb{R}^6$, can be considered as a transformed state through the differential kinematics $v_i = J_i(q_i)\dot{q}_i$ [26], where $J_i : \mathbb{R}^{n_i} \to \mathbb{R}^{6 \times n_i}$ is a smooth function representing the geometric Jacobian matrix, $\forall i \in \mathcal{N}$ [26]. The matrix inverse of J_i is well defined in the set away from *kinematic singularities* [26], which we define as $\mathbb{S}_i := \{q_i \in \mathbb{R}^{n_i} :$

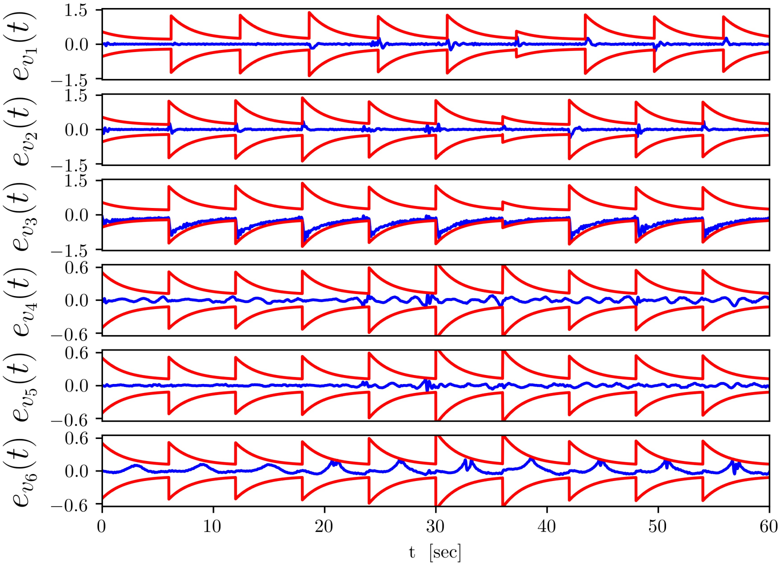


Figure 7.9: The velocity errors $e_v(t)$ (with blue) along with the performance functions $\rho_v(t)$ (with red) (in $m/s, m/s, m/s, r/s, r/s, r/s$, respectively).

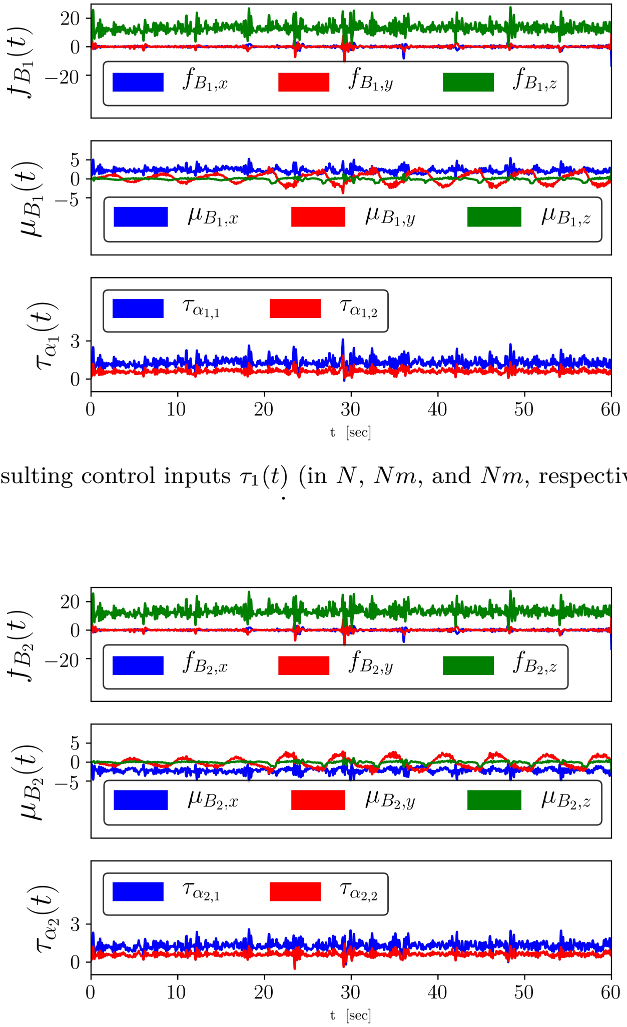
$\det(J_i(q_i)[J_i(q_i)]^\top) > 0\}$, $\forall i \in \mathcal{N}$.

The differential equation describing the *task-space* dynamics of each agent is [26]:

$$M_i(q_i)\dot{v}_i + C_i(q_i, \dot{q}_i)v_i + g_i(q_i) = u_i - f_i, \tag{7.23a}$$

with the standard dynamic terms (see previous chapters), which are well-defined in the set $\mathbb{S}_i$, away from kinematic singularities. Avoidance of such configurations is not explicitly taken account in this section. Note, however, that the agents' tasks consist of navigating as well as cooperatively transporting the objects to predefined points in the workspace. This along with the fact that the agents consist of fully actuated moving bases imposes a kinematic redundancy, which can be exploited to avoid kinematic singularities.

We consider that each agent i, for a given q_i, covers a spherical region $\mathcal{A}_i : \mathbb{R}^{n_i} \rightrightarrows \mathbb{R}^3$ of constant radius $r_i \in \mathbb{R}_{>0}$ that bounds its volume for that given q_i, i.e., $\mathcal{A}_i(q_i) \coloneqq \mathcal{B}(c_i(q_i), r_i)$, where $c_i : \mathbb{R}^{n_i} \to \mathbb{R}^3$ is the center of the spherical region (a point on the robotic arm), $\forall i \in \mathcal{N}$; $\mathcal{A}_i$ can be obtained by considering the smallest sphere that covers the workspace of the robotic arm, extended with the mobile base part. Moreover, we consider that the agents have specific power capabilities, which for simplicity, we match to positive integers $\zeta_i > 0$, $i \in \mathcal{N}$, via an analogous relation.



(a) The resulting control inputs $\tau_1(t)$ (in N, Nm, and Nm, respectively) .

(b) The resulting control inputs $\tau_2(t)$ (in N, Nm, and Nm, respectively)

Figure 7.10: The resulting control inputs $\tau_i = [f_{B_i}^T, \mu_{B_i}^T, \tau_{\alpha_{i,1}}, \tau_{\alpha_{i,2}}]$ for $i = 1$ and $i = 2$.

Regarding the objects, we denote by $x^O_j := [(p^O_j)^\top, (\eta^O_j)^\top]^\top \in \mathbb{M}$, $v^O_j := [(\dot{p}^O_j)^\top, (\omega^O_j)^\top]^\top \in \mathbb{R}^{12}$, $\forall j \in \mathcal{M}$, the pose (with p^O_j being the position of the center of mass with respect to (and expressed in) an inertial reference frame, and $\eta^O_j := [\eta^O_{j,1}, \eta^O_{j,2}, \eta^O_{j,3}]^\top$ denoting the extrinsic Euler angles) and generalized velocity of the jth object's center of mass, which is considered as the object's state. The object dynamic equations are given by the standard Newton-Euler form (see previous chapters):

$$\dot{x}^O_j = J^O_j(x^O_j)v^O_j, \tag{7.24a}$$

$$M_O(x^O_j)\dot{v}^O_j + C_O(x^O_j, v^O_j)v^O_j + g_O(x^O_j) = f^O_j, \tag{7.24b}$$

where $J^O_j(x^O_j)$ represents here the representation Jacobian matrix that is only defined in the subset of $\mathbb{M}$ that does not include the configurations where the pitch angle $\eta^O_{j,2}$ is $\pm\frac{\pi}{2}$, namely, *representation singularities*, i.e., $J^O_j : \mathbb{S}^O_j \to \mathbb{R}^{6\times 6}$, with $\mathbb{S}^O_j := \{x^O_j \in \mathbb{M} : |\eta^O_{j,2}| < \frac{\pi}{2}\}$, $\forall j \in \mathcal{M}$.

Similarly to the agents, each object's volume is represented by the spherical set $\mathcal{O}_j : \mathbb{R}^3 \rightrightarrows \mathbb{R}^3$ of a constant radius $r^O_j \in \mathbb{R}_{>0}$, i.e., $\mathcal{O}_j(x^O_j) := \mathcal{B}(x^O_j, r^O_j)$, $\forall j \in \mathcal{M}$.

Next, we provide the coupled dynamics between an object $j \in \mathcal{M}$ and a subset $\mathcal{T} \subseteq \mathcal{N}$ of agents that grasp it rigidly (see Fig. 7.1). Although the derivation of the coupled dynamics is identical to the previous chapters, we present it here due to the slight change of notation. In view of Fig. 7.1, one concludes that the pose of the agents and the object's center of mass are related as

$$p_i(q_i) = p^O_j + R_i(q_i)p^{E_i}_{E_i/O_j}, \tag{7.25a}$$

$$\eta_i(q_i) = \eta^O_j + \eta_{E_i/O_j}, \tag{7.25b}$$

$\forall i \in \mathcal{T}$, where $R_i : \mathbb{R}^{n_i} \to SO(3)$ is the rotation matrix from $\{I\}$ to the ith agent's end-effector $\{E_i\}$, and $p^{E_i}_{E_i/O_j}$, η_{E_i/O_j} are the *constant* distance and orientation offset between $\{O\}$ and $\{E_i\}$, respectively. Following (7.25), along with the fact that, due to the grasping rigidity, it holds that $\omega_i = \omega^O_j, \forall i \in \mathcal{T}$, one obtains

$$v_i = J^O_{i,j}(q_i)v^O_j, \tag{7.26}$$

where $J^O_{i,j} : \mathbb{R}^{n_i} \to \mathbb{R}^{6\times 6}$ is the object-to-agent Jacobian matrix, with

$$J^O_{i,j}(x) := \begin{bmatrix} I_3 & -S(R_i(x)p^{E_i}_{E_i/O_j}) \\ 0_{3\times 3} & I_3 \end{bmatrix}, \forall x \in \mathbb{R}^{n_i},$$

which is always full-rank.

The agent task-space dynamics (7.23) can be written in vector form as:

$$M_{\mathcal{T}}(q_{\mathcal{T}})\dot{v}_{\mathcal{T}} + C_{\mathcal{T}}(q_{\mathcal{T}}, \dot{q}_{\mathcal{T}})v_{\mathcal{T}} + g_{\mathcal{T}}(q_{\mathcal{T}}) = u_{\mathcal{T}} - f_{\mathcal{T}}, \tag{7.27}$$

where $q_{\mathcal{T}} := [q_i^\top]^\top_{i\in\mathcal{T}}$, $\dot{q}_{\mathcal{T}} := [\dot{q}_i^\top]^\top_{i\in\mathcal{T}}$, $v_{\mathcal{T}} := [v_i^\top]^\top_{i\in\mathcal{T}}$, $g_{\mathcal{T}}(q_{\mathcal{T}}) := \Big[[g_i(q_i)]^\top\Big]^\top_{i\in\mathcal{T}}$, $f_{\mathcal{T}} := [f_i^\top]^\top_{i\in\mathcal{T}}$ and $M_{\mathcal{T}}(q_{\mathcal{T}}) := \text{diag}\{[M_i(q_i)]_{i\in\mathcal{T}}\}$, $C_{\mathcal{T}}(q_{\mathcal{T}}, \dot{q}_{\mathcal{T}}) := \text{diag}\{[C_i(q_i, \dot{q}_i)]_{i\in\mathcal{T}}\}$.

The kineto-statics duality along with the grasp rigidity suggest that the force f_j^O acting on the object's center of mass and the generalized forces $f_i, i \in \mathcal{T}$, exerted by the agents at the grasping points, are related through:

$$f_j^O = [G_{\mathcal{T},j}(q_{\mathcal{T}})]^\top f_{\mathcal{T}}, \tag{7.28}$$

where $G_{\mathcal{T},j} : \mathbb{R}^{n_{\mathcal{T}}} \to \mathbb{R}^{6N\times 6}$, with $G_{\mathcal{T},j}(q_{\mathcal{T}}) := \Big[[J_{i,j}^O(q_i)]^\top\Big]_{i\in\mathcal{T}}^\top$ is the grasp matrix, and $n_{\mathcal{T}} := \sum_{i\in\mathcal{T}} n_i$. By combining (7.28) with (7.24), (7.27), and (7.25) we obtain the coupled dynamics

$$\widetilde{M}_{\mathcal{T},j}(x_{\mathcal{T},j})\dot{v}_j^O + \widetilde{C}_{\mathcal{T},j}(x_{\mathcal{T},j})v_j^O + \widetilde{g}_{\mathcal{T},j}(x_{\mathcal{T},j}) = [G_{\mathcal{T},j}(q_{\mathcal{T}})]^\top u_{\mathcal{T}}, \tag{7.29}$$

where

$$\begin{aligned}
\widetilde{M}_{\mathcal{T},j}(x_{\mathcal{T},j}) &:= M_O(x_j^O) + [G_{\mathcal{T},j}(q_{\mathcal{T}})]^\top M_{\mathcal{T}}(q_{\mathcal{T}})G_{\mathcal{T},j}(q_{\mathcal{T}}) \\
\widetilde{C}_{\mathcal{T},j}(x_{\mathcal{T},j}) &:= C_O(x_j^O, v_j^O) + [G_{\mathcal{T},j}(q_{\mathcal{T}})]^\top M_{\mathcal{T}}(q_{\mathcal{T}})\dot{G}_{\mathcal{T},j}(q_{\mathcal{T}}) \\
&\quad + [G_{\mathcal{T},j}(q_{\mathcal{T}})]^\top C_{\mathcal{T}}(q_{\mathcal{T}}, \dot{q}_{\mathcal{T}})G_{\mathcal{T},j}(q_{\mathcal{T}}) \\
\widetilde{g}_{\mathcal{T},j}(x_{\mathcal{T},j}) &:= g_O(x_j^O) + [G_{\mathcal{T},j}(q_{\mathcal{T}})]^\top g_{\mathcal{T}}(q_{\mathcal{T}}).
\end{aligned}$$

and $x_{\mathcal{T},j}$ is the overall state $x_{\mathcal{T},j} := [q_{\mathcal{T}}^\top, \dot{q}_{\mathcal{T}}^\top, (x_j^O)^\top, (v_j^O)^\top]^\top \in \mathbb{R}^{2n_{\mathcal{T}}+6} \times \mathbb{M}$. Note that the aforementioned coupled terms are defined only when $q_i \in \mathbb{S}_i \subset \mathbb{R}^{n_i}, \forall i \in \mathcal{T}$. We also use the following Lemma from Chapter 4 that is necessary for the following analysis.

Lemma 7.1. *The matrices $B_i(q_i)$ and $\widetilde{M}_{\mathcal{T},j}(x_{\mathcal{T},j})$ are symmetric and positive definite and the matrices $\dot{B}_i(q_i) - 2N_i(q_i, \dot{q}_i)$ and $\dot{\widetilde{M}}_{\mathcal{T},j}(x_{\mathcal{T},j}) - 2\widetilde{C}_{\mathcal{T},j}(x_{\mathcal{T},j})$ are skew symmetric, $\forall i \in \mathcal{N}, j \in \mathcal{M}, \mathcal{T} \subseteq \mathcal{N}$.*

Regarding the volume of the coupled agents-object system, we denote by $\mathcal{AO}_{\mathcal{T},j} : \mathbb{R}^3 \rightrightarrows \mathbb{R}^3$ the sphere centered at p_j^O with constant radius $r_{\mathcal{T},j} \in \mathbb{R}_{>0}$, i.e., $\mathcal{AO}_{\mathcal{T},j}(p_j^O) := \mathcal{B}(p_j^O, r_{\mathcal{T},j})$, which is large enough to cover the volume of the coupled system in all configurations $q_{\mathcal{T}}$[3]. This conservative formulation emanates from the sphere-world restriction of the multi-agent navigation function framework [36, 114]. In order to take into account other spaces, ideas from [174] could be employed or extensions of the respective works of [35], [175] to the multi-agent case could be developed.

Moreover, in order to take into account the introduced agents' power capabilities ζ_i, $i \in \mathcal{N}$, we consider a function $\Lambda \in \{\top, \bot\}$ that outputs whether the agents that grasp an object are able to transport the object, based on their power capabilities. For instance, $\Lambda(m_j^O, \zeta_{\mathcal{T}}) = \top$, where $m_j^O \in \mathbb{R}_{>0}$ is the mass of object j and $\zeta_{\mathcal{T}} := [\zeta_i]_{i\in\mathcal{T}}^\top$, implies that the agents $\mathcal{T}$ have sufficient power capabilities to cooperatively transport object j.

[3] $r_{\mathcal{T},j}$ can be chosen as the largest distance of the object's center of mass to a point in the agents' volume over all possible $q_{\mathcal{T}}$ (see previous section)

Problem Formulation

In this subsection, the problem formulation is provided. We first introduce some preliminary required notation. We define the boolean functions $\mathcal{AG}_{i,j} : \mathbb{R}^{n_i} \times \mathbb{M} \to \{\top, \bot\}, i \in \mathcal{N}, j \in \mathcal{M}$, to denote whether agent $i \in \mathcal{N}$ rigidly grasps an object $j \in \mathcal{M}$ at a given configuration q_i, x^O_j; We also define $\mathcal{AG}_{i,0} : \mathbb{R}^{n_i} \times \mathbb{M}^M \to \{\top, \bot\}$, to denote that agent i does not grasp any objects, i.e., $\mathcal{AG}_{i,j}(q_i, x^O_j) = \bot, \forall j \in \mathcal{M} \Leftrightarrow \mathcal{AG}_{i,0}(q_i, x^O) = \top$, $\forall i \in \mathcal{N}$, where $x^O \coloneqq [(x^O_j)^\top]^\top_{j \in \mathcal{M}} \in \mathbb{M}^M$. Note also that $\mathcal{AG}_{i,\ell}(q_i, x^O_\ell) = \top, \ell \in \mathcal{M} \Leftrightarrow \mathcal{AG}_{i,j}(q_i, x^O_j) = \bot, \forall j \in \mathcal{M} \backslash \{\ell\}$, i.e., agent i can grasp at most one object at a time.

In addition, we use the boolean functions $\mathcal{C}_{i,l} : \mathbb{R}^{n_i + n_l} \to \{\bot, \top\}$, $\mathcal{C}_{i,O_j} : \mathbb{R}^{n_i} \times \mathbb{M} \to \{\bot, \top\}$, $\mathcal{C}_{O_j,O_\ell} : \mathbb{M}^2 \to \{\bot, \top\}$, to denote collision between agents $i, l \in \mathcal{N}, i \neq l$, agent $i \in \mathcal{N}$ and object $j \in \mathcal{M}$ and objects $j, \ell \in \mathcal{M}, j \neq \ell$, respectively.

We also assume the existence of a procedure $\mathcal{P}_s$ that outputs whether or not a set of non-intersecting spheres fits in a larger sphere as well as possible positions of the spheres in the case they fit. More specifically, given a region of interest π_k and a number $\widetilde{N} \in \mathbb{N}$ of sphere radii (of agents and/or objects) the procedure can be seen as a function $\mathcal{P}_s \coloneqq [\mathcal{P}_{s,0}, \mathcal{P}^\top_{s,1}]^\top$, where $\mathcal{P}_{s,0} : \mathbb{R}^{\widetilde{N}+1}_{\geq 0} \to \{\top, \bot\}$ outputs whether the spheres fit in the region π_k whereas $\mathcal{P}_{s,1}$ provides possible configurations of the agents and the objects or 0 in case the spheres do not fit. For instance, $P_{s,0}(r_{\pi_2}, r_1, r_3, r^O_1, r^O_5)$ determines whether the agents 1, 3 and the objects 1, 5 fit in region π_2, without colliding with each other; $(q_1, q_3, x^O_1, x^O_5) = P_{s,1}(r_{\pi_2}, r_1, r_3, r^O_1, r^O_5)$ provides a set of configurations such that $\mathcal{A}_1(q_1), \mathcal{A}_3(q_3), \mathcal{O}_1(x^O_1), \mathcal{O}_5(x^O_5) \subset \pi_2$ and $\mathcal{C}_{1,3}(q_1, q_3) = \mathcal{C}_{O_1,O_5}(x^O_1, x^O_5) = \mathcal{C}_{i,O_j}(q_i, x^O_j) = \bot, \forall (i,j) \in \{1,3\} \times \{1,5\}$. The problem of finding an algorithm $\mathcal{P}_s$ is a special case of the sphere packing problem [176]. Note, however, that we are not interested in finding the maximum number of spheres that can be packed in a larger sphere but, rather, in the simpler problem of determining whether a set of spheres can be packed in a larger sphere.

The following definitions address the transitions of the agents and the objects between the regions of interest.

Definition 7.5. (**Transition**) Consider that $\mathcal{A}_i(q_i(t_0)) \subset \pi_k$, for some $i \in \mathcal{N}, k \in \mathcal{K}, t_0 \in \mathbb{R}_{\geq 0}$, and $\mathcal{C}_{i,l}(q_i(t_0), q_l(t_0)) = \mathcal{C}_{i,O_j}(q_j(t_0), x^O_j(t_0)) = \bot, \forall l \in \mathcal{N} \backslash \{i\}, j \in \mathcal{M}$. Then, there exists a transition for agent i from region π_k to $\pi_{k'}, k' \in \mathcal{K}$, denoted as $\pi_k \to_i \pi_{k'}$, if there exists a finite $t_f \geq t_0$ and a bounded feedback control trajectory u_i such that $\mathcal{A}_i(q_i(t_f)) \subset \pi_{k'}$, $\mathcal{C}_{i,l}(q_i(t), q_l(t)) = \mathcal{C}_{i,O_j}(q_i(t), x^O_j(t)) = \bot$, and $\mathcal{A}_i(q_i(t)) \cap \pi_m = \emptyset$, $\forall t \in [t_0, t_f], l \in \mathcal{N} \backslash \{i\}, j \in \mathcal{M}, m \in \mathcal{K} \backslash \{k, k'\}$.

Definition 7.6. (**Grasping**) Consider that $\mathcal{A}_i(q_i(t_0)) \subset \pi_k$, $\mathcal{O}_j(x^O_j(t_0)) \subset \pi_k$, $k \in \mathcal{K}$ for some $i \in \mathcal{N}$, $j \in \mathcal{M}$, $t_0 \in \mathbb{R}_{\geq 0}$, with $\mathcal{AG}_{i,0}(q_i(t_0), x^O(t_0)) = \top$, and

1. $\mathcal{C}_{i,l}(q_i(t_0), q_l(t_0)) = \mathcal{C}_{i,O_{j'}}(q_i(t_0), x^O_{j'}(t_0)) = \bot, \forall l \in \mathcal{N} \backslash \{i\}, j' \in \mathcal{M}$,

2. $\mathcal{C}_{i',O_j}(q_{i'}(t_0), x^O_j(t_0)) = \mathcal{C}_{O_j,O_\ell}(x^O_j(t_0), x^O_\ell(t_0)) = \bot, \forall i' \in \mathcal{N}, \ell \in \mathcal{M} \backslash \{j\}$.

Then, agent i *grasps* object j, denoted as $i \xrightarrow{g} j$, if there exists a finite $t_f \geq t_0$ and a bounded control trajectory u_i such that $\mathcal{AG}_{i,j}(q_i(t_f), x^O_j(t_f)) = \top$, $\mathcal{A}_i(q_i(t)) \subset \pi_k$, $\mathcal{O}_j(x^O_j(t)) \subset \pi_k$, $k \in \mathcal{K}$ with

1. $\mathcal{C}_{i,l}(q_i(t), q_l(t)) = \mathcal{C}_{i,O_\ell}(q_i(t), x^O_\ell(t)) = \bot$,
2. $\mathcal{C}_{l,O_j}(q_l(t), x^O_j(t)) = \mathcal{C}_{O_j,O_\ell}(x^O_j(t), x^O_\ell(t)) = \bot$,

$\forall t \in [t_0, t_f], l \in \mathcal{N}\backslash\{i\}, \ell \in \mathcal{M}\backslash\{j\}$.

Definition 7.7. (**Releasing**) Consider that $\mathcal{A}_i(q_i(t_0)) \subset \pi_k$, $\mathcal{O}_j(x^O_j(t_0)) \subset \pi_k$, $k \in \mathcal{K}$ for some $i \in \mathcal{N}$, $j \in \mathcal{M}$, $t_0 \in \mathbb{R}_{\geq 0}$, with $\mathcal{AG}_{i,j}(q_i(t_0), x^O_j(t_0)) = \top$, and

1. $\mathcal{C}_{i,l}(q_i(t_0), q_l(t_0)) = \mathcal{C}_{i,O_\ell}(q_i(t_0), x^O_\ell(t_0)) = \bot$,
2. $\mathcal{C}_{l,O_j}(q_l(t_0), x^O_j(t_0)) = \mathcal{C}_{O_j,O_\ell}(x^O_j(t_0), x^O_\ell(t_0)) = \bot$,

$\forall l \in \mathcal{N}\backslash\{i\}, \ell \in \mathcal{M}\backslash\{j\}$. Then, agent i *releases* object j, denoted as $i \xrightarrow{r} j$, if there exists a finite $t_f \geq t_0$ and a bounded control trajectory u_i such that $\mathcal{AG}_{i,0}(q_i(t_f), x^O(t_f)) = \top$, $\mathcal{A}_i(q_i(t)) \subset \pi_k$, $\mathcal{O}_j(x^O_j(t)) \subset \pi_k$, $k \in \mathcal{K}$ with

1. $\mathcal{C}_{i,l}(q_i(t), q_l(t)) = \mathcal{C}_{i,O_\ell}(q_i(t), x^O_\ell(t)) = \bot$,
2. $\mathcal{C}_{l,O_j}(q_l(t), x^O_j(t)) = \mathcal{C}_{O_j,O_\ell}(x^O_j(t), x^O_\ell(t)) = \bot$,

$\forall t \in [t_0, t_f], l \in \mathcal{N}\backslash\{i\}, \ell \in \mathcal{M}\backslash\{j\}$.

Definition 7.8. (**Transportation**) Consider a nonempty subset of agents $\mathcal{T} \subseteq \mathcal{N}$ with $\mathcal{A}_i(q_i(t_0)) \subset \pi_k$, $\forall i \in \mathcal{T}$, and $\mathcal{O}_j(x^O_j(t_0)) \subset \pi_k$, for some $j \in \mathcal{M}$, $k \in \mathcal{K}$, $t_0 \geq 0$, with $\mathcal{AG}_{i,j}(q_i(t_0), x^O_j(t_0)) = \top$, $\forall i \in \mathcal{T}$ and

1. $\mathcal{C}_{i,l}(q_i(t_0), q_l(t_0)) = \mathcal{C}_{i,O_\ell}(q_i(t_0), x^O_\ell(t_0)) = \bot$,
2. $\mathcal{C}_{z,O_j}(q_l(t_0), x^O_j(t_0)) = \mathcal{C}_{O_j,O_\ell}(x^O_j(t_0), x^O_\ell(t_0)) = \bot$,

$\forall i, l \in \mathcal{N}$, with $i \neq l$, $\ell \in \mathcal{M}\backslash\{j\}, z \in \mathcal{N}\backslash\mathcal{T}$. Then, the team of agents $\mathcal{T}$ transports the object j from region π_k to region $\pi_{k'}, k' \in \mathcal{K}$, denoted as $\pi_k \xrightarrow{T}_{\mathcal{T},j} \pi_{k'}$, if there exists a finite $t_f \geq t_0$ and bounded control laws $u_i, i \in \mathcal{T}$, such that $\mathcal{A}_i(q_i(t_f)) \subset \pi_{k'}, \forall i \in \mathcal{T}, \mathcal{O}_j(x^O_j(t_f)) \subset \pi_{k'}$, $\mathcal{AG}_{i,j}(q_i(t), x^O_j(t)) = \top$, and

1. $\mathcal{C}_{i,l}(q_i(t), q_l(t)) = \mathcal{C}_{i,O_\ell}(q_i(t), x^O_\ell(t)) = \bot$,
2. $\mathcal{C}_{z,O_j}(q_l(t), x^O_j(t)) = \mathcal{C}_{O_j,O_\ell}(x^O_j(t), x^O_\ell(t)) = \bot$,

and $\mathcal{AO}_{\mathcal{T},j}(p^O_j(t)) \cap \pi_m = \emptyset$, $\forall t \in [t_0, t_f], i, l \in \mathcal{N}$, with $i \neq l$, $\ell \in \mathcal{M}\backslash\{j\}, z \in \mathcal{N}\backslash\mathcal{T}, m \in \mathcal{K}\backslash\{k, k'\}$.

Loosely speaking, the aforementioned definitions correspond to specific actions of the agents, namely *transition*, *grasp*, *release*, and *transport*. We do not define these actions explicitly though, since we will employ directly designed continuous control inputs u_i, as will be seen later. Moreover, in the *grasping/releasing* definitions, we have not incorporated explicitly collisions between the agent and the object to be grasped/released other than the grasping point. Such collisions will be assumed to be avoided in the next section.

Our goal is to control the multi-agent system such that the agents and the objects obey a given specification over their atomic propositions $\Psi_i, \Psi^O_j, \forall i \in \mathcal{N}, j \in \mathcal{M}$. Given the trajectories $q_i(t), x^O_j(t), t \in \mathbb{R}_{\geq 0}$, of agent i and object j, respectively, their corresponding *behaviors* are given by the infinite sequences

$$\begin{aligned} b_i &\coloneqq (q_i(t), \sigma_i) \coloneqq (q_i(t_{i,1}), \sigma_{i,1})(q_i(t_{i,2}), \sigma_{i,2}) \dots, \\ b^O_j &\coloneqq (x^O_j(t), \sigma^O_j) \coloneqq (x^O_j(t^O_{j,1}), \sigma^O_{j,1})(x^O_j(t^O_{j,2}), \sigma^O_{j,2}) \dots, \end{aligned}$$

with $t_{i,\ell+1} > t_{i,\ell} \geq 0, t^O_{j,\ell+1} > t^O_{j,\ell} \geq 0, \forall \ell \in \mathbb{N}$, representing specific time stamps. The sequences σ_i, σ^O_j are the services provided to the agent and the object, respectively, over their trajectories, i.e., $\sigma_{i,\ell} \in 2^{\Psi_i}, \sigma^O_{j,l} \in 2^{\Psi^O_j}$ with $\mathcal{A}_i(q_i(t_{i,\ell})) \subset \pi_{k_{i,\ell}}, \sigma_{i,\ell} \in \mathcal{L}_i(\pi_{k_{i,\ell}})$ and $\mathcal{O}_j(x^O_j(t^O_{j,l})) \subset \pi_{k^O_{j,l}}, \sigma^O_{j,l} \in \mathcal{L}^O_j(\pi_{k^O_{j,l}}), k_{i,\ell}, k^O_{j,l} \in \mathcal{K}, \forall \ell, l \in \mathbb{N}, i \in \mathcal{N}, j \in \mathcal{M}$, where $\mathcal{L}_i$ and $\mathcal{L}^O_j$ are the previously defined labeling functions. The following Lemma then follows:

Lemma 7.2. *The behaviors b_i, b^O_j satisfy formulas ϕ_i, ϕ_{O_j} if $\sigma_i \models \phi_i$ and $\sigma^O_j \models \phi^O_j$, respectively.*

The control objectives are given as LTL formulas ϕ_i, ϕ^O_j over Ψ_i, Ψ^O_j, respectively, $\forall i \in \mathcal{N}, j \in \mathcal{M}$. The LTL formulas ϕ_i, ϕ^O_j are satisfied if there exist behaviors b_i, b^O_j of agent i and object j that satisfy ϕ_i, ϕ^O_j. We are now ready to give a formal problem statement:

Problem 7.2. Consider N robotic agents and M objects in $\mathcal{W}$ subject to the dynamics (7.23) and (7.24), respectively, and

1. $\dot{q}_i(0) = 0, v^O_j = 0$, $\mathcal{A}_i(q_i(0)) \subset \pi_{\text{init}(i)}, \mathcal{O}_j(x^O_j(0)) \subset \pi_{\text{init}_O(j)}$, $\forall i \in \mathcal{N}, j \in \mathcal{M}$,
2. $\mathcal{C}_{i,l}(q_i(0), q_l(0)) = \mathcal{C}_{O_j,O_\ell}(x^O_j(0), x^O_\ell(0)) = \mathcal{C}_{i,O_j}(q_i(0), x^O_\ell(0)) = \bot$, $\forall i, l \in \mathcal{N}, i \neq l, j, \ell \in \mathcal{M}, j \neq \ell$.

Given the disjoint sets Ψ_i, Ψ^O_j, N LTL formulas ϕ_i over Ψ_i and M LTL formulas ϕ^O_j over Ψ^O_j, develop a control strategy that achieves behaviors b_i, b^O_j which yield the satisfaction of $\phi_i, \phi^O_j, \forall i \in \mathcal{N}, j \in \mathcal{M}$.

Note that it is implicit in the problem statement the fact that the agents/objects starting in the same region can actually fit without colliding with each other. Technically, it holds that $\mathcal{P}_{s,0}(r_{\pi_k}, [r_i]_{i \in \{i \in \mathcal{N}: \text{init}(i) = k\}}, [r^O_j]_{j \in \{j \in \mathcal{M}: \text{init}_O(j) = k\}}) = \top$, $\forall k \in \mathcal{K}$.

7.3.2 Main Results

Continuous Control Design

The first ingredient of our solution is the development of feedback control laws that establish agent transitions and object transportations as defined in Def. 7.5 and 7.8, respectively. We do not focus on the grasping/releasing actions of Def. 7.6, 7.7 and we refer to some existing methodologies that can derive the corresponding control laws (e.g., [177],[178]).

Assume that the conditions of Problem 7.2 hold for some $t_0 \in \mathbb{R}_{\geq 0}$, i.e., all agents and objects are located in regions of interest with zero velocity. We design a control law such that a subset of agents performs a transition between two regions of interest and another subset of agents performs cooperative object transportation, according to Def. 7.5 and 7.8, respectively. Let $\mathcal{Z}, \mathcal{T}, \mathcal{G}, \mathcal{R} \subseteq \mathcal{N}$ denote disjoint sets of agents corresponding to transition, transportation, grasping and releasing actions, respectively, with $|\mathcal{Z}| + |\mathcal{T}| + |\mathcal{G}| + |\mathcal{R}| \leq |\mathcal{N}|$ and $\mathcal{A}_z(q_z(t_0)) \subset \pi_{k_z}$, $\mathcal{A}_\tau(q_\tau(t_0)) \subset \pi_{k_\tau}$, $\mathcal{A}_g(q_g(t_0)) \subset \pi_{k_g}$, $\mathcal{A}_\rho(q_\rho(t_0)) \subset \pi_{k_\rho}$, where $k_z, k_\tau, k_g, k_\rho \in \mathcal{K}$, $\forall z \in \mathcal{Z}, \tau \in \mathcal{T}, g \in \mathcal{G}, \rho \in \mathcal{R}$. Note that there might be idle agents in some regions, not performing any actions, i.e., the set $\mathcal{N} \backslash (\mathcal{Z} \cup \mathcal{V} \cup \mathcal{G} \cup \mathcal{Q})$ might not be empty.

More specifically, regarding the transportation actions, we consider that the set $\mathcal{T}$ consists of $\bar{T}$ disjoint teams of agents, with each team consisting of agents that are in the same region of interest and aim to collaboratively transport an object, i.e. $\mathcal{T} = \mathcal{T}_1 \cup \mathcal{T}_2 \cup \dots \mathcal{T}_{\bar{T}}$, and $\mathcal{A}_\tau(q_\tau(t_0)) \subset \pi_{k_{\mathcal{T}_m}}, \forall \tau \in \mathcal{T}_m, m \in \{1, \dots, \bar{T}\}$, where $k_{\mathcal{T}_m} \in \mathcal{K}, \forall m \in \{1, \dots, \bar{T}\}$. Let also $\mathcal{S} \coloneqq \{s_{\mathcal{T}_1}, s_{\mathcal{T}_2}, \dots, s_{\mathcal{T}_{\bar{T}}}\}, \mathcal{X} \coloneqq \{[x_g]_{g \in \mathcal{G}}\}, \mathcal{Y} \coloneqq \{[y_\rho]_{\rho \in \mathcal{R}}\} \subseteq \mathcal{M}$ be disjoint sets of objects to be transported, grasped, and released, respectively. More specifically, each team $\mathcal{T}_m$ in the set $\mathcal{T}$ will transport cooperatively object $s_{\mathcal{T}_m}$, $m \in \{1, \dots, \bar{T}\}$, each agent $g \in \mathcal{G}$ will grasp object $x_g \in \mathcal{X}$ and each agent $\rho \in \mathcal{R}$ will release object $y_\rho \in \mathcal{Y}$. Then, suppose that the following conditions also hold at t_0:

- $\mathcal{O}_{s_{\mathcal{T}_m}}(x^O_{s_{\mathcal{T}_m}}(t_0)) \subset \pi_{k_{\mathcal{T}_m}}, \forall m \in \{1, \dots, \bar{T}\}$, $\mathcal{O}_{x_g}(x^O_{x_g}(t_0)) \subset \pi_{k_g}, \forall g \in \mathcal{G}$, $\mathcal{O}_{y_\rho}(x^O_{y_\rho}(t_0)) \subset \pi_{k_\rho}, \forall \rho \in \mathcal{R}$,
- $\mathcal{AG}_{\rho, y_\rho}(q_\rho(t_0), x^O_{y_\rho}(t_0)) = \top, \forall \rho \in \mathcal{R}$, $\mathcal{AG}_{z,0}(q_z(t_0), x^O(t_0)) = \top, \forall z \in \mathcal{Z}$, $\mathcal{AG}_{g,0}(q_g(t_0), x^O(t_0)) = \top, \forall g \in \mathcal{G}$, $\mathcal{AG}_{\tau, s_{\mathcal{T}_m}}(q_\tau(t_0), x^O_{s_{\mathcal{T}_m}}(t_0)) = \top$, $\forall \tau \in \mathcal{T}_m, m \in \{1, \dots, \bar{T}\}$,

which mean, intuitively, that the objects $s_{\mathcal{T}_m}$, x_g, y_ρ to be transported, grasped, released, are in the regions $\pi_{k_{\mathcal{T}_m}}$, π_{k_g}, π_{k_ρ}, respectively, and there is also grasping compliance with the corresponding agents. By also assuming that the agents do not collide with each other or with the objects (except for the transportation/releasing task agents), we guarantee that the conditions of Def. 7.5-7.8 hold.

In the following, we design u_z and u_τ such that $\pi_{k_z} \rightarrow_z \pi_{k'_z}$ and $\pi_{k_{\mathcal{T}_m}} \xrightarrow{T}_{\mathcal{T}_m, s_{\mathcal{T}_m}} \pi_{k'_{\mathcal{T}_m}}$, with $k'_z, k'_{\mathcal{T}_m} \in \mathcal{K}, \forall z \in \mathcal{Z}, m \in \{1, \dots, \bar{T}\}$, assuming that (i) there exist appropriate u_g and u_ρ that guarantee $g \xrightarrow{g} x_g$ and $\rho \xrightarrow{r} y_\rho$ in π_{k_g}, π_{k_ρ}, respectively,

$\forall g \in \mathcal{G}, \rho \in \mathcal{R}$ and (ii) that the agents and objects fit in their respective goal regions, i.e.,

$$\begin{aligned}&\mathcal{P}_{s,0}\Big(r_{\pi_k}, [r_z]_{z\in\mathcal{Q}_{\mathcal{Z},k}}, [r_g]_{g\in\mathcal{Q}_{\mathcal{G},k}}, [r_\rho]_{\rho\in\mathcal{Q}_{\mathcal{R},k}},\\ &[r_{\mathcal{T}_m, s_{\mathcal{T}_m}}]_{m\in\mathcal{Q}_{\mathcal{T},k}}, [r^O_{x_g}]_{g\in\mathcal{Q}_{\mathcal{G},k}}, [r^O_{y_\rho}]_{\rho\in\mathcal{Q}_{R,k}}\Big) = \top \qquad (7.30)\end{aligned}$$

$\forall k \in \mathcal{K}$, where we define the sets: $\mathcal{Q}_{\mathcal{Z},k} := \{z \in \mathcal{Z} : k'_z = k\}, \mathcal{Q}_{\mathcal{G},k} := \{g \in \mathcal{G} : k_g = k\}, \mathcal{Q}_{\mathcal{R},k} := \{\rho \in \mathcal{R} : k_r = k\}, \mathcal{Q}_{\mathcal{T},k} := \{m \in \{1,\dots,\bar{T}\} : k'_{\mathcal{T}_m} = k\}$, that correspond to the indices of the agents and objects that are in region $k \in \mathcal{K}$.

Example 7.1. As an example, consider $N = 6$ agents, $\mathcal{N} = \{1,\dots,6\}$, $M = 3$ objects, $\mathcal{M} = \{1,2,3\}$ in a workspace that contains $K = 4$ regions of interest, $\mathcal{K} = \{1,\dots,4\}$. Let $t_0 = 0$ and, according to Problem 7.2, take $\text{init}(1) = \text{init}(5) = 1, \text{init}(2) = 2, \text{init}(3) = \text{init}(4) = 3$, and $\text{init}(6) = 4$, i.e., agents 1 and 5 are in region $\pi_{\text{init}(1)} = \pi_{\text{init}(5)} = \pi_1$, agent 2 is in region $\pi_{\text{init}(2)} = \pi_2$, agents 3 and 4 are in region $\pi_{\text{init}(3)} = \pi_{\text{init}(4)} = \pi_3$ and agent 6 is in region $\pi_{\text{init}(6)} = \pi_4$. We also consider $\text{init}_O(1) = 1, \text{init}_O(2) = 2, \text{init}_O(3) = 3$ implying that the 3 objects are in regions π_1, π_2 and π_3, respectively. We assume that agents $1,5$ grasp objet 1, and agents $3,4$ grasp object 3, i.e., $\mathcal{AG}_{1,1}(q_1(0), x^O_1(0)) = \mathcal{AG}_{5,1}(q_5(0), x^O_1(0)) = \mathcal{AG}_{3,3}(q_3(0), x^O_3(0)) = \mathcal{AG}_{4,3}(q_4(0), x^O_4(0)) = \mathcal{AG}_{2,0}(q_2(0), x^O(0)) = \mathcal{AG}_{6,0}(q_6(0), x^O(0)) = \top$. Agents 1 and 5 aim to cooperatively transport object 1 to π_4, agent 2 aims to grasp object 2, agents 3 and 4 aim to cooperatively transport object 3 to π_1 and agent 6 aims to perform a transition to region π_2. Therefore, $\mathcal{Z} = \{6\}, \bar{T} = 2, \mathcal{T}_1 = \{1,5\}, \mathcal{T}_2 = \{3,4\}$, $\mathcal{T} = \mathcal{T}_1 \cup \mathcal{T}_2 = \{1,5,4,3\}$, $\mathcal{G} = \{2\}, \mathcal{R} = \emptyset$, $s_{\mathcal{T}_1} = 1$, $s_{\mathcal{T}_2} = 2$, $\mathcal{S} = \{s_{\mathcal{T}_1}, s_{\mathcal{T}_2}\} = \{1,2\}$, $\mathcal{X} = \{x_2\} = \{2\}, \mathcal{Y} = \emptyset$. Moreover, the region indices $k_z, k_\tau, k_g, k_r, k_{\mathcal{T}_m}, k'_z, k'_{\mathcal{T}_m}, z \in \mathcal{Z} = \{6\}, \tau \in \mathcal{T} = \{1,5,4,3\}, g \in \mathcal{G} = \{2\}, r \in \mathcal{R} = \emptyset, m \in \{1,2\}$, take the form $k_6 = 4, k_1 = k_5 = 1, k_2 = 2, k_3 = k_4 = 3, k_{\mathcal{T}_1} = 1, k_{\mathcal{T}_3} = 3, k'_6 = 2, k'_{\mathcal{T}_1} = 4, k'_{\mathcal{T}_2} = 1$. Finally, the actions that need to be performed by the agents are $\pi_1 \xrightarrow{T}_{\mathcal{T}_{1,1}} \pi_4$, $2 \xrightarrow{g} 2$, $\pi_3 \xrightarrow{T}_{\mathcal{T}_{2,3}} \pi_1$ and $\pi_4 \rightarrow \pi_2$.

Next, for each region π_k, we compute from $\mathcal{P}_s$ a set of configurations for the agents and objects in this region. More specifically,

$$\begin{aligned}&([q^\star_z]_{z\in\mathcal{Q}_{\mathcal{Z},k}}, [q^\star_g]_{g\in\mathcal{Q}_{\mathcal{G},k}}, [q^\star_\rho]_{\rho\in\mathcal{Q}_{\mathcal{R},k}}, [x^{O\star}_{s_{\mathcal{T}_m}}]_{m\in\mathcal{Q}_{\mathcal{T},k}}, [x^{O\star}_{x_g}]_{g\in\mathcal{Q}_{\mathcal{G},k}}, [x^{O\star}_{y_\rho}]_{\rho\in\mathcal{Q}_{\mathcal{R},k}}) =\\ &\mathcal{P}_{s,1}\Big(r_{\pi_k}, [r_z]_{z\in\mathcal{Q}_{\mathcal{Z},k}}, [r_g]_{g\in\mathcal{Q}_{\mathcal{G},k}}, [r_\rho]_{\rho\in\mathcal{Q}_{\mathcal{R},k}}, [r_{\mathcal{T}_m, s_{\mathcal{T}_m}}]_{m\in\mathcal{Q}_{\mathcal{T},k}}, [r^O_{x_g}]_{g\in\mathcal{Q}_{\mathcal{G},k}}, [r^O_{y_\rho}]_{\rho\in\mathcal{Q}_{R,k}}\Big),\end{aligned}$$

where we have used the notation of (7.30). Hence, we now have the goal configurations for the agents $\mathcal{Z}$ performing the transitions as well as agents $\mathcal{T}$ performing the cooperative transportations.

Following Section 2.3.1, we define the error functions $\gamma_z : \mathbb{R}^{n_\mathcal{Z}} \to \mathbb{R}_{\geq 0}$ with $\gamma_z(q_z) := \|q_z - q^\star_z\|^2$, $\forall z \in \mathcal{Z}$, $n_\mathcal{Z} := \sum_{z\in\mathcal{Z}} n_z$, and $\gamma_{\mathcal{T}_m} : \mathbb{M} \to \mathbb{R}_{\geq 0}$ as $\gamma_{\mathcal{T}_m}(x^O_{s_{\mathcal{T}_m}}) := \|p^O_{s_{\mathcal{T}_m}} - p^{O\star}_{s_{\mathcal{T}_m}}\|^2$, where $p^{O\star}_{s_{\mathcal{T}_m}}$ is the position part of $x^{O\star}_{s_{\mathcal{T}_m}}$.

Regarding the collision avoidance, we have the following collision functions:

$$\begin{aligned}
&\beta_{i,l}(q_i, q_l) := \|c_i(q_i) - c_l(q_l)\|^2 - (r_i + r_l)^2, \forall i, l \in \mathcal{N}\backslash\mathcal{T}, i \neq l,\\
&\beta_{i,O_j}(q_i) := \|c_i(q_i) - p_j^O\|^2 - (r_i + r_j^O)^2, \forall i \in \mathcal{N}\backslash\mathcal{T}, j \in \mathcal{M}\backslash\mathcal{S}\\
&\beta_{i,\mathcal{T}_m}(q_i, x^O_{s_{\mathcal{T}_m}}) := \|c_i(q_i) - p^O_{s_{\mathcal{T}_m}}\|^2 - (r_i + r_{\mathcal{T}_m, s_{\mathcal{T}_m}})^2, \forall i \in \mathcal{N}\backslash\mathcal{T}, m \in \{1, \ldots, \bar{T}\},\\
&\beta_{\mathcal{T}_m,\mathcal{T}_\ell}(x^O_{s_{\mathcal{T}_m}}, x^O_{s_{\mathcal{T}_\ell}}) := \|p^O_{s_{\mathcal{T}_m}} - p^O_{s_{\mathcal{T}_\ell}}\|^2 - (r_{\mathcal{T}_m, s_{\mathcal{T}_m}} + r_{\mathcal{T}_\ell, s_{\mathcal{T}_\ell}})^2, \forall m, \ell \in \{1, \ldots, \bar{T}\}, m \neq \ell,\\
&\beta_{\mathcal{T}_m,O_j}(x^O_{s_{\mathcal{T}_m}}) := \|p^O_{s_{\mathcal{T}_m}} - p_j^O\|^2 - (r_{\mathcal{T}_m, s_{\mathcal{T}_m}} + r_j^O)^2, \forall m \in \{1, \ldots, \bar{T}\}, j \in \mathcal{M}\backslash\mathcal{S},\\
&\beta_{i,\pi_k}(q_i) := \|c_i(q_i) - p_{\pi_k}\|^2 - (r_i + r_{\pi_k})^2, \forall i \in \mathcal{Z}, k \in \mathcal{K}\backslash\{k_z, k'_z\},\\
&\beta_{\mathcal{T}_m,\pi_k}(x^O_{s_{\mathcal{T}_m}}) := \|p^O_{s_{\mathcal{T}_m}} - p_{\pi_k}\|^2 - (r_{\mathcal{T}_m, s_{\mathcal{T}_m}} + r_{\pi_k})^2, \forall m \in \{1, \ldots, \bar{T}\}, k \in \mathcal{K}\backslash\{k_{\mathcal{T}_m}, k'_{\mathcal{T}_m}\},\\
&\beta_{i,\mathcal{W}}(q_i) := (r_0 - r_i)^2 - \|c_i(q_i)\|^2, \forall i \in \mathcal{N}\backslash\mathcal{T}\\
&\beta_{\mathcal{T}_m,\mathcal{W}}(x^O_{s_{\mathcal{T}_m}}) := (r_0 - r_{\mathcal{T}_m, s_{\mathcal{T}_m}})^2 - \|p^O_{s_{\mathcal{T}_m}}\|^2, \forall m \in \{1, \ldots, \bar{T}\},
\end{aligned}$$

that incorporate collisions among the navigating agents, the navigating agents and the objects, the transportation agents, the transportation agents and the objects, the navigating agents and the undesired regions, the transportation agents and the undesired regions, the navigating agents and the workspace boundary, and the transportation agents and the workspace boundary, respectively. Therefore, by following the procedure described in Section 2.3.1, we can form the total obstacle function $G : \mathbb{R}^{n_{\mathcal{Z}}} \times \mathbb{M}^{|\mathcal{S}|} \to \mathbb{R}_{\geq 0}$ and thus, define the navigation function [35, 36] $\varphi : \mathbb{R}^{n_{\mathcal{Z}}} \times \mathbb{M}^{|\mathcal{S}|} \to [0, 1]$ as

$$\varphi(q_{\mathcal{Z}}, x^O_{\mathcal{S}}) := \frac{\gamma(q_{\mathcal{Z}}, x^O_{\mathcal{S}})}{\Big([\gamma(q_{\mathcal{Z}}, x^O_{\mathcal{S}})]^\kappa + G(q_{\mathcal{Z}}, x^O_{\mathcal{S}})\Big)^{\frac{1}{\kappa}}},$$

where $x^O_{\mathcal{S}} := [(x^O_{s_{\mathcal{T}_m}})^\top]^\top_{m \in \{1,\ldots,\bar{T}\}} \in \mathbb{M}^{|\mathcal{S}|}$, $\gamma(q_{\mathcal{Z}}, x^O_{\mathcal{S}}) := \sum_{z \in \mathcal{Z}} \gamma_z(q_z) + \sum_{m \in \{1,\ldots,\bar{T}\}} \gamma_{\mathcal{T}_m}(x^O_{s_{\mathcal{T}_m}})$ and $\kappa > 0$ is a positive gain used to derive the proof correctness of φ [35, 36]. Note that, a sufficient condition for avoidance of the undesired regions and avoidance of collisions and singularities is $\varphi(q_{\mathcal{Z}}, x^O_{\mathcal{S}}) < 1$.

Next, we design the feedback control protocols $u_z : \mathbb{R}^{n_z} \times \mathbb{R}^{n_z} \to \mathbb{R}^6, u_\tau : \mathbb{S}_\tau \times \mathbb{S}^O_{s_{\mathcal{T}_m}} \times \mathbb{R}^6$, $\forall z \in \mathcal{Z}, \tau \in \mathcal{T}_m, m \in \{1, \ldots, \bar{T}\}$ as follows:

$$\begin{aligned}
&u_z(q_z, x^O_{\mathcal{S}}, \dot{q}_z) = g_{q_z}(q_z) - \nabla_{q_z}\varphi(q_{\mathcal{Z}}, x^O_{\mathcal{S}}) - K_z\dot{q}_z, &(7.31a)\\
&u_\tau(q_\tau, x^O_{s_{\mathcal{T}_m}}, v^O_{s_{\mathcal{T}_m}}) = [J^O_{\tau, s_{\mathcal{T}_m}}(q_\tau)]^{-\top}\Big\{c_\tau\Big(g_O(x^O_{s_{\mathcal{T}_m}}) -\\
&[J^O_{s_{\mathcal{T}_m}}(x^O_{s_{\mathcal{T}_m}})]^\top \nabla_{x^O_{s_{\mathcal{T}_m}}}\varphi(q_{\mathcal{Z}}, x^O_{\mathcal{S}}) - v^O_{s_{\mathcal{T}_m}}\Big)\Big\} + g_\tau(q_\tau), &(7.31b)
\end{aligned}$$

where c_τ are load sharing coefficients, with the properties $c_\tau > 0$, $\forall \tau \in \mathcal{T}_m$, $\sum_{\tau \in \mathcal{T}_m} c_\tau = 1$, $\forall m \in \{1, \ldots, \bar{T}\}$, $K_z = \text{diag}\{k_z\} \in \mathbb{R}^{n_z \times n_z}$, with $k_z > 0, \forall z \in \mathcal{Z}$, is a constant positive definite gain matrix. The proof of convergence of the closed loop system is stated in the next Lemma.

Lemma 7.3. *Consider the sets of agent $\mathcal{Z}, \mathcal{T}, \mathcal{G}, \mathcal{R}$ and the set of objects $\mathcal{S}, \mathcal{X}, \mathcal{R}$ in their respective regions interest, as defined above, described by the dynamics* (7.23), (7.24), (7.29) *at* $t_0 > 0$*. Then, under the assumptions that: (i) the actions* $g \xrightarrow{g} x_g, \rho \xrightarrow{r} y_\rho$ *are guaranteed, (ii)* (7.30) *holds and (iii) the robots and objects operate in singularity-free (kinematic- and representation ones, respectively) configurations, the control protocols* (7.31) *guarantee the existence of a* $t_f > t_0$ *such that* $\pi_{k_z} \rightarrow_z \pi_{k'_z}$ *and* $\pi_{k_{\mathcal{T}_m}} \xrightarrow{T}_{\mathcal{T}_m, s_{\mathcal{T}_m}} \pi_{k'_{\mathcal{T}_m}}, \forall z \in \mathcal{Z}, m \in \{1, \dots, \bar{T}\}$*, according to Def. 7.5 and 7.8, respectively.*

Proof. Define $\bar{\mathcal{T}} \coloneqq \{1, \dots, \bar{T}\}$ and, following the notation of Section 7.3.1, consider the stacked vector states $x_{\mathcal{T}_m, s_{\mathcal{T}_m}} \coloneqq [q_{\mathcal{T}_m}^\top, \dot{q}_{\mathcal{T}_m}^\top, (x^O_{s_{\mathcal{T}_m}})^\top, (v^O_{s_{\mathcal{T}_m}})^\top]^\top, m \in \bar{\mathcal{T}}$, $d \coloneqq [q_{\mathcal{Z}}^\top, \dot{q}_{\mathcal{Z}}^\top, [x^\top_{\mathcal{T}_m, s_{\mathcal{T}_m}}]^\top_{m \in \bar{\mathcal{T}}}]^\top$ as well as the domain: $\mathbb{D} \coloneqq \mathbb{R}^{n_{\mathcal{Z}}} \times \mathbb{R}^{n_{\mathcal{Z}}} \times \mathbb{S}_{\mathcal{T}_1} \times \cdots \times \mathbb{S}_{\mathcal{T}_{\bar{T}}} \times \mathbb{R}^{n_{\mathcal{T}}} \times \mathbb{S}^O_{s_{\mathcal{T}_1}} \times \cdots \times \mathbb{S}^O_{s_{\mathcal{T}_{\bar{T}}}} \times \mathbb{R}^{6|\mathcal{S}|}$, where $\mathbb{S}_{\mathcal{T}_m} \coloneqq \prod_{\tau \in \mathcal{T}_m} \mathbb{S}_\tau, \forall m \in \bar{\mathcal{T}}$, and $n_{\mathcal{T}} \coloneqq \sum_{m \in \bar{\mathcal{T}}} \sum_{\tau \in \mathcal{T}_m} n_\tau$. Consider now the candidate Lyapunov function $V : \mathbb{D} \to \mathbb{R}_{\geq 0}$, with

$$V(d) = \varphi(q_{\mathcal{Z}}, x^O_{\mathcal{S}}) + \frac{1}{2} \sum_{z \in \mathcal{Z}} \dot{q}_z^T B_z(q_z) \dot{q}_z + \frac{1}{2} \sum_{m \in \bar{\mathcal{T}}} [v^O_{s_{\mathcal{T}_m}}]^\top \widetilde{M}_{\mathcal{T}_m, s_{\mathcal{T}_m}}(x_{\mathcal{T}_m, s_{\mathcal{T}_m}}) v^O_{s_{\mathcal{T}_m}}.$$

Note that, since no collisions occur and the robots and objects have zero velocity at t_0, we conclude that $V_0 \coloneqq V(d(t_0)) = \varphi(q_{\mathcal{Z}}(t_0), x^O_{\mathcal{S}}(t_0)) =: \varphi_0 < 1$, and hence $d(t_0) \in \widetilde{\mathbb{D}} \coloneqq \{d \in \mathbb{D} : \varphi(q_{\mathcal{Z}}, x^O_{\mathcal{S}}) \leq \varphi_0 < 1\}$. By considering the closed loop system $\frac{\partial}{\partial t} d = f_{\text{cl}}(d)$ (An explicit expression for f_{cl} can be obtained by combining (7.23), (7.29), (7.31)), we can verify the locally Lipschitz property of f_{cl}, and thus the existence of a unique maximal solution $d : [t_0, t_{\max}) \to \widetilde{\mathbb{D}}$, for a finite time instant $t_{\max} > t_0$. By differentiating V and substituting (7.23), (7.29), we obtain

$$\begin{aligned}
\dot{V} = &\sum_{z \in \mathcal{Z}} \left\{ [\nabla_{q_z} \varphi(q_{\mathcal{Z}}, x^O_{\mathcal{S}})]^\top \dot{q}_z + \dot{q}_z^\top \Big(\tau_z - N_z(q_z, \dot{q}_z)\dot{q}_z - g_{q_z}(q_z) \Big) + \frac{1}{2} \dot{q}_z^\top \dot{M}_z(q_z) \dot{q}_z \right\} \\
&+ \sum_{m \in \bar{\mathcal{T}}} \Bigg\{ [\nabla_{x^O_{s_{\mathcal{T}_m}}} \varphi(q_{\mathcal{Z}}, x^O_{\mathcal{S}})]^\top \dot{x}^O_{s_{\mathcal{T}_m}} + [v^O_{s_{\mathcal{T}_m}}]^\top \Bigg(\sum_{\tau \in \mathcal{T}_m} [J^O_{\tau, s_{\mathcal{T}_m}}(q_\tau)]^\top u_\tau - g_O(x^O_{s_{\mathcal{T}_m}}) v^O_{s_{\mathcal{T}_m}} \\
&- \sum_{\tau \in \mathcal{T}_m} [J^O_{\tau, s_{\mathcal{T}_m}}(q_\tau)]^\top g_\tau(q_\tau) - \widetilde{C}_{\mathcal{T}_m, s_{\mathcal{T}_m}}(x_{\mathcal{T}_m, s_{\mathcal{T}_m}}) \Bigg) + \frac{1}{2} [v^O_{s_{\mathcal{T}_m}}]^\top \dot{\widetilde{M}}_{\mathcal{T}_m, s_{\mathcal{T}_m}}(x_{\mathcal{T}_m, s_{\mathcal{T}_m}}) v^O_{s_{\mathcal{T}_m}} \Bigg\},
\end{aligned}$$

$\forall d \in \widetilde{\mathbb{D}}$, where we have also used the fact that $f_z = 0, \forall z \in \mathcal{Z}$, since the agents performing transportation actions are not in contact with any objects (and there are no collisions in $\widetilde{\mathbb{D}}$). By employing Lemma 7.1 as well as (7.24a), $\dot{V}$ becomes:

$$\begin{aligned}
\dot{V} = &\sum_{z \in \mathcal{Z}} \dot{q}_z^\top \Big(\nabla_{q_z} \varphi(q_{\mathcal{Z}}, x^O_{\mathcal{S}}) + \tau_z - g_{q_z}(q_z) \Big) \sum_{m \in \bar{\mathcal{T}}} [v^O_{s_{\mathcal{T}_m}}]^\top \Bigg([J^O_{s_{\mathcal{T}_m}}(x^O_{s_{\mathcal{T}_m}})]^\top \nabla_{x^O_{s_{\mathcal{T}_m}}} \varphi(q_{\mathcal{Z}}, x^O_{\mathcal{S}}) + \\
&\sum_{\tau \in \mathcal{T}_m} [J^O_{\tau, s_{\mathcal{T}_m}}(q_\tau)]^\top u_\tau - \sum_{\tau \in \mathcal{T}_m} [J^O_{\tau, s_{\mathcal{T}_m}}(q_\tau)]^\top g_\tau(q_\tau) - g_O(x^O_{s_{\mathcal{T}_m}}) \Bigg),
\end{aligned}$$

and after substituting (7.31): $\dot{V} = -\sum_{z\in\mathcal{Z}} \dot{q}_z K_z \dot{q}_z - \sum_{m\in\widetilde{\mathcal{T}}} \|v^O_{s_{\mathcal{T}_m}}\|^2$, $\forall d \in \widetilde{\mathbb{D}}$, which is strictly negative unless $\dot{q}_z = 0, v^O_{s_{\mathcal{T}_m}} = 0, \forall z \in \mathcal{Z}, m \in \widetilde{\mathcal{T}}$. Since $J^O_{\tau, s_{\mathcal{T}_m}}(q_\tau)$ is always non-singular, and $J_\tau(q_\tau(t))$ has full-rank by assumption for the maximal solution, $\forall \tau \in \mathcal{T}_m, m \in \widetilde{\mathcal{T}}$, the latter implies also that $\dot{q}_\tau = 0$, $\forall \tau \in \mathcal{T}_m, m \in \widetilde{\mathcal{T}}$. Hence, $V(d(t)) \leq V_0 < 1$, $\forall t \in [t_0, t_{\max})$, which suggests that $\varphi(q_{\mathcal{Z}}(t), x^O_{\mathcal{S}}(t)) \leq \varphi_0 < 1$ and $d(t) \in \widetilde{\mathbb{D}}$, $\forall t \in [t_0, t_{\max})$. Therefore, since $\widetilde{\mathbb{D}}$ is compact, the solution $d(t)$ is defined over the entire time horizon in $\widetilde{\mathbb{D}}$ [28], i.e. $d : [t_0, \infty) \to \widetilde{\mathbb{D}}$. Moreover, according to La Salle's Invariance Principle [28], the system will converge to the largest invariant set contained in the set $\{d \in \widetilde{\mathbb{D}} : \dot{q}_z = 0, v^O_{s_{\mathcal{T}_m}} = 0, \forall z \in \mathcal{Z}, m \in \widetilde{\mathcal{T}}\}$. In order for this set to be invariant, we require that $\ddot{q}_z = 0$, $\dot{v}^O_{s_{\mathcal{T}_m}} = 0$, which, by employing (7.31), (7.23), (7.29), and the assumption of non-singular $J^O_{s_{\mathcal{T}_m}}(x^O_{s_{\mathcal{T}_m}}(t))$, $\forall t \in \mathbb{R}_{\geq 0}$, implies that $\nabla_{q_z}\varphi(q_{\mathcal{Z}}, x^O_{\mathcal{S}}) = 0$, $\nabla_{x^O_{s_{\mathcal{T}_m}}}\varphi(q_{\mathcal{Z}}, x^O_{\mathcal{S}}) = 0$, $\forall z \in \mathcal{Z}, m \in \widetilde{\mathcal{T}}$. Since φ is a navigation function [36], this condition is true only at the destination configurations (i.e., where $\gamma(q_{\mathcal{Z}}, x^O_{\mathcal{S}}) = 0$) and a set of isolated saddle points. By choosing κ sufficiently large, the region of attraction of the saddle points is a set of measure zero [35, 114]. Thus, the system converges to the destination configuration from almost everywhere, i.e., $\|q_z(t) - q^\star_z\| \to 0$ and $\|p^O_{s_{\mathcal{T}_m}}(t) - p^{O^\star}_{s_{\mathcal{T}_m}}\| \to 0$. Therefore, there exist finite time instants $t_{f_z}, t_{f_m} > t_0$, such that $\mathcal{A}_z(q_z(t_{f_z})) \subset \pi_{k'_z}$ and $\mathcal{A}_\tau(q_\tau(t_{f_m})), \mathcal{O}_{s_{\mathcal{T}_m}}(x^O_{s_{\mathcal{T}_m}}(t_{f_v})) \subset \pi_{k'_{\mathcal{T}_m}}$, with inter-agent collision avoidance, $\forall z \in \mathcal{Z}, \tau \in \mathcal{T}_m, m \in \widetilde{\mathcal{T}}$. Since the actions $g \xrightarrow{g} x_g$, $\rho \xrightarrow{r} y_\rho$ are also performed, we denote as t_{f_g}, t_{f_ρ} the times that these actions have been completed, $g \in \mathcal{G}, \rho \in \mathcal{R}$. Hence, by setting $t_f := \max\{\max\limits_{z\in\mathcal{Z}} t_{f_z}, \max\limits_{m\in\widetilde{\mathcal{T}}} t_{f_m}, \max\limits_{g\in\mathcal{G}} t_{f_g}, \max\limits_{\rho\in\mathcal{R}} t_{f_\rho}\}$, all the actions of all agents will be completed at t_f. □

Remark 7.3. We could modify the dynamic model (7.24) by employing the physical acceleration $\ddot{x}^O_j$ instead of the generalized accelerations $\dot{v}^O_j$, $j \in \mathcal{M}$. In that way, we would avoid using the term J^O_j and hence ensure that representation singularities (when $|\eta^O_{j,2}| = \frac{\pi}{2}$) do not affect our scheme. Note that the actual difference lies in the use of $\dot{\eta}^O_j$ instead of $\omega^O_j, j \in \mathcal{M}$. Feedback, however, of $\dot{\eta}^O_j$ is not a realistic assumption, since most sensors provide on-line measurements of the angular velocity ω^O_j and hence, the conversion via J^O_j cannot be avoided.

Remark 7.4. The fact that we consider fully actuated holonomic mobile bases is not restrictive, since a similar analysis can be performed for non-holonomic agents (see [105]). Note also that in our analysis we do not take into account potential collisions between agents that grasp and transport the same object, since we just consider the bounded spherical volume of the system. This specification constitutes part of our ongoing work.

High-Level Plan Generation

The second part of the solution is the derivation of a high-level plan that satisfies the given LTL formulas ϕ_i and ϕ^O_j and can be generated by using standard techniques from automata-based formal verification methodologies. Thanks to (i) the proposed control laws that allow agent transitions and object transportations $\pi_k \rightarrow_i \pi_{k'}$ and $\pi_k \xrightarrow{T}_{\mathcal{T},j} \pi_{k'}$, respectively, and (ii) the off-the-self control laws that guarantee grasp and release actions $i \xrightarrow{g} j$ and $i \xrightarrow{r} j$, we can abstract the behavior of the agents using a finite transition system as presented in the sequel.

Definition 7.9. The coupled behavior of the overall system of all the N agents and M objects is modeled by the transition system $\mathcal{TS} = (\Pi_s, \Pi_s^{\text{init}}, \rightarrow_s, \mathcal{AG}, \Psi, \mathcal{L}, \Lambda, P_s, \chi)$, where

1. $\Pi_s \subset \bar{\Pi} \times \bar{\Pi}^O \times \bar{\mathcal{AG}}$ is the set of states; $\bar{\Pi} := \Pi_1 \times \cdots \times \Pi_N$ and $\bar{\Pi}^O := \Pi^O_1 \times \cdots \times \Pi^O_M$ are the set of states-regions that the agents and the objects can be at, with $\Pi_i = \Pi^O_j = \Pi, \forall i \in \mathcal{N}, j \in \mathcal{M}$; $\mathcal{AG} := \mathcal{AG}_1 \times \cdots \times \mathcal{AG}_N$ is the set of boolean grasping variables introduced in Section 7.3.1, with $\mathcal{AG}_i := \{\mathcal{AG}_{i,0}\} \cup \{[\mathcal{AG}_{i,j}]_{j\in\mathcal{M}}\}, \forall i \in \mathcal{N}$. By defining $\bar{\pi} := (\pi_{k_1}, \cdots, \pi_{k_N}), \bar{\pi}_O := (\pi_{k^O_1}, \cdots, \pi_{k^O_M}), \bar{w} = (w_1, \cdots, w_N)$, with $\pi_{k_i}, \pi_{k^O_j} \in \Pi$ (i.e., $k_i, k^O_j \in \mathcal{K}, \forall i \in \mathcal{N}, j \in \mathcal{M}$) and $w_i \in \mathcal{AG}_i, \forall i \in \mathcal{N}$, then the coupled state $\pi_s := (\bar{\pi}, \bar{\pi}_O, \bar{w})$ belongs to Π_s, i.e., $(\bar{\pi}, \bar{\pi}_O, \bar{w}) \in \Pi_s$ if

   a) $\mathcal{P}_{s,0}\Big(r_{\pi_k}, [r_i]_{i\in\{i\in\mathcal{N}:k_i=k\}}, [r^O_j]_{j\in\{j\in\mathcal{M}:k^O_j=k\}}\Big) = \top$, i.e., the respective agents and objects fit in the region, $\forall k \in \mathcal{K}$,

   b) $k_i = k^O_j$ for all $i \in \mathcal{N}, j \in \mathcal{M}$ such that $w_i = \mathcal{AG}_{i,j} = \top$, i.e., an agent must be in the same region with the object it grasps,

2. $\Pi_s^{\text{init}} \subset \Pi_s$ is the initial set of states at $t = 0$, which, owing to **(i)**, satisfies the conditions of Problem 7.2,

3. $\rightarrow_s \subset \Pi_s \times \Pi_s$ is a transition relation defined as follows: given the states $\pi_s, \widetilde{\pi}_s \in \Pi$, with

$$\begin{aligned} \pi_s :=& (\bar{\pi}, \bar{\pi}_O, \bar{w}) := (\pi_{k_1}, \ldots, \pi_{k_N}, \pi_{k^O_1}, \ldots, \pi_{k^O_M}, w_1, \ldots, w_N), \\ \widetilde{\pi}_s :=& (\widetilde{\bar{\pi}}, \widetilde{\bar{\pi}}_O, \widetilde{\bar{w}}) := (\pi_{\widetilde{k}_1}, \ldots, \pi_{\widetilde{k}_N}, \pi_{\widetilde{k}^O_1}, \ldots, \pi_{\widetilde{k}^O_1}, \widetilde{w}_1, \ldots, \widetilde{w}_N), \end{aligned} \tag{7.32}$$

   a transition $\pi_s \rightarrow_s \widetilde{\pi}_s$ occurs if all the following hold:

   a) $\nexists i \in \mathcal{N}, j \in \mathcal{M}$ such that $w_i = \mathcal{AG}_{i,j} = \top$, $\widetilde{w}_i = \mathcal{AG}_{i,0} = \top$, (or $w_i = \mathcal{AG}_{i,0} = \top$, $\widetilde{w}_i = \mathcal{AG}_{i,j} = \top$) and $k_i \neq \widetilde{k}_i$, i.e., there are no simultaneous grasp/release and navigation actions,

b) $\nexists i \in \mathcal{N}, j \in \mathcal{M}$ such that $w_i = \mathcal{AG}_{i,j} = \top$, $\widetilde{w}_i = \mathcal{AG}_{i,0} = \top$, (or $w_i = \mathcal{AG}_{i,0} = \top$, $\widetilde{w}_i = \mathcal{AG}_{i,j} = \top$) and $k_i = k^O_j \neq \widetilde{k}_i = \widetilde{k}^O_j$, i.e., there are no simultaneous grasp/release and transportation actions,

c) $\nexists i \in \mathcal{N}, j, j' \in \mathcal{M}$, with $j \neq j'$, such that $w_i = \mathcal{AG}_{i,j} = \top$ and $\widetilde{w}_i = \mathcal{AG}_{i,j'} = \top$ ($w_i = \mathcal{AG}_{i,j'} = \top$ and $\widetilde{w}_i = \mathcal{AG}_{i,j'} = \top$), i.e., there are no simultaneous grasp and release actions,

d) $\nexists j \in \mathcal{M}$ such that $k^O_j \neq \widetilde{k}^O_j$ and $w_i \neq \mathcal{AG}_{i,j}, \forall i \in \mathcal{N}$ (or $\widetilde{w}_i \neq \mathcal{AG}_{i,j}, \forall i \in \mathcal{N}$), i.e., there is no transportation of a non-grasped object,

e) $\nexists j \in \mathcal{M}, \mathcal{T} \subseteq \mathcal{N}$ such that $k^O_j \neq \widetilde{k}^O_j$ and $\Lambda(m^O_j, \zeta_{\mathcal{T}}) = \bot$, where $w_i = \widetilde{w}_i = \mathcal{AG}_{i,j} = \top \Leftrightarrow i \in \mathcal{T}$, i.e., the agents grasping an object are powerful enough to transfer it,

4. $\Psi := \bar{\Psi} \cup \bar{\Psi}^O$ with $\bar{\Psi} = \bigcup_{i \in \mathcal{N}} \Psi_i$ and $\bar{\Psi}^O = \bigcup_{j \in \mathcal{M}} \Psi^O_j$, are the atomic propositions of the agents and objects, respectively, as defined in Section 7.3.1.

5. $\mathcal{L} : \Pi_s \to 2^\Psi$ is a labeling function defined as follows: Given a state π_s as in (7.32) and $\psi_s := \left(\bigcup_{i \in \mathcal{N}} \psi_i\right) \bigcup \left(\bigcup_{j \in \mathcal{M}} \psi^O_j\right)$ with $\psi_i \in 2^{\Psi_i}, \psi^O_j \in 2^{\Psi^O_j}$, then $\psi_s \in \mathcal{L}(\pi_s)$ if $\psi_i \in \mathcal{L}_i(\pi_{k_i})$ and $\psi^O_j \in \mathcal{L}^O_j(\pi_{k^O_j}), \forall i \in \mathcal{N}, j \in \mathcal{M}$.

6. Λ and P_s as defined in Section 7.3.1.

7. $\chi : (\to_s) \to \mathbb{R}_{\geq 0}$ is a function that assigns a cost to each transition $\pi_s \to_s \widetilde{\pi}_s$. This cost might be related to the distance of the agents' regions in π_s to the ones in $\widetilde{\pi}_s$, combined with the cost efficiency of the agents involved in transport tasks (according to $\zeta_i, i \in \mathcal{N}$).

Next, we form the global LTL formula $\phi := (\wedge_{i \in \mathcal{N}} \phi_i) \wedge (\wedge_{j \in \mathcal{M}} \phi^O_j)$ over the set Ψ. Then, we translate ϕ to a Büchi Automaton $\mathcal{BA}$ and we build the product $\widetilde{\mathcal{TS}} := \mathcal{TS} \times \mathcal{BA}$. Using basic graph-search theory, we can find the accepting runs of $\widetilde{\mathcal{TS}}$ that satisfy ϕ and minimize the total cost χ. These runs are directly projected to a sequence of desired states to be visited in the $\mathcal{TS}$. Although the semantics of LTL are defined over infinite sequences of services, it can be proven that there always exists a high-level plan that takes the form of a finite state sequence followed by an infinite repetition of another finite state sequence. For more details on the followed technique, the reader is referred to the related literature, e.g., [38].

Following the aforementioned methodology, we obtain a high-level plan as sequences of states and atomic propositions $\pi_{\text{pl}} := \pi_{s,1}\pi_{s,2}\ldots$ and $\psi_{\text{pl}} := \psi_{s,1}\psi_{s,1}\ldots \models \phi$, which minimizes the cost χ, with

$$\pi_{s,\ell} := (\bar{\pi}_\ell, \bar{\pi}_{O,\ell}, \bar{w}_\ell) \in \Pi_s, \forall \ell \in \mathbb{N},$$

$$\psi_{s,\ell} := \Big(\bigcup_{i \in \mathcal{N}} \psi_{i,\ell}\Big) \bigcup \Big(\bigcup_{j \in \mathcal{M}} \psi^O_{j,\ell}\Big) \in 2^\Psi, \mathcal{L}(\pi_{s,\ell}), \forall \ell \in \mathbb{N},$$

where

- $\bar{\pi}_\ell \coloneqq \pi_{k_{1,\ell}}, \pi_{k_{2,\ell}}, \ldots$ with $k_{i,\ell} \in \mathcal{K}, \forall i \in \mathcal{N}$,
- $\bar{\pi}_{O,\ell} \coloneqq \pi_{k^O_{1,\ell}}, \pi_{k^O_{2,\ell}}, \ldots$ with $k^O_{j,\ell} \in \mathcal{K}, \forall j \in \mathcal{M}$,
- $\bar{w}_\ell \coloneqq w_{1,\ell}, w_{2,\ell}, \ldots$ with $w_{i,\ell} \in \mathcal{AG}_i, \forall i \in \mathcal{N}$,
- $\psi_{i,\ell} \in 2^{\Psi_i}, \mathcal{L}_i(\pi_{k_{i,\ell}}), \forall i \in \mathcal{N}$,
- $\psi^O_{j,\ell} \in 2^{\Psi^O_j}, \mathcal{L}^O_j(\pi_{k^O_{j,\ell}}), \forall j \in \mathcal{M}$.

The path π_{pl} is then projected to the individual sequences of the regions $\pi_{k^O_{j,1}} \pi_{k^O_{j,2}} \ldots$ for each object $j \in \mathcal{M}$, as well as to the individual sequences of the regions $\pi_{k_{i,1}} \pi_{k_{i,2}} \ldots$ and the boolean grasping variables $w_{i,1} w_{i,2} \ldots$ for each agent $i \in \mathcal{N}$. The aforementioned sequences determine the behavior of agent $i \in \mathcal{N}$, i.e., the sequence of actions (transition, transportation, grasp, release or stay idle) it must take.

By the definition of $\mathcal{L}$ in Def. 7.9, we obtain that $\psi_{i,\ell} \in \mathcal{L}_i(\pi_{k_{i,\ell}}), \psi^O_{j,\ell} \in \mathcal{L}^O_j(\pi_{k^O_{j,\ell}}), \forall i \in \mathcal{N}, j \in \mathcal{M}, \ell \in \mathbb{N}$. Therefore, since $\phi = (\wedge_{i \in \mathcal{N}} \phi_i) \wedge (\wedge_{j \in \mathcal{M}} \phi_{O_j})$ is satisfied by ψ, we conclude that $\psi_{i,1} \psi_{i,2} \ldots \models \phi_i$ and $\psi^O_{j,1} \psi^O_{j,2} \ldots \models \phi^O_j, \forall i \in \mathcal{N}, j \in \mathcal{M}$.

The sequences $\pi_{k_{i,1}} \pi_{k_{i,2}} \ldots, \psi_{i,1} \psi_{i,2} \ldots$ and $\pi_{k^O_{j,1}} \pi_{k^O_{j,2}} \ldots, \psi^O_{j,1} \psi^O_{j,2} \ldots$ over $\Pi, 2^{\Psi_i}$ and $\Pi, 2^{\Psi^O_j}$, respectively, produce the trajectories $q_i(t)$ and $x^O_j(t), \forall i \in \mathcal{N}, j \in \mathcal{M}$. The corresponding behaviors are $\beta_i = (q_i(t), \sigma_i) = (q_i(t_{i,1}), \sigma_{i,1})(q_i(t_{i,2}), \sigma_{i,2}) \ldots$ and $\beta^O_j = (x^O_j(t), \sigma^O_j) = (x^O_j(t^O_{j,1}), \sigma^O_{j,1})(x^O_j(t^O_{j,2}), \sigma^O_{j,2}) \ldots$, respectively, according to Section 7.3.1, with $\mathcal{A}_i(q_i(t_{i,\ell})) \subset \pi_{k_{i,\ell}}, \sigma_{i,\ell} \in \mathcal{L}_i(\pi_{k_{i,\ell}})$ and $\mathcal{O}_j(x_{O_j}(t_{O_{j,m}})) \in \pi_{k^O_{j,\ell}}, \sigma^O_{j,\ell} \in \mathcal{L}^O_j(\pi_{k^O_{j,\ell}})$. Thus, it is guaranteed that $\sigma_i \models \phi_i, \sigma^O_j \models \phi^O_j$ and consequently, the behaviors β_i and β^O_j satisfy the formulas ϕ_i and ϕ^O_j, respectively, $\forall i \in \mathcal{N}, j \in \mathcal{M}$. The aforementioned reasoning is summarized in the next theorem:

Theorem 7.3. *The execution of the path (π_{pl}, ψ_{pl}) of $\mathcal{TS}$ guarantees behaviors β_i, β^O_j that yield the satisfaction of ϕ_i and ϕ^O_j, respectively, $\forall i \in \mathcal{N}, j \in \mathcal{M}$, providing, therefore, a solution to Problem 7.2.*

Remark 7.5. Note that although the overall set of states of $\mathcal{TS}$ increases exponentially with respect to the number of agents/objects/regions, some states are not reachable, due to our constraints for the object transportation and the size of the regions, reducing thus the state complexity.

7.3.3 Simulation Results

In this section we demonstrate our approach with computer simulations. We consider a workspace of radius $r_0 = 30\text{m}$, with $K = 4$ regions of interest or radius $r_{\pi_k} = 3.5\text{m}$, $\forall k \in \mathcal{K}$, centered at $p_{\pi_1} = (0, 0, 0), p_{\pi_2} = (-14\text{m}, -14\text{m}, 0), p_{\pi_3} = (20\text{m}, -10\text{m}, 0), p_{\pi_4} = (-16\text{m}, 15\text{m}, 0)$, respectively (see Fig. 7.11). Moreover, we consider two cuboid objects of bounding radius $r^O_j = 0.5\text{m}$, and mass $m^O_j = 0.5\text{kg}$,

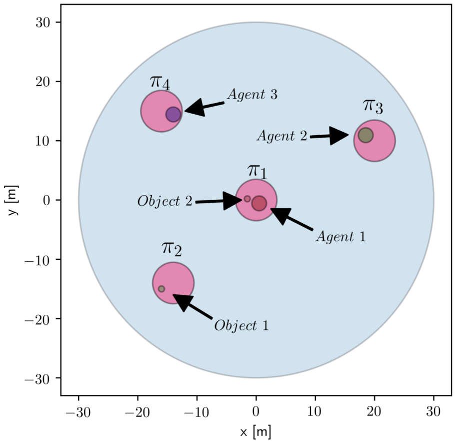


Figure 7.11: The initial workspace of the second simulation example, consisting of 3 agents and 2 objects. The agents and the objects are indicated via their corresponding radii.

$\forall j \in \{1,2\}$, initiated at $x_1^O(0) = [-16\text{m}, 15\text{m}, 0.5\text{m}, 0, 0, 0]^\top$ $x_2^O(0) = [-1.5\text{m}, 0.2\text{m}, 0.5\text{m}, 0, 0, 0]^\top$, which implies that $\mathcal{O}_1(x_1^O(0)) \subset \pi_2$, and $\mathcal{O}_2(x_1^O(0)) \subset \pi_1$. The considered agents consist of a mobile base and a 2-dof rotational robotic arm. The mobile base is rectangular with dimensions 0.5m × 0.5m × 0.2m and mass 0.5kg, and the two arm links have length 1m and mass 0.5kg each. The state vectors of the agents are $q_i = [x_{c_i}, y_{c_i}, q_{i_1}, q_{i_2}]^\top \in \mathbb{R}^4, \dot{q} = [\dot{x}_{c_i}, \dot{y}_{c_i}, \dot{q}_{i_1}, \dot{q}_{i_2}]^\top \in \mathbb{R}^4$, where x_{c_i}, y_{c_i} are the planar position of the bases' center of mass, and q_{i_1}, q_{i_2} the angles of the arms' joints. The geometric characteristics of the considered agents lead to a bounding radius of $r_i = 1.25\text{m}$, $\forall i \in \mathcal{N}$. The atomic propositions are $\Psi_i = \{\text{“}i\text{-}\pi_1\text{”}, \dots, \text{“}i\text{-}\pi_4\text{”}\}$, $\forall i \in \mathcal{N}$, and $\Psi^O = \{\text{“}O_j\text{-}\pi_1\text{”}, \dots, \text{“}O_j\text{-}\pi_4\text{”}\}$, $\forall j \in \mathcal{M}$, indicating whether the agents/objects are in the corresponding regions. The labeling functions are, therefore, $\mathcal{L}_i(\pi_k) = \{\text{“}i\text{-}\pi_k\text{”}\}$, $\mathcal{L}_j^O(\pi_k) = \{\text{“}O_j\text{-}\pi_k\text{”}\}$, $\forall k \in \mathcal{K}, i \in \mathcal{N}, j \in \mathcal{M}$. We test two scenarios with $N = 2, 3$ agents, respectively. We generate the optimal high-level plan for these scenarios and present two indicative transitions of the continuous execution for the second case. The simulations were carried out using Python environment on a laptop computer with 4 cores at 2.6GHz CPU and 8GB of RAM memory.

1. We consider $N = 2$ agents with initial conditions $q_1(0) = [0.5\text{m}, 0, \frac{\pi}{4}\text{rad}, \frac{\pi}{4}\text{rad}]^\top$, $q_2(0) = [18.5\text{m}, 11.5\text{m}, \frac{\pi}{4}\text{rad}, \frac{\pi}{4}\text{rad}]^\top$, $\dot{q}_i(0) = [0,0,0,0]^\top, \forall i \in \{1,2\}$ which imply that $\mathcal{A}_1(q_1(0)) \subset \pi_1$, $\mathcal{A}_2(q_2(0)) \subset \pi_3$, and that no collisions occur at $t = 0$, i.e., $\mathcal{C}_{1,2}(q_1(0), q_2(0)) = \mathcal{C}_{O_1,O_2}(x_1^O(0), x_2^O(0)) = \mathcal{C}_{i,O_j}(q_1(0), x_j^O(0)) = \bot, \forall (i,j) \in \{1,2\} \times \{1,2\}$. We also assume that $\mathcal{AG}_{i,0}(q_i(0), x^O(0)) = \top, \forall i \in \{1,2\}$. We represent the agents' power capabilities with the scalars $\zeta_1 = 2, \zeta_2 =$

Table 7.1: The agent actions for the discrete path of the first simulation example

$\pi_{s,\ell}$	Actions	$\pi_{s,\ell}$	Actions
$\pi_{s,1}$	$(-)$	$\pi_{s,14}$	$(\pi_1 \xrightarrow{T}_{\{1,2\},2} \pi_2)$
$\pi_{s,2}$	$(-,\, \pi_3 \rightarrow_2 \pi_1)$	$\pi_{s,15}$	$(1 \xrightarrow{r} 2,\, 2 \xrightarrow{r} 2)$
$\pi_{s,3}$	$(1 \xrightarrow{g} 2,\, 2 \xrightarrow{g} 2)$	$\pi_{s,16}$	$(\pi_2 \rightarrow_2 \pi_4,\, \pi_2 \rightarrow_2 \pi_4)$
$\pi_{s,4}$	$(\pi_1 \xrightarrow{T}_{\{1,2\},2} \pi_4)$	$\pi_{s,17}$	$(1 \xrightarrow{g} 1,\, 2 \xrightarrow{g} 1)$
$\pi_{s,5}$	$(\pi_4 \xrightarrow{T}_{\{1,2\},2} \pi_1)$	$\pi_{s,18}$	$(\pi_4 \xrightarrow{T}_{\{1,2\},1} \pi_1)$
$\pi_{s,6}$	$(1 \xrightarrow{r} 2,\, 2 \xrightarrow{r} 2)$	$\pi_{s,19}$	$(\pi_1 \xrightarrow{T}_{\{1,2\},1} \pi_4)$
$\pi_{s,7}$	$(\pi_1 \rightarrow_1 \pi_2,\, \pi_1 \rightarrow_2 \pi_2)$	$\pi^\star_{s,20}$	$(-,\, 2 \xrightarrow{r} 1)$
$\pi_{s,8}$	$(1 \xrightarrow{g} 1,\, 2 \xrightarrow{g} 1)$	$\pi^\star_{s,21}$	$(-,\, \pi_4 \rightarrow_2 \pi_3)$
$\pi_{s,9}$	$(\pi_2 \xrightarrow{T}_{\{1,2\},1} \pi_4)$	$\pi^\star_{s,22}$	$(-,\, \pi_3 \rightarrow_2 \pi_4)$
$\pi_{s,10}$	$(1 \xrightarrow{r} 1,\, 2 \xrightarrow{r} 1)$	$\pi^\star_{s,23}$	$(-,\, 2 \xrightarrow{g} 1)$
$\pi_{s,11}$	$(-,\, \pi_4 \rightarrow_2 \pi_3)$	$\pi^\star_{s,24}$	$(\pi_4 \xrightarrow{T}_{\{1,2\},1} \pi_1)$
$\pi_{s,12}$	$(\pi_4 \rightarrow_2 \pi_1,\, \pi_3 \rightarrow_2 \pi_1)$	$\pi^\star_{s,25}$	$(\pi_1 \xrightarrow{T}_{\{1,2\},1} \pi_4)$
$\pi_{s,13}$	$(1 \xrightarrow{g} 2,\, 2 \xrightarrow{g} 2)$		

4 and construct the functions $\Lambda(m^O_1, \zeta_{\mathcal{T}}) = \top$ if and only if $\sum_{\tau\in\mathcal{T}} \zeta_\tau \geq 5$, with $\mathcal{AG}_{\tau,1} = \top \Leftrightarrow \tau \in \mathcal{T}$, and $\Lambda(m^O_2, \zeta_{\mathcal{T}}) = \top$ if and only if $\sum_{\tau\in\mathcal{T}} \zeta_\tau \geq 6$, with $\mathcal{AG}_{\tau,2} = \top \Leftrightarrow \tau \in \mathcal{T}$, i.e., the objects can be transported only if the agents that grasp them have a sum of capability scalars no less than 5 and 6, respectively. Regarding the cost χ, we simply choose the sum of the distances of the transition and transportation regions, i.e., given $\pi_s, \widetilde{\pi}_s$ as in (7.32) such that $\pi_s \rightarrow_s \widetilde{\pi}_s$, we have that $\chi = \sum_{i\in\{1,2\}}\{\|p_{\pi_{k_i}} - p_{\pi_{\widetilde{k}_i}}\|^2\} + \sum_{j\in\{1,2\}} \|p_{\pi_{k^O_j}} - p_{\pi_{\widetilde{k}^O_j}}\|^2\}$. The LTL formula is taken as $(\square\neg$"1-π_3"$) \wedge (\square\lozenge$"2-$\pi_3$"$) \wedge (\square\lozenge$"$O_1$-$\pi_1$"$) \wedge \square($"$O_1$-$\pi_1$" $\rightarrow \bigcirc$"O_1-π_4"$) \wedge (\lozenge$"O_2-π_4"$)$, which represents the following behavior. Agent 1 must never go to region π_3, which must be visited by agent 2 infinitely many times, object 1 must be taken infinitely often to region π_1, always followed by a visit in region π_4, and object 2 must be eventually taken to region π_4.

The resulting transition system $\mathcal{TS}$ consists of 560 reachable states and 7680 transitions and it was created in 3.19 sec. The Büchi automaton $\mathcal{BA}$ contains 7 states and 29 transitions and the product $\widetilde{\mathcal{TS}}$ contains 3920 states and 50976 transitions. Table 7.1 shows the actions of the agents for the derived path, which is the sequence of states $\pi_{s,1}\pi_{s,2}\ldots\ldots(\pi^\star_{s,20},\ldots,\pi^\star_{s,25})^\omega$, where the states with $(^\star)$ constitute the suffix that is run infinitely many times.

Loosely speaking, the derived path describes the following behavior: Agent 2 goes first to π_1 to grasp and transfer object 2 to π_4 and back to π_1 with agent 1. The two agents then navigate to π_2 to take object 1 to π_4. In the following, after agent 2 goes to π_3, they both go to π_1 to transfer object 2 to π_2. Then, they navigate to π_4 to transfer object 1 to π_1 and back. Finally, the actions that are run infinitely many times consist of agent 2 going to from π_4 to π_3 and back, and transferring object 1 to π_1 and π_4 with agent 1. One can verify that the resulting path satisfies the LTL formula. Note also that the regions are not large enough to contain both agents and objects in a grasping configuration, which played an important role in the derivation of the plan. The time taken for the construction of the product $\widetilde{\mathcal{TS}}$ and the derivation of the path was 2.79 sec.

2. We now consider $N = 3$ agents with $q_1(0)$, $q_2(0)$ as in case (i), $q_3(0) = [-14\text{m}, 15\text{m}, \frac{\pi}{4}\text{rad}, \frac{\pi}{4}\text{rad}]^\top \implies \mathcal{A}_3(q_3(0)) \in \pi_4$, $\mathcal{AG}_{3,0}(q_i(0), x^O(0)) = \top$, $\zeta_3 = 3$, and no collisions occurring at $t = 0$. The functions Λ and χ are the same as in case (i). The formula in this scenario is $(\Box\neg\text{“}1\text{-}\pi_3\text{”}) \wedge (\Box\Diamond\text{“}2\text{-}\pi_3\text{”}) \wedge (\Box\Diamond\text{“}O_1\text{-}\pi_1\text{”}) \wedge \Box(\text{“}O_1\text{-}\pi_1\text{”} \rightarrow \Diamond\text{“}O_1\text{-}\pi_4\text{”}) \wedge (\Box\Diamond\text{“}O_2\text{-}\pi_3\text{”})$, which represents the following behavior. Agent 1 must never visit region π_3, which must be visited infinitely many times by agent 2, object 1 must be taken infinitely many times to region π_1, eventually followed by a visit in region π_4, and object 2 must be taken infinitely many times to region π_2.

   The resulting transition system $\mathcal{TS}$ consists of 3112 reachable states and 154960 transitions and it was created in 100.74 sec. The Büchi automaton $\mathcal{BA}$ contains 9 states and 49 transitions and the product $\widetilde{\mathcal{TS}}$ contains 28008 states and 1890625 transitions. Table 7.2 shows the agent actions for the derived path as the sequence of states $\pi_{s,1}\pi_{s,2}\ldots\ldots(\pi^\star_{s,10}, \pi^\star_{s,11})^\omega$. In this case, the three agents navigate first to regions π_2, π_1, and π_1, respectively, and agents 2 and 3 take object 2 to π_3. Next, agent 3 goes to π_2 to transfer object 1 to π_1 and then π_4 with agent 1. The latter transportations occur infinitely often. The time taken for the construction of the product $\widetilde{\mathcal{TS}}$ and the derivation of the path was 4573.89 sec. It is worth noting the exponential increase of the computation time with the simple addition of just one agent, which can be attributed to the centralized manner of the proposed methodology. The necessity, therefore, of less computational, decentralized schemes is evident and constitutes the main focus of our future directions.

Next, we present the continuous execution of the transitions $\pi_{s,1} \rightarrow_s \pi_{s,2}$, and $\pi_{s,3} \rightarrow_s \pi_{s,4}$ for the second simulation scenario. More specifically, Fig. 7.12 depicts the navigation of the three agents $\pi_1 \rightarrow_1 \pi_2$, $\pi_3 \rightarrow_2 \pi_1$, and $\pi_4 \rightarrow_3 \pi_1$, that corresponds to $\pi_{s,1} \rightarrow_s \pi_{s,2}$, with gains $K_z = \text{diag}\{0.01, 0.01, 0.01\}$, $\forall z \in \{1,2,3\}$, and which had a duration of 900 sec. Moreover, Fig. 7.13 depicts the transportation of object 2 by agents 2 and 3, i.e., $\pi_1 \xrightarrow{T}_{\{2,3\}} \pi_3$, that corresponds to $\pi_{s,3} \rightarrow_s \pi_{s,4}$, with load sharing coefficients $c_1 = c_2 = 0.5$, and corresponding time duration 300 sec.

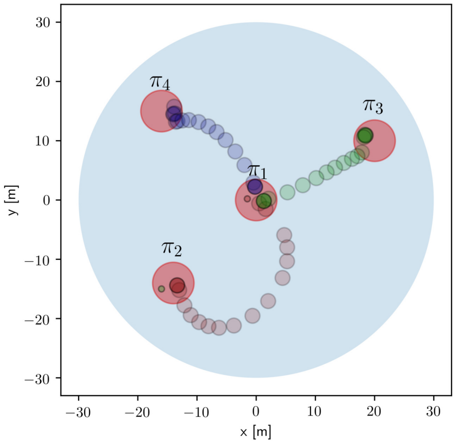


Figure 7.12: The transition $\pi_{s,1} \rightarrow_s \pi_{s,2}$ (a), that corresponds to the navigation of the agents $\pi_1 \rightarrow_1 \pi_2$, $\pi_3 \rightarrow_2 \pi_1$, $\pi_4 \rightarrow_3 \pi_1$.

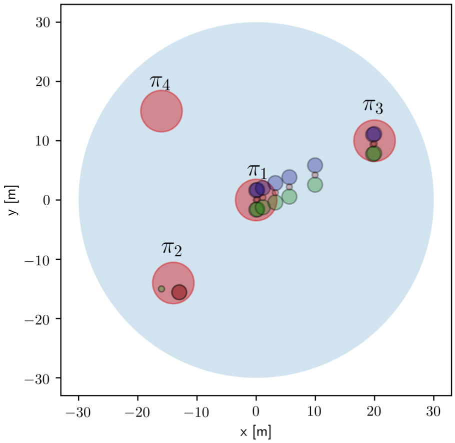


Figure 7.13: The transition $\pi_{s,3} \rightarrow_s \pi_{s,4}$ (b), that corresponds to the transportation $\pi_1 \xrightarrow{T}_{\{2,3\}} \pi_3$.

Table 7.2: The agent actions for the discrete path of the second simulation example

$\pi_{s,\ell}$	Actions
$\pi_{s,1}$	$(-)$
$\pi_{s,2}$	$(\pi_1 \rightarrow_1 \pi_2, \pi_3 \rightarrow_2 \pi_1, \pi_4 \rightarrow_3 \pi_1)$
$\pi_{s,3}$	$(-, 2 \xrightarrow{g} 1, 3 \xrightarrow{g} 2)$
$\pi_{s,4}$	$(-, \pi_1 \xrightarrow{T}_{\{2,3\},2} \pi_3,)$
$\pi_{s,5}$	$(-, -, 3 \xrightarrow{r} 2)$
$\pi_{s,6}$	$(-, -, \pi_3 \rightarrow_3 \pi_2)$
$\pi_{s,7}$	$(1 \xrightarrow{g} 1, 3 \xrightarrow{g} 1)$
$\pi_{s,8}$	$(\pi_2 \xrightarrow{T}_{\{1,3\},1} \pi_1, -)$
$\pi_{s,9}$	$(\pi_1 \xrightarrow{T}_{\{1,3\},1} \pi_4, -)$
$\pi^\star_{s,10}$	$(\pi_4 \xrightarrow{T}_{\{1,3\},1} \pi_1, -)$
$\pi^\star_{s,11}$	$(\pi_1 \xrightarrow{T}_{\{1,3\},1} \pi_4, -)$

7.4 Conclusion and Future Work

We addressed the problem of defining abstractions for cooperative manipulation schemes by designing continuous control protocols. Firstly, we abstracted the motion of an object in the workspace into a timed transition system via a decentralized continuous control law for the trajectory tracking of the object's center of mass. In the following, we presented a novel hybrid control framework for the motion planning of a system comprising of N agents and M objects. We designed appropriate continuous control protocols that guarantee the agent transition and object transportation among predefined regions of interest. In that way, the coupled multi-agent system is abstracted in a finite transition system, which is used to derive plans that satisfy complex LTL formulas. Future efforts will be devoted towards compensating uncertainties in the object's geometrical characteristics, considering non-rigid grasps, as well as incorporating limited sensing information in the second abstraction that deals with the multi-agent-object system.

Chapter 8

Summary and Future Research Directions

This thesis focused on solving the problem of multi-agent and multi-agent-object planning and control under complex specifications expressed as temporal logic formulas. We divided the thesis into three main subproblems, namely formation-control, cooperative object manipulation, and hybrid control synthesis for the temporal-logic-based planning of multi-agent manipulator-endowed and multi-agent-object systems.

In Chapter 3, we proposed a decentralized control protocol based on the prescribed performance control methodology for the formation of tree graph in $SE(3)$, while guaranteeing collision avoidance and connectivity maintenance among the initially connected agents. Simulation results have verified the validity of the proposed approach. Future efforts will be devoted towards ensuring collision avoidance among all the agents as well performing real-time experiments.

In Chapter 4, we proposed two novel decentralized control protocols for the cooperative manipulation of an object by a team of robotic agents without the use of force/torque sensors. Firstly, we designed an adaptive control law based on quaternion feedback to avoid potential representation singularities. Secondly, we employed the prescribed control methodology, to achieve predefined transient and steady state for the center of mass of the object. Both control laws are robust against modeling uncertainties and external disturbances. Future directions will aim at dealing with non-rigid grasps as well as unknown geometric characteristics of the object.

In Chapter 5, we proposed two novel control protocols for the cooperative transportation of an object by a team of robotic agents by using Nonlinear Model Predictive Control. Firstly, we designed a centralized NMPC control protocol, where a central unit computes the control signals of all the agents, and secondly, we designed a decentralized NMPC control protocol, based on inter-agent communication. In both methodologies, we have dealt with inter-agent collision avoidance, collision avoidance between the agents/object and workspace obstacles as well as singularity avoidance. Future efforts will be devoted towards addressing non-rigid grasps, and reducing the NMPC complexity.

In Chapter 6, we proposed decentralized abstractions for teams of robotic agents over predefined regions of interest in the workspace. In particular, based on previous results on navigation functions, we synthesize hybrid controllers for UAVs as well as teams of robotic mobile manipulators, to achieve satisfaction of their individual LTL formulas. The proposed control protocols are decentralized, since each agent is based on local information to determine its actions. Simulation as well as experimental results verify the validity of the proposed approach. Future efforts aim at incorporating cooperative actions in the abstraction in a decentralized manner.

In Chapter 7, we design abstractions for multi-agent systems while incorporating the motion of unactuated objects. Firstly, we employed the PPC design and the prescribed transient and steady state performance from Chapter 4 to design timed transitions for the object among a predefined workspace partition. That allowed us to abstract the motion of the object as a timed transition system and apply MITL formulas. Secondly, we designed navigation function-based control laws for the multi-agent navigation and object transportation among predefined regions of interest. That allowed us to defined a coupled abstracted transition system for the multi-agent-object system, that incorporates the motion and task specifications of the objects and the agents, and apply complex tasks expressed as LTL formulas. Simulation results verified the effectiveness of the proposed methods. Future efforts will aim at designing decentralized abstractions for a team of multiple robotic agents and objects.

Bibliography

[1] W. Ren and R. Beard. Consensus Seeking in Multi-agent Systems under Dynamically Changing Interaction Topologies. *IEEE Transactions on Automatic Control (TAC)*, 50(5):655–661, 2005.

[2] R. Olfati-Saber and R. Murray. Consensus problems in networks of agents with switching topology and time-delays. *IEEE Transactions on Automatic Control (TAC)*, 49(9):1520–1533, 2004.

[3] A. Jadbabaie, J. Lin, and S. Morse. Coordination of groups of mobile autonomous agents using nearest neighbor rules. *IEEE Transactions on Automatic Control (TAC)*, 48(6):988–1001, 2003.

[4] H. Tanner, A. Jadbabaie, and G. J. Pappas. Flocking in fixed and switching networks. *IEEE Transactions on Automatic Control (TAC)*, 52(5):863–868, 2007.

[5] D. V. Dimarogonas and K. Kyriakopoulos. On the rendezvous problem for multiple nonholonomic agents. *IEEE Transactions on Automatic Control (TAC)*, 52(5):916–922, 2007.

[6] M. Egerstedt and X. Hu. Formation constrained multi-agent control. *IEEE Transactions on Robotics and Automation*, 17(6):947–951, 2001.

[7] K. Oh, M. Park, and H. Ahn. A survey of multi-agent formation control. *Automatica*, 53:424–440, 2015.

[8] M. Ji and M. Egerstedt. Distributed Coordination Control of Multi-Agent Systems While Preserving Connectedness. *IEEE Transactions on Robotics (TRO)*, 23(4):693–703, 2007.

[9] M. Zavlanos and G. J. Pappas. Potential Fields for Maintaining Connectivity of Mobile Networks. *IEEE Transactions on Robotics (TRO)*, 23(4):812–816, 2007.

[10] M. Zavlanos and G. J. Pappas. Distributed connectivity control of mobile networks. *IEEE Transactions on Robotics (TRO)*, 24(6):1416–1428, 2008.

[11] D. V. Dimarogonas, S. G. Loizou, K. J. Kyriakopoulos, and M. Zavlanos. A Feedback Stabilization and Collision Avoidance Scheme for Multiple Independent Non-Point Agents. *Automatica*, 42(2):229–243, 2006.

[12] H. Bai and J. T. Wen. Cooperative load transport: A formation-control perspective. *IEEE Transactions on Robotics*, 26(4):742–750, 2010.

[13] A. Nikou, C. K. Verginis, and D. V. Dimarogonas. Robust distance-based formation control of multiple rigid bodies with orientation alignment. *IFAC Proceedings Volumes, Toulouse, France*, 2017.

[14] C. K. Verginis, M. Mastellaro, and D. V. Dimarogonas. Robust quaternion-based cooperative manipulation without force/torque information. *IFAC Proceedings Volumes, Toulouse, France*, 2017.

[15] C. K. Verginis, M. Mastellaro, and D. V. Dimarogonas. Cooperative manipulation without force/torque measurements control design and experiments. *IEEE Transactions on Control Systems Technology*, 2018, submitted.

[16] A. Nikou, C. K. Verginis, S. Heshmati-alamdari, and D. V. Dimarogonas. A nonlinear model predictive control scheme for cooperative manipulation with singularity and collision avoidance. *Proceedings of the 25th IEEE Mediterranean Conference on Control and Automation (MED), Valletta, Malata*, pages 707–712, 2017.

[17] C. K. Verginis, A. Nikou, and D. V. Dimarogonas. Communication-based decentralized cooperative object transportation using nonlinear model predictive control. *European Control Conference (ECC), Limassol, Cyprus*, 2017, submitted.

[18] C. K. Verginis, Z. Xu, and D. V. Dimarogonas. Decentralized motion planning with collision avoidance for a team of uavs under high level goals. *Proceedings of the International Conference on Robotics and Automation (ICRA)*, pages 781–787, 2017.

[19] C. K. Verginis and D. V. Dimarogonas. Robust decentralized abstractions for multiple mobile manipulators. *Proceedings of the Conference on Decision and Control (CDC), Melbourne, Australia*, 2017.

[20] C. K. Verginis and D. V. Dimarogonas. Distributed cooperative manipulation under timed temporal specifications. *American Control Conference (ACC), Seattle, WA, USA*, pages 1358–1363, 2016.

[21] C. K. Verginis and D. V. Dimarogonas. Timed abstractions for distributed cooperative manipulation. *Autonomous Robots*, pages 1–19, 2017.

[22] C. K. Verginis and D. V. Dimarogonas. Multi-agent motion planning and object transportation under high level goals. *IFAC Proceedings Volumes, Toulouse, France*, 2017.

[23] C. K. Verginis and D. V. Dimarogonas. Motion and cooperative transportation planning for multi-agent systems under temporal logic formulas. *IEEE Transactions on Automation Science and Engineering*, 2017, submitted.

[24] L. Lindemann, C. K. Verginis, and D. V. Dimarogonas. Prescribed performance control for signal temporal logic specifications. *Proceedings of the Conference on Decision and Control (CDC), Melbourne, Australia*, 2017.

[25] A. Nikou, S. Heshmati-alamdari, C. K. Verginis, and D. V. Dimarogonas. Decentralized abstractions and timed constrained planning of a general class of coupled multi-agent systems. *Proceedings of the IEEE Conference on Decision and Control (CDC), Melbourne, Australia*, 2017.

[26] B. Siciliano L. Sciavicco, L. Villani, and G. Oriolo. Robotics: modelling, planning and control. *Springer Science & Business Media*, 2010.

[27] R. Horn and C. Johnson. Topics in Matrix Analysis. *Cambridge University Press*, 1994.

[28] H. K. Khalil. Nonlinear Systems. *Prentice Hall*, 2002.

[29] C. P. Bechlioulis and G. A. Rovithakis. A low-complexity global approximation-free control scheme with prescribed performance for unknown pure feedback systems. *Automatica*, 50(4):1217–1226, 2014.

[30] H. Michalska and R. Vinter. Nonlinear Stabilization Using Discontinuous Moving-Horizon Control. *IMA Journal of Mathematical Control and Information*, 11(4):321–340, 1994.

[31] E. D Sontag. Input to state stability: basic concepts and results. *Nonlinear and Optimal Control Theory*, pages 163–220, 2008.

[32] E. D. Sontag. On characterizations of the input-to-state stability property. *Systems and Control Letters*, 24(5):351–359, 1995.

[33] E. D. Sontag. Mathematical control theory: deterministic finite dimensional systems. *Springer Science & Business Media*, 6, 2013.

[34] A. Bressan and B. Piccoli. Introduction to the mathematical theory of control. *American institute of mathematical sciences Springfield*, 2, 2007.

[35] E. Rimon and D. Koditschek. Exact robot navigation using artificial potential functions. *IEEE Transactions on Robotics and Automation (TRA)*, 8(5):501–518, 1992.

[36] S. G. Loizou and K. J. Kyriakopoulos. A feedback-based multiagent navigation framework. *International Journal of Systems Science*, 37(6):377–384, 2006.

[37] D. V. Dimarogonas and K. J. Kyriakopoulos. Decentralized navigation functions for multiple robotic agents with limited sensing capabilities. *Journal of Intelligent & Robotic Systems*, 48(3):411–433, 2007.

[38] C. Baier, J.P. Katoen, and K. G. Larsen. Principles of model checking. *MIT Press*, 2008.

[39] R. Alur and D. L. Dill. A theory of timed automata. *Theoretical computer science*, 126(2):183–235, 1994.

[40] D. Souza and P. Prabhakar. On the expressiveness of mtl in the pointwise and continuous semantics. *International Journal on Software Tools for Technology Transfer*, 9(1):1–4, 2007.

[41] J. Ouaknine and J. Worrell. On the decidability of metric temporal logic. *Annual IEEE Symposium on Logic in Computer Science (LICS'05)*, pages 188–197, 2005.

[42] W. Ren and R. Beard. Consensus seeking in multi-agent systems under dynamically changing interaction topologies. *IEEE Transactions on Automatic Control (TAC)*, 50(5):655–661, 2005.

[43] R. Olfati-Saber and R. Murray. Distributed cooperative control of multiple vehicle formations using structural potential functions. *IFAC Proceedings Volumes*, 15(1):242–248, 2002.

[44] S. Smith, Mireille E Broucke, and Bruce A Francis. Stabilizing a multi-agent system to an equilateral polygon formation. *17th International Symposium on Mathematical Theory of Networks and Systems*, pages 2415–2424, 2006.

[45] J. Hendrickx, B. Anderson, J. Delvenne, and V. Blondel. Directed graphs for the analysis of rigidity and persistence in autonomous agent systems. *International Journal of Robust and Nonlinear Control (IJRNC)*, 17(10-11):960–981, 2007.

[46] B. Anderson, C. Yu, S. Dasgupta, and S. Morse. Control of a three-coleader formation in the plane. *Systems and Control Letters*, 56(9):573–578, 2007.

[47] B. Anderson, C. Yu, B. Fidan, and J. Hendrickx. Rigid Graph Control Architectures for Autonomous Formations. *IEEE Control Systems*, 28:48–63, 2008.

[48] D. V. Dimarogonas and K. Johansson. On the stability of distance-based formation control. *Proceedings of the IEEE Conference on Decision and Control (CDC), Cancun, Mexico*, pages 1200–1205, 2008.

[49] M. Cao, B. Anderson, S. Morse, and C. Yu. Control of acyclic formations of mobile autonomous agents. *Proceedings of the IEEE Conference on Decision and Control (CDC), Cancun, Mexico*, pages 1187–1192, 2008.

[50] C. Yu, B. Anderson, S. Dasgupta, and B. Fidan. Control of minimally persistent formations in the plane. *SIAM Journal on Control and Optimization*, 48(1):206–233, 2009.

[51] L. Krick, M. Broucke, and B. Francis. Stabilisation of infinitesimally rigid formations of multi-robot networks. *International Journal of Control (IJC)*, 82(3):423–439, 2009.

[52] F. Dorfler and B. Francis. Geometric Analysis of the Formation Problem for Autonomous Robots. *IEEE Transactions on Automatic Control (TAC)*, 55(10):2379–2384, 2010.

[53] K. Oh and H. Ahn. Formation control of mobile agents based on inter-agent distance dynamics. *Automatica*, 47(10):2306–2312, 2011.

[54] M. Cao, S. Morse, C. Yu, B. Anderson, and S. Dasgupta. Maintaining a Directed, Triangular Formation of Mobile Autonomous Agents. *Communications in Information and Systems*, 11(1):1, 2011.

[55] T. Summers, Changbin C. Yu, S. Dasgupta, and B. Anderson. Control of minimally persistent leader-remote-follower and coleader formations in the plane. *IEEE Transactions on Automatic Control (TAC)*, 56(12):2778–2792, 2011.

[56] M. Park, K. Oh, and H. Ahn. Modified gradient control for acyclic minimally persistent formations to escape from collinear position. *Proceedings of the IEEE Conference on Decision and Control (CDC), Maui, HI, USA*, pages 1423–1427, 2012.

[57] A. Belabbas, S. Mou, S. Morse, and B. Anderson. Robustness Issues with Undirected Formations. *51st IEEE Conference on Decision and Control (CDC)*, pages 1445–1450, 2012.

[58] K. Oh and H. Ahn. Distance-based undirected formations of single-integrator and double-integrator modeled agents in n-dimensional space. *International Journal of Robust and Nonlinear Control (IJRNC)*, 24(12):1809–1820, 2014.

[59] M. Basiri, A. Bishop, and P. Jensfelt. Distributed control of triangular formations with angle-only constraints. *Systems and Control Letters*, 59(2):147–154, 2010.

[60] Tolga Eren. Formation shape control based on bearing rigidity. *International Journal of Control (IJC)*, 85(9):1361–1379, 2012.

[61] M. Trinh, K. Oh, and H. Ahn. Angle-based control of directed acyclic formations with three-leaders. *2014 International Conference on Mechatronics and Control (ICMC), Jinzhou, China*, pages 2268–2271, 2014.

[62] S. Zhao and D. Zelazo. Bearing rigidity and almost global bearing-only formation stabilization. *IEEE Transactions on Automatic Control (TAC)*, 61(5):1255–1268, 2016.

[63] A. Bishop, M. Deghat, B. Anderson, and Y. Hong. Distributed formation control with relaxed motion requirements. *International Journal of Robust and Nonlinear Control (IJRNC)*, 25(17):3210–3230, 2015.

[64] K. Fathian, D. Rachinskii, M. Spong, and N. Gans. Globally asymptotically stable distributed control for distance and bearing based multi-agent formations. *Proceedings of the IEEE American Control Conference (ACC), Boston, MA, USA*, pages 4642–4648, 2016.

[65] Y. Karayiannidis, D. V. Dimarogonas, and D. Kragic. Multi-agent average consensus control with prescribed performance guarantees. *Proceedings of the IEEE Conference on Decision and Control (CDC), Maui, HI, USA*, pages 2219–2225, 2012.

[66] C. Bechlioulis and K. Kyriakopoulos. Robust model-free formation control with prescribed performance and connectivity maintenance for nonlinear multi-agent systems. *Proceedings of the IEEE Conference on Decision and Control (CDC), Los Angeles, CA, USA*, pages 4509–4514, 2014.

[67] M. Mesbahi and M. Egerstedt. Graph Theoretic Methods in Multiagent Networks. *Princeton University Press*, 2010.

[68] T. Lee, D. E. Chang, and Y. Eun. Attitude control strategies overcoming the topological obstruction on so (3). *American Control Conference (ACC), Seattle, WA, USA*, pages 2225–2230, 2017.

[69] T. Lee, M .Leok, and N. H. McClamroch. Control of complex maneuvers for a quadrotor uav using geometric methods on se(3). *arXiv:1003.2005*, 2010.

[70] S. P. Bhat and D. S. Bernstein. A topological obstruction to continuous global stabilization of rotational motion and the unwinding phenomenon. *Systems & Control Letters*, 39(1):63–70, 2000.

[71] C. G. Mayhew, R. G. Sanfelice, and A. R. Teel. Quaternion-based hybrid control for robust global attitude tracking. *IEEE Transactions on Automatic Control*, 56(11):2555–2566, 2011.

[72] S. A. Schneider and R. H. Cannon. Object impedance control for cooperative manipulation: Theory and experimental results. *IEEE Transactions on Robotics and Automation*, 8(3):383–394, 1992.

[73] T. G. Sugar and V. Kumar. Control of cooperating mobile manipulators. *IEEE Transactions on robotics and automation*, 18(1):94–103, 2002.

[74] O. Khatib, K. Yokoi, K. Chang, D. Ruspini, R. Holmberg, and A. Casal. Decentralized cooperation between multiple manipulators. *IEEE International Workshop onRobot and Human Communication*, pages 183–188, 1996.

[75] Y.-H. Liu, S. Arimoto, and T. Ogasawara. Decentralized cooperation control: non-communication object handling. *Proceedings of the IEEE Conference on Robotics and Automation (ICRA)*, 3:2414–2419, 1996.

[76] Y.-H. Liu and S. Arimoto. Decentralized adaptive and nonadaptive position/force controllers for redundant manipulators in cooperations. *The International Journal of Robotics Research*, 17(3):232–247, 1998.

[77] Mohamed Zribi and Shuheen Ahmad. Adaptive control for multiple cooperative robot arms. *Proceedings of the IEEE Conference on Decision and Control (CDC)*, pages 1392–1398, 1992.

[78] L. Gudiño-Lau, M. Arteaga, L. Munoz, and V. Parra-Vega. On the control of cooperative robots without velocity measurements. *IEEE Transactions on Control Systems Technology*, 12(4):600–608, 2004.

[79] J. T. Wen and K. Kreutz-Delgado. Motion and force control of multiple robotic manipulators. *Automatica*, 28(4):729–743, 1992.

[80] T. Yoshikawa and X.-Z. Zheng. Coordinated dynamic hybrid position/force control for multiple robot manipulators handling one constrained object. *The International Journal of Robotics Research*, 12(3):219–230, 1993.

[81] C. D. Kopf. Dynamic two arm hybrid position/force control. *Robotics and Autonomous Systems*, 5(4):369–376, 1989.

[82] F. Caccavale, P. Chiacchio, and S. Chiaverini. Task-space regulation of cooperative manipulators. *Automatica*, 36(6):879–887, 2000.

[83] F. Caccavale, P. Chiacchio, A. Marino, and L. Villani. Six-dof impedance control of dual-arm cooperative manipulators. *IEEE/ASME Transactions On Mechatronics*, 13(5):576–586, 2008.

[84] D. Heck, D. Kostic, A. Denasi, and H. Nijmeijer. Internal and external force-based impedance control for cooperative manipulation. *Proceedings of the IEEE European Control Conference (ECC)*, pages 2299–2304, 2013.

[85] S. Erhart and S. Hirche. Adaptive force/velocity control for multi-robot cooperative manipulation under uncertain kinematic parameters. *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 307–314, 2013.

[86] S. Erhart, D. Sieber, and S. Hirche. An impedance-based control architecture for multi-robot cooperative dual-arm mobile manipulation. *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 315–322, 2013.

[87] Y. Kume, Y. Hirata, and K. Kosuge. Coordinated motion control of multiple mobile manipulators handling a single object without using force/torque sensors. *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4077–4082, 2007.

[88] J. Szewczyk, F. Plumet, and P. Bidaud. Planning and controlling cooperating robots through distributed impedance. *Journal of Robotic Systems*, 19(6):283–297, 2002.

[89] A. Tsiamis, C. K. Vergini, C. P. Bechlioulis, and K. J. Kyriakopoulos. Cooperative manipulation exploiting only implicit communication. *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 864–869, 2015.

[90] F. Ficuciello, A. Romano, L. Villani, and B. Siciliano. Cartesian impedance control of redundant manipulators for human-robot co-manipulation. *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 2120–2125, 2014.

[91] A.-N. Ponce-Hinestroza, J.-A. Castro-Castro, H.-I. Guerrero-Reyes, V. Parra-Vega, and E. Olguỳn-Dỳaz. Cooperative redundant omnidirectional mobile manipulators: Model-free decentralized integral sliding modes and passive velocity fields. *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 2375–2380, 2016.

[92] W. Gueaieb, F. Karray, and S. Al-Sharhan. A robust hybrid intelligent position/force control scheme for cooperative manipulators. *IEEE/ASME Transactions on Mechatronics*, 12(2):109–125, 2007.

[93] Z. Li, C. Yang, C. Y. Su, S. Deng, F. Sun, and W. Zhang. Decentralized fuzzy control of multiple cooperating robotic manipulators with impedance interaction. *IEEE Transactions on Fuzzy Systems*, 23(4):1044–1056, 2015.

[94] K. G. Tzierakis and F. N. Koumboulis. Independent force and position control for cooperating manipulators. *Journal of the Franklin Institute*, 340(6):435–460, 2003.

[95] A. Marino. Distributed adaptive control of networked cooperative mobile manipulators. *IEEE Transactions on Control Systems Technology*, 2017.

[96] M. Ciocarlie, F. Hicks, R. Holmberg, J. Hawke, M. Schlicht, J. Gee, S. Stanford, and R. Bahadur. The velo gripper: A versatile single-actuator design for

enveloping, parallel and fingertip grasps. *The International Journal of Robotics Research*, 2014.

[97] A. Petitti, A. Franchi, D. Di Paola, and A. Rizzo. Decentralized motion control for cooperative manipulation with a team of networked mobile manipulators. *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 441–446, 2016.

[98] F. Aghili. Self-tuning cooperative control of manipulators with position/orientation uncertainties in the closed-kinematic loop. *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4187–4193, 2011.

[99] S. Erhart and S. Hirche. Model and analysis of the interaction dynamics in cooperative manipulation tasks. *IEEE Transactions on Robotics*, 32(3):672–683, 2016.

[100] S. Erhart and S. Hirche. Internal force analysis and load distribution for cooperative multi-robot manipulation. *IEEE Transactions on Robotics*, 31(5):1238–1243, 2015.

[101] Z. Wang and M. Schwager. Multi-robot manipulation with no communication using only local measurements. *Proceedings of the IEEE Conference on Decision and Control (CDC)*, pages 380–385, 2015.

[102] L. Chaimowicz, M. F. M. Campos, and V. Kumar. Hybrid systems modeling of cooperative robots. *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 3:4086–4091, 2003.

[103] T. D. Murphey and M. Horowitz. Adaptive cooperative manipulation with intermittent contact. *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 1483–1488, 2008.

[104] Z. Wang and M. Schwager. Kinematic multi-robot manipulation with no communication using force feedback. *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 427–432, 2016.

[105] H. G. Tanner, S. G. Loizou, and K. J. Kyriakopoulos. Nonholonomic navigation and control of cooperating mobile manipulators. *IEEE Transactions on Robotics and Automation*, 19(1):53–64, 2003.

[106] R. Campa, K. Camarillo, and L. Arias. Kinematic modeling and control of robot manipulators via unit quaternions: Application to a spherical wrist. *Proceedings of the IEEE Conference on Decision and Control (CDC)*, pages 6474–6479, 2006.

[107] J.-J. E. Slotine and W. Li. On the adaptive control of robot manipulators. *The International Journal of Robotics Research*, 6(3):49–59, 1987.

[108] J.-J. E. Slotine and W. Li. Applied nonlinear control. *prentice-Hall Englewood Cliffs, NJ*, 199(1), 1991.

[109] R. Eric, P. N. S. Surya F., and Marc. V-rep: a versatile and scalable robot simulation framework. *Proceedings of The International Conference on Intelligent Robots and Systems (IROS)*, 2013.

[110] Trossenrobotics. Widowx robot arm kit w/ros. *www.trossenrobotics.com/widowxrobotarm.*

[111] Robotis. Dynamixel actuators. *www.robotis.us/dynamixel.*

[112] Trossenrobotics. Arbotix-m robocontroller. *http://www.trossenrobotics.com/p/arbotix-robot-controller.aspx.*

[113] K. Oliveira and M. Morari. Contractive model predictive control for constrained nonlinear systems. *IEEE Transactions on Automatic Control*, 45(6):1053–1071, 2000.

[114] D. Koditschek and E. Rimon. Robot navigation functions on manifolds with boundary. *Advances in Applied Mathematics*, 11(4):412–442, 1990.

[115] R. Findeisen, L. Imsland, F. Allgöwer, and B. A. Foss. State and output feedback nonlinear model predictive control: An overview. *European Journal of Control*, 9(2-3):190–206, 2003.

[116] H. Chen and F. Allgöwer. A quasi-infinite horizon nonlinear model predictive control scheme with guaranteed stability. *Automatica*, 34(10):1205–1217, 1998.

[117] R. Findeisen, L. Imsland, F. Allgöwer, and B. Foss. Towards a sampled-data theory for nonlinear model predictive control. *New Trends in Nonlinear Dynamics and Control and their Applications*, pages 295–311, 2003.

[118] F. Fontes. A general framework to design stabilizing nonlinear model predictive controllers. *Systems and Control Letters*, 42(2):127–143, 2001.

[119] L. Grüne and J. Pannek. Nonlinear Model Predictive Control. *Springer London*, 2011.

[120] E. Camacho and C. Bordons. Nonlinear model predictive control: An introductory review. *Assessment and Future Directions of Nonlinear Model Predictive Control*, pages 1–16, 2007.

[121] B. Kouvaritakis and M. Cannon. Nonlinear predictive control: Theory and practice. *Iet*, (61), 2001.

[122] J. Frasch, A. Gray, M. Zanon, H. Ferreau, S. Sager, F. Borrelli, and M. Diehl. An auto-generated nonlinear mpc algorithm for real-time obstacle cvoidance of ground vehicles. *European Control Conference (ECC)*, 2013.

[123] F. Fontes, L. Magni, and Éva Gyurkovics. Sampled-data model predictive control for nonlinear time-varying systems: Stability and robustness. *Assessment and Future Directions of Nonlinear Model Predictive Control*, pages 115–129, 2007.

[124] V. Lippiello and F. Ruggiero. Cartesian impedance control of a uav with a robotic arm. *IFAC Proceedings Volumes*, 45(22):704 – 709, 2012.

[125] S. Liu, C. Wang, G. Liang, H. Chen, and X. Wu. *Proceedings of the IEEE International Conference on Information and Automation*, pages 2504–2511, 2015.

[126] Y. Bocheng, D. Xiwang, S. Zongying, and Z. Yisheng Zhong. Formation control for quadrotor swarm systems: Algorithms and experiments. *Proceedings of the 32nd Chinese Control Conference (CCC)*, pages 7099–7104, 2013.

[127] N. Koeksal, B. Fidan, and K. Bueyuekkabasakal. Real-time implementation of decentralized adaptive formation control on multi-quadrotor systems. *European Control Conference (ECC), Linz, Austria*, pages 3162–3167, 2015.

[128] Z. Hou and I.Fantoni. Composite nonlinear feedback-based bounded formation control of multi-quadrotor systems. *European Control Conference (ECC), Aalborg, Denmark*, pages 1538–1543, 2016.

[129] R. L. Pereira and K. H. Kienitz. Tight formation flight control based on h-infinity approach. *24th Mediterranean Conference on Control and Automation (MED), Athens, Greece*, 2016.

[130] X. Dong, B. Yu, Z. Shi, and Y. Zhong. Time-varying formation control for unmanned aerial vehicles: Theories and applications. *IEEE Transactions on Control Systems Technology*, 23(1):340–348, 2015.

[131] D. A. Mercado, R. Castro, and R. Lozano. Quadrotors flight formation control using a leader-follower approach. *European Control Conference (ECC), Zürich, Switzerland*, pages 3858–3863, 2013.

[132] V. Roldao, R. Cunha, D. Cabecinhas, C. Silvestre, and P. Oliveira. A novel leader-following strategy applied to formations of quadrotors. *European Control Conference (ECC), Zürich, Switzerland*, pages 1817–1822, 2013.

[133] S. Ulrich. Nonlinear passivity-based adaptive control of spacecraft formation flying. *American Control Conference (ACC), Boston, MA, USA*, pages 7432–7437, 2016.

[134] N. D. Hao, B. Mohamed, H. Rafaralahy, and M. Zasadzinski. Formation of leader-follower quadrotors in cluttered environment. *American Control Conference (ACC), Boston, MA, USA*, pages 6477–6482, 2016.

[135] K. A. Ghamry and Y. Zhang. Formation control of multiple quadrotors based on leader-follower method. *International Conference on Unmanned Aircraft Systems (ICUAS), Denver, CO, USA*, pages 1037–1042, 2015.

[136] Z. N. Sunberg, M. J. Kochenderfer, and M. Pavone. Optimized and trusted collision avoidance for unmanned aerial vehicles using approximate dynamic programming. *Proceedings of the International Conference on Robotics and Automation (ICRA), Stockholm, Sweden*, pages 1455–1461, 2016.

[137] A. Eskandarpour and V. J. Majd. Cooperative formation control of quadrotors with obstacle avoidance and self collisions based on a hierarchical mpc approach. *Proceedings of the RSI/ISM International Conference on Robotics and Mechatronics (ICRoM), Tehran, Iran*, pages 351–356, 2014.

[138] Y. Zhou and J. S. Baras. Reachable set approach to collision avoidance for uavs. *Proceedings of the IEEE Conference Decision and Control (CDC), Osaka, Japan*, pages 5947–5952, 2015.

[139] A. Pierson, A. Ataei, I. C. Paschalidis, and M. Schwager. Cooperative multi-quadrotor pursuit of an evader in an environment with no-fly zones. *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), Stockholm, Sweden*, pages 320–326, 2016.

[140] M. M. Quottrup, T. Bak, and R. I. Zamanabadi. Multi-robot planning : a timed automata approach. *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), New Orleans, LA, USA*, 5:4417–4422, 2004.

[141] S. G. Loizou and K. J. Kyriakopoulos. Automated planning of motion tasks for multi-robot systems. *Proceedings of the IEEE Conference on Decision and Control (CDC), European Control Conference (ECC), Seville, Spain*, pages 78–83, 2005.

[142] I. Filippidis, D. V. Dimarogonas, and K. J. Kyriakopoulos. Decentralized Multi-Agent Control from Local LTL Specifications. *51st IEEE Conference on Decision and Control (CDC), Maui, Hawaii, USA*, pages 6235–6240, 2012.

[143] M. Guo and D. V. Dimarogonas. Multi-Agent Plan Reconfiguration Under Local LTL Specifications. *The International Journal of Robotics Research (IJRR)*, 34(2):218–235, 2015.

[144] V. Nenchev and C. Belta. Receding horizon robot control in partially unknown environments with temporal logic constraints. *European Control Conference (ECC), Aalborg, Denmark*, pages 2614–2619, 2016.

[145] S. Feyzabadi and S. Carpin. Multi-objective planning with multiple high level task specifications. *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), Stockholm, Sweden*, pages 5483–5490, 2016.

[146] M. Guo, J. Tumova, and D. V. Dimarogonas. Hybrid control of multi-agent systems under local temporal tasks and relative-distance constraints. *Proceedings of the IEEE Conference on Decision and Control (CDC), Osaka, Japan*, pages 1701–1706, 2015.

[147] G. Fainekos, A. Girard, K.G. Hadas, and G. J. Pappas. Temporal Logic Motion Planning for Dynamic Robots. *Automatica*, 45(2):343–352, 2009.

[148] M. Kloetzer and C. Belta. Ltl planning for groups of robots. *Proceedings of the IEEE International Conference on Networking, Sensing and Control (ICNSC), Ft. Lauderdale, FL, USA*, pages 578–583, 2006.

[149] A. Ulusoy, S. L. Smith, X. C. Ding, C. Belta, and D. Rus. Optimality and robustness in multi-robot path planning with temporal logic constraints. *The International Journal of Robotics Research*, 32(8):889–911, 2013.

[150] Z. Zhang and R. V. Cowlagi. Motion-planning with global temporal logic specifications for multiple nonholonomic robotic vehicles. *Proceedings of the IEEE American Control Conference (ACC), Boston, MA, USA*, pages 7098–7103, 2016.

[151] D. Aksaray, C.-I. Vasile, and C. Belta. Dynamic routing of energy-aware vehicles with temporal logic constraints. *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), Stockholm, Sweden*, pages 3141–3146, 2016.

[152] C. Belta, A. Bicchi, M. Egerstedt, E. Frazzoli, E. Klavins, and G. J. Pappas. Symbolic planning and control of robot motion. *IEEE Robotics & Automation Magazine*, 14(1):61–70, 2007.

[153] A. Bhatia, L. E. Kavraki, and M. Y. Vardi. Sampling-based motion planning with temporal goals. *IEEE International Conference on Robotics and Automation (ICRA), Anchorage, AK, USA*, pages 2689–2696, 2010.

[154] A. Bhatia, M. R. Maly, L. E. Kavraki, and M. Y. Vardi. Motion planning with complex goals. *IEEE Robotics & Automation Magazine*, 18(3):55–64, 2011.

[155] R. V. Cowlagi and Z. Zhang. Motion-planning with linear temporal logic specifications for a nonholonomic vehicle kinematic model. *Proceedings of the IEEE American Control Conference (ACC), Boston, MA, USA*, pages 6411–6416, 2016.

[156] Y. Diaz-Mercado, A. Jones, C. Belta, and M. Egerstedt. Correct-by-construction control synthesis for multi-robot mixing. *Proceedings of the IEEE Conference on Decision and Control (CDC), Osaka, Japan*, pages 221–226, 2015.

[157] S. Karaman and E. Frazzoli. Vehicle routing problem with metric temporal logic specifications. *Proceedings of the IEEE Conference on Decision and Control (CDC), Cancun, Mexico*, pages 3953–3958, 2008.

[158] S. Karaman and E. Frazzoli. Complex mission optimization for multiple-uavs using linear temporal logic. *Proceedings of the IEEE American Control Conference (ACC), Seattle, WA, USA*, pages 2003–2009, 2008.

[159] J. Xiaoting and N. Yifeng. Robust strategy planning for uav with ltl specifications. *Proceedings of the IEEE Chinese Control Conference (CCC), Yinchuan, China*, pages 2890–2895, 2016.

[160] C. Belta, V. Isler, and G. J. Pappas. Discrete abstractions for robot motion planning and control in polygonal environments. *IEEE Transactions on Robotics*, 21(5):864–874, 2005.

[161] C. Belta and L. C. Habets. Controlling a class of nonlinear systems on rectangles. *IEEE Transactions on Automatic Control*, 51(11):1749–1759, 2006.

[162] G. Reissig. Computing abstractions of nonlinear systems. *IEEE Transactions on Automatic Control*, 56(11):2583–2598, 2011.

[163] A. Tiwari. Abstractions for hybrid systems. *Formal Methods in System Design*, 32(1):57–83, 2008.

[164] M. Rungger, A. Weber, and G. Reissig. State space grids for low complexity abstractions. *Proceedings of the IEEE Conference on Decision and Control (CDC), Osaka, Japan*, pages 6139–6146, 2015.

[165] D. Boskos and D. V. Dimarogonas. Decentralized abstractions for feedback interconnected multi-agent systems. *Proceedings of the IEEE Conference on Decision and Control (CDC), Osaka, Japan*, pages 282–287, 2015.

[166] C. Belta and V. Kumar. Abstraction and control for groups of robots. *IEEE Transactions on robotics*, 20(5):865–875, 2004.

[167] Y.-K. Choi, J.-W. Chang, W. Wang, M.-S. Kim, and G. Elber. Continuous collision detection for ellipsoids. *Transactions on visualization and Computer Graphics*, 15(2):311–325, 2009.

[168] P. Gastin and D. Oddoux. Ltl2ba tool, http://www.lsv.ens-cachan.fr/ gastin/ltl2ba/. 2012.

[169] B. Siciliano and O. Khatib. *Springer handbook of robotics.* Springer Science & Business Media, 2008.

[170] D. Panagou. A distributed feedback motion planning protocol for multiple unicycle agents of different classes. *IEEE Transactions on Automatic Control*, 62(3):1178–1193, 2017.

[171] R. Alur, T. Feder, and T. A. Henzinger. The benefits of relaxing punctuality. *Journal of the ACM (JACM)*, 43(1):116–146, 1996.

[172] A. Nikou, J. Tumova, and D. V. Dimarogonas. Cooperative task planning of multi-agent systems under timed temporal specifications. *Proceedings of the IEEE American Control Conference (ACC), Boston, MA, USA*, pages 7104–7109, July 2016.

[173] S. Karaman and E. Frazzoli. Linear temporal logic vehicle routing with applications to multi-uav mission planning. *International Journal of Robust and Nonlinear Control*, 21(12):1372–1395, 2011.

[174] S. G. Loizou. The multi-agent navigation transformation: Tuning-free multi-robot navigation. *Proceedings of Robotics: Science and Systems*, 2014.

[175] S. G. Loizou. The navigation transformation. *IEEE Transactions on Robotics*, 33(6):1516–1523, 2017.

[176] D. Z. Chen. Sphere packing problem. *Encyclopedia of Algorithms*, pages 1–99, 2008.

[177] M. R. Cutkosky. *Robotic grasping and fine manipulation*, volume 6. Springer Science & Business Media, 2012.

[178] M. F. Reis, A. C. Leite, and F. Lizarralde. Modeling and control of a multifingered robot hand for object grasping and manipulation tasks. *IEEE Conference on Decision and Control (CDC), Osaka, Japan*, pages 159–164, 2015.